

БИОНИКА ИНТЕЛЛЕКТА

ИНФОРМАЦИЯ, ЯЗЫК, ИНТЕЛЛЕКТ

№ 2 (91)

2018

НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Основан в октябре 1967 г.

Учредитель и издатель
Харьковский национальный университет радиоэлектроники

Периодичность издания – *2 раза в год*



Бодянский Е.В.¹, Шафроненко А.Ю.², Патлань Е.В.³

¹ХНУРЕ, г. Харьков, Украина, yevgeniy.bodyanskiy@nure.ua

²ХНУРЕ, г. Харьков, Украина, alina.shafronenko@nure.ua

³НТУ «Харьковский политехнический институт»,
г. Харьков, Украина, sonnyanishakal@gmail.com

НЕЧЕТКАЯ КЛАСТЕРИЗАЦИЯ МАССИВОВ ДАННЫХ НА ОСНОВЕ ЭВОЛЮЦИОННОГО МЕТОДА ОПТИМИЗАЦИИ КОШАЧЬИХ СТАЙ

Рассмотрена задача нечеткой кластеризации массива наблюдений на основе нечеткого вероятностного подхода, в основу которого положен алгоритм нечетких С-средних, переформулированный в задачу безусловной многоэкстремальной оптимизации.

НЕЧЕТКАЯ КЛАСТЕРИЗАЦИЯ, МНОГОЭКСТРЕМАЛЬНАЯ ОПТИМИЗАЦИЯ, ЭВОЛЮЦИОННЫЙ АЛГОРИТМ

Є.В. Бодяньскій, А.Ю. Шафроненко, Е.В. Патлань. Нечітка кластеризація масивів даних на основі еволюційного методу оптимізації кошачих зграй. Розглянуто задачу нечіткої кластеризації масиву спостережень на основі нечіткого імовірнісного підходу, в основу якого покладено алгоритм нечітких С-середніх, який було переформулювало в задачу безумовної багатоекстремальної оптимізації.

НЕЧІТКА КЛАСТЕРИЗАЦІЯ, БАГАТОЭКСТРЕМАЛЬНА ОПТИМІЗАЦІЯ, ЕВОЛЮЦІЙНИЙ АЛГОРИТМ

Ye. Bodyankiy, A. Shafronenko, K. Patlan. Fuzzy clustering of data arrays based on the evolutionary method of cat swarm optimization. The problem of fuzzy clustering of an array of observations based on a fuzzy probabilistic approach, based on the algorithm of fuzzy C-means, reformulated into the problem of unconditional multi-extremal optimization, is considered.

FUZZY CLUSTERING, MULTIEXTREMAL OPTIMIZATION, EVOLUTIONARY ALGORITHM

Введение

Задача кластеризации массивов многомерных данных, основной целью которой является нахождение однородных в смысле принятой метрики классов наблюдений, является важной частью интеллектуального анализа данных Data Mining [1-3], интенсивно развивающегося в настоящее время. В рамках традиционного кластерного анализа априори предполагается, что каждый вектор-наблюдение может принадлежать только одному классу-кластеру, хотя в реальных приложениях достаточно часто возникает ситуация, когда это наблюдение с разными уровнями принадлежности (возможности, вероятности) относится сразу к нескольким взаимно перекрывающимся кластерам. Подобная ситуация является предметом рассмотрения нечеткого (фаззи-) кластерного анализа [4-6], в рамках которого необходимо оценить не только факт принадлежности каждого наблюдения к конкретным классам, но и дать количественную оценку уровня этой принадлежности.

Исходной информацией для решения задачи нечеткой кластеризации является массив многомерных векторов-данных, образованный выборкой наблюдений $X = \{x(1), x(2), \dots, x(k), \dots, x(N)\} \subset R^n$, где k — в общем случае номер наблюдения в исходном массиве, $x(k) = (x_1(k), x_2(k), \dots, x_i(k), \dots, x_n(k))^T$. Результатом кластеризации является разбиение этого массива на m пересекающихся классов Cl_j с прототипами — центроидами $c_j \in R^n, j = 1, 2, \dots, m$,

при этом наряду с нахождением центроидов c_j должен быть оценен уровень принадлежности $0 < U_j(k) < 1$ каждого $x(k)$ к каждому из кластеров Cl_j .

Заметим, что исходные данные должны быть преобразованы (центрированы, нормированы, кодированы, стандартизированы) так, чтобы все наблюдения принадлежали либо некоторому гиперкубу (обычно $[-1, 1]^n$ или $[0, 1]^n$), либо лежали на гиперсфере с единичным радиусом. В задачах нечеткой кластеризации с использованием наиболее распространенного метода нечетких С-средних (FCM) [4] данные обычно преобразуются так, чтобы исходная выборка имела вид

$$\tilde{X} = (\tilde{x}(1), \tilde{x}(2), \dots, \tilde{x}(k), \dots, \tilde{x}(N)) \subset R^n,$$

$$\tilde{x}(k) = (\tilde{x}_1(k), \dots, \tilde{x}_i(k), \dots, \tilde{x}_n(k))^T, \quad -1 \leq \tilde{x}_i(k) \leq 1,$$

$$1 < m < N, \quad 1 \leq j \leq m, \quad 1 \leq i \leq n, \quad 1 \leq k \leq N.$$

И хотя на сегодня кроме FCM разработано множество методов и алгоритмов нечеткой кластеризации со своими достоинствами и недостатками, все они позволяют отыскать только локальный экстремум принятой целевой функции [6, 7], что ведет к тому, что использование процедур оптимизации (нелинейного программирования) на основе производных принятого критерия в общем случае не позволяет получить искомое наилучшее решение. Преодолеть эту проблему можно, многократно решая задачу при разных начальных

условиях и выбирая наилучший вариант из множества полученных. Понятно, что подобный подход существенно увеличивает время решения задачи.

В связи с этим в [8] был предложен метод нечетких J-средних (FJM), сочетающий в себе стандартный FCM с элементами глобального случайного поиска [9-12]. FJM обеспечивает нахождение глобального экстремума с высокой вероятностью, однако время поиска может быть достаточно велико, что естественно, ограничивает возможности этого подхода.

Преодолеть указанные затруднения можно, воспользовавшись аппаратом гибридных систем вычислительного интеллекта (HSCI) [13-17], сочетающих в себе обучаемость искусственных нейронных сетей, интерпретируемость результатов и возможность работы в условиях перекрывающихся классов систем нечеткого вывода и высокую скорость отыскания глобального экстремума, обеспечиваемую эволюционными алгоритмами оптимизации, основанными на «роях частиц» (PCO).

1. Алгоритм нечеткой кластеризации на основе оптимизации с помощью кошачьих стай

В основе широко распространенного алгоритма вероятностной нечеткой кластеризации [5], лежит процедура минимизации целевой функции

$$E(U_j(k), c_j) = \sum_{k=1}^N \sum_{j=1}^m U_j^\beta(k) \|\tilde{x}(k) - c_j\|^2 \quad (1)$$

при ограничениях

$$\sum_{j=1}^m U_j(k) = 1, \quad 0 \leq \sum_{j=1}^m U_j(k) \leq N, \quad (2)$$

(здесь β — неотрицательный параметр фаззификации (фаззификатор), задающий размытость границ между кластерами), в основе которой лежат стандартные методы нелинейного (при $\beta = 2$ — квадратичного) программирования.

Записав функцию Лагранжа

$$L(U_j(k), c_j, \lambda(k)) = \sum_{k=1}^N \sum_{j=1}^m U_j^\beta(k) \|\tilde{x}(k) - c_j\|^2 + \sum_{k=1}^N \lambda(k) (\sum_{j=1}^m U_j(k) - 1)$$

(здесь $\lambda(k)$ — неопределенные множители Лагранжа) и решив систему уравнений Каруша-Куна-Таккера

$$\begin{cases} \frac{\partial L(U_j(k), c_j, \lambda(k))}{\partial U_j(k)} = 0, \\ \nabla_{c_j} L(U_j(k), c_j, \lambda(k)) = \vec{0}, \\ \frac{\partial L(U_j(k), c_j, \lambda(k))}{\partial \lambda_j(k)} = 0, \end{cases}$$

получаем искомое решение вида

$$\begin{cases} U_j(k) = \frac{(\|\tilde{x}(k) - c_j\|^2)^{\frac{1}{1-\beta}}}{\sum_{l=1}^m (\|\tilde{x}(k) - c_l\|^2)^{\frac{1}{1-\beta}}}, \\ c_j = \frac{\sum_{k=1}^N U_j^\beta(k) \tilde{x}(k)}{\sum_{k=1}^N U_j^\beta(k)}, \end{cases} \quad (3)$$

которое при $\beta = 2$ совпадает с алгоритмом нечетких C-средних (FCM) Дж. Бездека [4]:

$$\begin{cases} U_j(k) = \frac{(\|\tilde{x}(k) - c_j\|^2)^{-1}}{\sum_{l=1}^m (\|\tilde{x}(k) - c_l\|^2)^{-1}}, \\ c_j = \frac{\sum_{k=1}^N U_j^2(k) \tilde{x}(k)}{\sum_{k=1}^N U_j^2(k)}. \end{cases} \quad (4)$$

В [7] была доказана сходимость процедур (3), (4) к локальному минимуму, при этом достижение глобального экстремума в общем случае не гарантируется.

В работах [18-19] задача условной оптимизации (1), (2) была переформулирована в задачу безусловной оптимизации целевой функции вида

$$E(c_j) = \sum_{k=1}^N (\sum_{j=1}^m \|\tilde{x}(k) - c_j\|^{2(1-\beta)})^{1-\beta}, \quad (5)$$

при $\beta = 2$ принимающей вид

$$E(c_j) = \sum_{k=1}^N (\sum_{j=1}^m \|\tilde{x}(k) - c_j\|^2)^{-1}, \quad (6)$$

при этом интересно отметить, что в процессе минимизации (5), (6) отыскиваются только координаты центроидов $A_j, j = 1, 2, \dots, m$, а для нахождения уровней нечеткой принадлежности могут быть использованы первые уравнения соотношений (3), (4).

Таким образом, задача нечеткой кластеризации может быть сведена к поиску глобального экстремума целевых функций (5), (6). Для решения задачи могут быть использованы интенсивно развивающиеся в настоящее время в рамках HSCI эволюционные биоинспирированные «роевые» процедуры оптимизации [20], среди которых в качестве одних из наиболее быстродействующих можно отметить, так называемые, алгоритмы кошачьих стай [21,22]. Заметим, что именно кошачьи стаи с успехом были использованы для решения задач четкой кластеризации в рамках процедуры K-средних [23,24], порождаемой целевой функцией (1) при $\beta \rightarrow 1, U_j(k) = \{0,1\}$. В рамках этого подхода предполагается [23], что каждый центроид c_j представлен одной из кошек стаи, а конечное

решение определяется кошками, обеспечивающими минимум целевой функции $E(c_j)$ (5) или (6).

В рамках стандартного «кошачьего» алгоритма [21, 22] предполагается, что каждая кошка cat_p стаи, состоящей из Q особей ($p=1,2,\dots,Q$), может находиться в одном из двух состояний: режиме поиска (Seeking Mode - SM) и режиме погони (Tracing Mode - TM). При этом режим поиска связан с медленными движениями с незначительной амплитудой около исходной позиции (сканирование пространства в окрестности текущей позиции), а режим погони определяется быстрыми скачками с большой амплитудой и позволяет вывести кошку cat_p из локального экстремума, если она туда попала. Сочетание локального сканирования и резких изменений текущего состояния позволяет с большей вероятностью отыскать глобальный экстремум по сравнению с традиционными методами многоэкстремальной оптимизации.

Процесс отыскания экстремума с помощью кошачьей стаи может быть реализован в виде следующей последовательности шагов:

Шаг CS 1: создать стаю из Q кошек в виде набора n -мерных векторов $c_p(0)$, случайным образом распределенных на множестве допустимых значений аргументов R_c^n , т.е. $c_p(0) \in R_c^n \subset R^n$; оценить значение оптимизируемой функции (фитнесс - функции) $E(c_p(0))$ во всех Q точках, при этом предполагается, что целью оптимизации является отыскание глобального минимума $E(c)$.

Шаг CS 2: ввести параметр состояния SPC (self position consideration), принимающий два значения 1 или 0; случайным образом разделить стаю на две группы: кошки в поиске (SPC=1) и кошки в погоне (SPC=0).

Шаг CS 3: если SPC=1, запустить соответствующую группу кошек в поиск, оставшихся кошек с SPC=0 запустить в режим погони.

Шаг CS 4: оценить значение фитнес - функции и сохранить новые состояния $c_p(1)$, соответствующие наименьшим значениям $E(c_p(1))$.

Шаг CS 5: вернуться к шагу CS1 с обновленной стаей $c_p(1)$, $p=1,2,\dots,Q$.

Режимы поиска и погони могут быть реализованы параллельно и также состоять из последовательности итераций. При этом режим поиска кошачьей стаи соответствует процессу локального поиска в задаче оптимизации. Режим поиска определяется тремя основными факторами: объемом памяти поиска (seeking memory pool - SMP), который определяет количество создаваемых копий каждой кошки cat_p , шагом изменения по каждой координате пространства R_c^n (seeking range of the selected dimension - SRD) и изменяемых координат (counts of dimension to change - CDC). Собственно

режим поиска может быть реализован в виде следующей последовательности шагов:

Шаг SM 1: если SPC = 1, создать C ($C=SMP$) копий cat_p .

Шаг SM2: в соответствии с принятым CDC изменить состояние cat_p .

Шаг SM3: оценить значения оптимизируемой фитнес-функции для каждого измененного состояния cat_p .

Шаг SM 4: ввести вероятности выбора каждого изменяемого состояния

$$P_p = \frac{E(c_p(\tau)) - E_{\min}(c_p(\tau))}{E_{\max}(c_p(\tau)) - E_{\min}(c_p(\tau))}, \tau = 1, 2, \dots, T$$

и кошку с максимальным значением P_p исключить из дальнейшего рассмотрения. Кошка с $P_p = 0$ является «наилучшей» копией cat_p , поскольку ей соответствует наименьшее значение оптимизируемой функции $E_{\min}(c_p(\tau))$.

Режим погони соответствует процессу глобального поиска, позволяющего «проскакать» локальные экстремумы оптимизируемой функции, и также может быть реализован в виде последовательности шагов:

Шаг TM 1: если SPC = 0, для группы кошек в погоне рассчитать для каждой cat_p скорости движения по каждой координате с помощью рекуррентного выражения

$$v_{pi}(\tau+1) = v_{pi}(\tau) + r(\tau)\eta_{TM}(c_{best,i}(\tau) - c_{pi}(\tau)),$$

где $v_{pi}(\tau)$ — скорость движения i -й кошки по i -й координате на τ -й итерации погони, $0 < r(\tau) < 1$ — случайный параметр погони, η_{TM} — постоянный шаг погони, $c_{best,i}(\tau)$ — наилучшее решение задачи оптимизации, полученное на τ -й итерации.

Шаг TM 2: ввести предельно возможные значения скоростей v_{\min} и v_{\max} , для каждой кошки проверить условие

$$v_{\min} < v_{pi}(\tau+1) < v_{\max}$$

и если оно нарушается, положить $v_{pi}(\tau+1)$ равным соответствующему значению v_{\min} или v_{\max} .

Шаг TM 3: изменить положение каждой кошки в погоне согласно соотношению

$$c_{pi}(\tau+1) = c_{pi}(\tau) + v_{pi}(\tau).$$

Шаг TM 4: проверить, принадлежит ли $c_p(\tau+1) R_c^n$.

Можно заметить, что рассмотренный алгоритм поиска реализует по сути покоординатный спуск (метод Гаусса - Зайделя), требующий многократного оценивания значений оптимизируемой функции и характеризующийся низкой скоростью сходимости. В режиме погони реализуется градиентный поиск с большим шагом, что в общем случае не гарантирует отыскание глобального экстремума. В связи с этим представляется целесообразным

модернизировать процедуру оптимизации на основе кошачьих стай путем ее рандомизации на основе случайного поиска [9-11], обладающего целым рядом преимуществ перед детерминированными процедурами поиска экстремума.

2. Рандомизированный алгоритм оптимизации на основе кошачьих стай в задаче нечеткой кластеризации

Поскольку режим поиска SM есть по сути процесс локальной оптимизации, движение каждой из кошек cat_p с $SPC=1$ целесообразно организовать в антиградиентном направлении согласно стандартной рекуррентной градиентной процедуре

$$A_p(\tau+1) = A_p(\tau) - \eta_{SM} \hat{\nabla} E(A_p(\tau)), \quad (7)$$

где $\hat{\nabla} E(A_p(\tau))$ — оценка градиента оптимизируемой функции в точке $A_p(\tau)$, η_{SM} — шаг поиска в пространстве R_c^n .

Составляющие градиента $\hat{\nabla} E(A_p(\tau))$, являющиеся частными производными $\frac{\partial E(A_p(\tau))}{\partial A_{pi}}$, могут быть оценены путем измерения оптимизируемой функции в пробных состояниях в окрестности точки $A_p(\tau)$. Наиболее простым с вычислительной точки зрения является поиск с центральной пробой [9], при этом производится оценка оптимизируемой функции в $(n+1)$ -й точке ($CDC=n$): $A_p(\tau)$, $A_p(\tau) + \eta_{SRD}e_1$, $A_p(\tau) + \eta_{SRD}e_2, \dots, A_p(\tau) + \eta_{SRD}e_n$, где e_i — координатные орты, η_{SRD} — величина пробного шага, определяемая принятым значением SRD.

Определив $n+1$ значение функции $E(A_p(\tau))$, $E(A_p(\tau) + \eta_{SRD}e_2), \dots, E(A_p(\tau) + \eta_{SRD}e_n)$, вместо градиента

$$\nabla E(A_p(\tau)) = \left(\frac{\partial E(A_p(\tau))}{\partial A_{p1}}, \frac{\partial E(A_p(\tau))}{\partial A_{p2}}, \dots, \frac{\partial E(A_p(\tau))}{\partial A_{pn}} \right)^T,$$

можно ввести его оценку $\hat{\nabla} E(A_p(\tau))$ с компонентами

$$\frac{\partial \hat{E}(A_p(\tau))}{\partial A_{pi}} = \frac{1}{\eta_{SRD}} (E(A_p(\tau) + \eta_{SRD}e_i) - E(A_p(\tau))), i=1, 2, \dots, n.$$

Реализовав далее шаг в пространстве R_c^n в соответствии с (7), приходим к новому состоянию cat_p в режиме поиска с координатами

$$\begin{cases} c_{p1}(\tau+1) = c_{p1}(\tau) - \frac{\eta_{SM}}{\eta_{SRD}} (E(c_p(\tau) + \eta_{SRD}e_1) - E(c_p(\tau))), \\ c_{p2}(\tau+1) = c_{p2}(\tau) - \frac{\eta_{SM}}{\eta_{SRD}} (E(c_p(\tau) + \eta_{SRD}e_2) - E(c_p(\tau))), \\ \dots \\ c_{pn}(\tau+1) = c_{pn}(\tau) - \frac{\eta_{SM}}{\eta_{SRD}} (E(c_p(\tau) + \eta_{SRD}e_n) - E(c_p(\tau))). \end{cases}$$

Можно заметить, что в случае

$$E(c_p(\tau+1)) < E(c_p(\tau)),$$

cat_p приближается к локальному минимуму, т.е. улучшает свое состояние и может далее оставаться в режиме поиска. Если же

$$E(c_p(\tau+1)) \geq E(c_p(\tau)),$$

cat_p находится в окрестности локального минимума, вывести из которого ее можно, переведя в режим погони.

В качестве недостатка этой процедуры оптимизации можно отметить фиксированное значение $CDC=n$, что требует поочередного изменения всех координат cat_p в пространстве R_c^n . Расширить возможности процесса поиска можно, обратившись к рандомизированным процедурам, простейшей из которых является чисто случайная оценка направления спуска, смысл которой состоит в том, что из состояния $c_p(\tau)$ делается случайная проба $c_p(\tau) + \eta_{SRD}\Xi$, где $\Xi = (\xi_1, \xi_2, \dots, \xi_n)^T$ — единичный случайный вектор, равномерно распределенный в пространстве R_c^n . В случае, если $c_p(\tau) + \eta_{SRD}\Xi < E(c_p(\tau))$, делается рабочий шаг поиска

$$c_p(\tau+1) = c_p(\tau) - \eta_{SM}\Xi \quad (8)$$

(при этом можно принять $\eta_{SRD} = \eta_{SM}$), в противном случае проба признается неудачной и реализуется попытка с новым вектором Ξ .

Обобщением этой процедуры является оценка направления поиска по наилучшей из нескольких случайных проб. При этом из исходного состояния $c_p(\tau)$ делается несколько случайных проб оптимизируемой функции $c_p(\tau) + \eta_{SRD}\Xi_l$ в случайных направлениях $\Xi_l (l=1, 2, \dots, n, \dots, L)$, при этом фактор CDC может превышать значение n . За направление спуска выбирается то направление Ξ^* , которое обеспечило наименьшее значение функции $E(c_p)$, т.е. cat_p переводится в новое состояние согласно выражению

$$c_p(\tau+1) = c_p(\tau) + \eta_{SRD}\Xi^*. \quad (9)$$

Заметим также, что при $L=1$, процедуры (8) и (9) совпадают.

Объединяя описанные процедуры поиска, можно ввести в рассмотрение поиск на основе статистического градиента. В этом случае за оценку градиента принимается средневзвешенное из L случайных направлений, каждое из которых берется с весом, соответствующим вариации $E(c_p)$ вдоль этого направления:

$$\hat{\nabla} E(c_p(\tau)) = - \frac{\sum_{l=1}^L \Xi_l (E(c_p(\tau) + \eta_{SRD}\Xi_l) - E(c_p(\tau)))}{\left\| \sum_{l=1}^L \Xi_l (E(c_p(\tau) + \eta_{SRD}\Xi_l) - E(c_p(\tau))) \right\|}. \quad (10)$$

Подставляя далее (10) в (9), получаем процедуру градиентного спуска в направлении минимума оптимизируемой функции.

Таким образом, все кошки с $SPC=1$ смещаются в направлении локальных минимумов оптимизируемой функции.

Режим погони ТМ в отличие от локального режима поиска SM обеспечивает общей процедуре оптимизации на основе CS глобальные свойства, позволяющие не застревать ей в локальных экстремумах. Понятно, что кроме рассматриваемых процедур существуют и другие алгоритмы, обладающие требуемыми свойствами.

Одним из таких наиболее эффективных численно простых алгоритмов является метод тяжелого шарика, опирающийся на аналогию движения тяжелого тела по искривленной поверхности с учетом сил тяжести и трения. При этом в силу инерции шарик-кошка «проскакивает» локальные экстремумы, а в силу трения движение должно остановиться в глобальном экстремуме.

Данный алгоритм для кошек в режиме погони ($SPC=0$) может быть записан в виде [25]

$$c_p(\tau+1) = c_p(\tau) - \alpha(c_p(\tau) - c_p(\tau-1)) - \eta_{TM} \hat{\nabla} E(c_p(\tau)), \quad (11)$$

где α — параметр, определяющий инерционные свойства процесса погони. При $\alpha = 0$ (11) полностью совпадает с (7), отличаясь только шагом η_S . При $\alpha = 1$ процесс погони становится незатухающим, поэтому этот параметр выбирается в интервале $0 < \alpha < 1$, при этом чем ближе α к единице, тем сильнее проявляются инерционные свойства, однако процесс слабо затухает в окрестности экстремума. В связи с этим целесообразно каждой кошке с $SPC=0$ назначить разные значения параметра α .

Заметим также, что в процедуру (11) может быть введена случайная компонента, вводящая дополнительное «рыскание» в процесс погони, улучшающее глобальные свойства алгоритма. При этом (11) модифицируется к виду

$$c_p(\tau+1) = c_p(\tau) - \alpha(c_p(\tau) - c_p(\tau-1)) - \eta_{TM} \hat{\nabla} E(c_p(\tau)) + \eta_{SRD} \Xi,$$

т.е. cat_p одновременно находится и в режиме погони и в режиме поиска-сканирования пространства R_c^n .

3. Экспериментальные исследования

Эксперименты по кластеризации FCMCSO проводились с использованием четырех наборов данных: Iris, Cancer, Wine and Glass. Каждый из наборов данных имеет ряд параметров, представленных в таблице 1.

Таблица 1

Характеристические параметры выборок

Название выборки	Число классов	Число атрибутов	Кол-во набл.
Iris	3	4	150
Cancer	2	9	683
Wine	3	13	178
Glass	6	8	214

Таблица 2

Параметры алгоритма нечеткой кластеризации на основе оптимизации стаи кошек (FCMCSO)

Параметры	Значения
Поиск диапазона выбранного измерения (SRD)	Случайно [0,1]
Поиск пула памяти(SMP)	5
Размер популяции	Количество кластеров
r_1	Случайное значение в диапазоне [0,1]
c_1	Константа
Самооценка позиции (SPC)	Случайно в диапазоне [0,1]
Количество итераций	Manually

Таблица 3

Сравнительные результаты временной обработки алгоритмов кластеризации таких, как FCM, PSO, GSA, CSO и FCMCSO

Название выборки	FCM	PSO	GSA	CSO	FCM CSO
Iris	0.008	0.020	0.022	0.043	0.012
Cancer	0.009	0.138	0.204	0.026	0.007
Wine	0.009	0.282	0.098	0.076	0.013
Glass	0.010	0.431	0.431	0.021	0.020

Таблица 4

Результаты кластеризации CSO и FCMCSO с различным количеством итераций (средняя ошибка в %)

Название выборки	Количество итераций CSO			Количество итераций FCMCSO		
	50	100	150	50	100	150
Iris	23.34	20.84	21.67	17.55	14.78	16.46
Cancer	40,23	40,55	41,47	38.89	39.22	39.15
Wine	24,55	21,44	22,20	18.43	17.37	16.32
Glass	56.34	56.48	55.67	51.63	51.7	49.79

Выводы

Рассмотрена задача нечеткой кластеризации массива наблюдений на основе нечеткого вероятностного подхода, в основу которого положен алгоритм нечетких С-средних, переформулированный в задачу безусловной многоэкстремальной оптимизации. Для решения задачи использована рандомизированная модификация алгоритма оптимизации кошачьих стай, отличающаяся от известной введением в процессы поиска и погони элементов случайного поиска. Использование рандомизированной модификации позволило улучшить точность определения направления движения в режиме кошачьего поиска и улучшить глобальные свойства процедуры в режиме погони, что, в свою очередь, улучшает качество решения собственно задачи кластеризации. Использование эволюционного метода оптимизации кошачьих стай позволило упростить численную реализацию процесса кластеризации, сократить необходимый объем стай и исключить использование, так называемых, копий каждой кошки.

Список литературы

- [1] Gan G., Ma Ch., Wu J. Data Clustering: Theory, Algorithms and Applications. – Philadelphia, Pennsylvania: SIAM, 2007. – 455 p.
- [2] Abonyi J., Feil B. Cluster Analysis for Data Mining and System Identification. – Basel: Birkhauser, 2007. – 303p.
- [3] Xu R., Wunsch D.C. II. Clustering– Hoboken, N.J.: John Wiley & Sons, Inc., 2009. – 341 p.
- [4] Bezdek J.C. Pattern Recognition with Fuzzy Objective Function Algorithms.–N.Y.:Plenum Press, 1981.–272p.
- [5] Höppner F., Klawonn F., Kruse R., Runkler T. Fuzzy Clustering Analysis: Methods for Classification, Data Analysis and Image Recognition.–Chichester: John Wiley & Sons, 1999. – 289 p.
- [6] Bezdek J.C., Keller J., Krishnapuram R., Pal N.R. Fuzzy Models and Algorithms for Pattern Recognition and Image Processing. – N.Y.: Springer Science + Business Media, Inc., 2015. – 776 p.
- [7] Bezdek J.C. A convergence theorem for the fuzzy ISO-DATA clustering algorithms. – IEEE Trans. Pattern Anal. Mach. Intell. – 1980 – 2. – P. 1-8.
- [8] Belacel N., Hansen P., Mladenovic N. Fuzzy J-Means: a new heuristic for fuzzy clustering – Pattern Recognition. – 2007. – 35. – P. 2193-2200.
- [9] Растрингин Л.А. Статистические методы поиска. – Москва: Наука, 1968. – 376 с.
- [10] Растрингин Л.А., Рипа К.К. Автоматическая теория случайного поиска. – Рига: Зинатне, 1973. – 343 с.
- [11] Растрингин Л.А. Случайный поиск в процессах адаптации. – Рига: Зинатне, 1973. – 132 с.
- [12] Kirkpatrick S.C.G., Vecci M. Optimization by simulated annealing – Science – 1983 – 220 – P. 49-58.
- [13] L.Rutkowski. Computational Intelligence. Methods and Techniques. Berlin-Heidelberg: Springer-Verlag, 2008. – 514 p.
- [14] Mumford C. L., Jain L.C. Computational Intelligence. Berlin: Springer-Verlag, 2009. – 729 p.
- [15] Kroll A. Computational Intelligence. Eine Einführung in Probleme, Methoden und technishe Anwendungen – München: Oldenbourg Verlag, 2013. – 428 S.
- [16] Kruse R., Borgelt C., Klawonn F., Moawes C., Steinbrecher M., Held P. Computational Intelligence. A Methodological Introduction. – Berlin: Springer-Verlag, 2013. – 488 p.
- [17] Kacprzyk J., Pedrycz W. Springer Handbook of Computational Intelligence. – Berlin Heidelberg: Springer-Verlag, 2015. – 1634 p.
- [18] R.J. Hathaway, J.C Bezdek. Optimization of clustering criteria by reformulation. – IEEE Trans. Fuzzy Systems.- 1995. – 3. – P. 241-245.
- [19] Pal N.R., Bezdek J.C.,Hathaway R.J. Sequential competitive learning algorithm. – Neural Networks. – 1996. – 9. – № 5. – P. 787-796.
- [20] Grosan C., Abraham A., Chis M. Swarm intelligence in Data Mining – Studies in Computational Intelligence. – 2006. – 34. – P. 1-20.
- [21] Chu S.-C., Tsai P.-W., Pan J.S. Cat swarm optimization // Lecture Notes in Artificial Intelligence. – 4099. – Berlin Heidelberg: Springer-Verlag, 2006. – P. 854-858.
- [22] Chu S.-C., Tsai P.-W. Computational Intelligence based on the behavior of cats // Int. J. of Innovative Computing, Information, and Control. – 2007. – 3. – №1. – P.163-173.
- [23] B. Santosa and M. K. Ningrum, Cat Swarm Optimization for Clustering, Soft Computing and Pattern Recognition, International Conference of (SOCPAR), Malacca, Malaysia, 2009, P. 54-59.
- [24] Liu Y., Wu, Shen Y. Cat swarm optimization clustering (KSACSOC): A cat swarm optimization clustering algorithm. – Sci. Reseach and Essays – 2012 – 7. №49. – P. 4176-4185.
- [25] Бодянский Е.В, Шафроненко А.Ю. Рандомизированная модификация метода оптимизации на основе кошачьих стай. – Системи обробки інформації. – 2018. – № 1(152). – С. 142-147.

Поступила в редколлегию 12.09.2018

УДК 004.032.26

Е.В.Бодянский¹, К.Э.Петров², А.А.Дейнеко³¹ХНУРЭ, г.Харьков, Украина, yevgeniy.bodyanskiy@nure.ua²ХНУРЭ, г.Харьков, Украина, kostiantyn.petrov@nure.ua³ХНУРЭ, г.Харьков, Украина, anastasiia.deineko@nure.ua

ЭВОЛЮЦИОНИРУЮЩАЯ РАДИАЛЬНО-БАЗИСНАЯ НЕЙРОННАЯ СЕТЬ И ЭКСТРЕМАЛЬНОЕ ОБУЧЕНИЕ ЕЕ ПАРАМЕТРОВ

Предложен подход к формированию архитектуры и обучению эволюционирующей радиально-базисной нейронной сети (RBFN), которая обрабатывает данные, поступающие в режиме online. Он состоит из двух этапов. Первый из них базируется на использовании самоорганизующейся карты Т. Кохонена (SOM), которая отвечает за управление количеством нейронов RBFN и настройку расположения центров ядерных функций в режиме самообучения. Второй этап отвечает за ее обучение с применением метода экстремального обучения (ELM). Данный подход позволяет избежать «проклятия размерности» при формировании RBFN, а также существенно повысить скорость ее обучения.

ЭВОЛЮЦИОНИРУЮЩАЯ СИСТЕМА, САМООРГАНИЗУЮЩАЯСЯ КАРТА Т. КОХОНЕНА, ЯДЕРНАЯ ФУНКЦИЯ, МЕТОД ЭКСТРЕМАЛЬНОГО ОБУЧЕНИЯ

Є.В. Бодянський, К.Е. Петров, А.А. Дейнеко. Еволюціонуюча радіально-базисна нейронна мережа та екстремальне навчання її параметрів. Запропоновано підхід до формування архітектури і навчання еволюціонуючої радіально-базисної нейронної мережі (RBFN), яка обробляє дані, що надходять в режимі online. Він складається з двох етапів. Перший з них базується на використанні самоорганізаційної карти Т. Кохонена (SOM), яка відповідає за управління кількістю нейронів RBFN і налаштування розташування центрів ядерних функцій в режимі самонавчання. Другий етап відповідає за її навчання із застосуванням методу екстремального навчання (ELM). Даний підхід дозволяє уникнути «прокляття розмірності» при формуванні RBFN, а також істотно підвищити швидкість її навчання.

ЕВОЛЮЦІОНУЮЧА СИСТЕМА, САМООРГАНІЗАЦІЙНА КАРТА Т. КОХОНЕНА, ЯДЕРНА ФУНКЦІЯ, МЕТОД ЕКСТРЕМАЛЬНОГО НАВЧАННЯ

Y.V. Bodyansky, K.E. Petrov, A.A. Deineko. *Evolving radial basis neural network and extreme learning of its parameters.* In the paper proposes an approach to the formation of the architecture and training of the evolving radial-basis neural network (RBFN), which processes the data in sequential online mode. It consists of two stages. The first one is based on the use of the T. Kohonen self-organizing map (SOM), which is responsible for managing the number of RBFN neurons and setting the location of the centers of kernel functions in self-learning mode. The second stage is responsible for its training using the method of extreme learning (ELM). This approach allows to avoid the “curse of dimensionality” in the formation of RBFN and significantly increase the speed of its learning.

EVOLVING SYSTEM, T. KOHONEN SELF-ORGANIZING MAP, KERNEL FUNCTION, EXTREME LEARNING MACHINE

Введение

В настоящее время искусственные нейронные сети (ИНС) широко используются для эффективного решения различного рода задач интеллектуального анализа данных, таких как прогнозирование, классификация, кластеризация и т. п. И все это благодаря своим универсальным аппроксимирующим свойствам и способности обучаться на основе экспериментальных данных поступающих от объекта исследования.

Особый интерес представляет разработка систем, которые обрабатывают поступающие последовательности данных в реальном масштабе времени (в режиме online). Эти задачи относятся к области динамического анализа данных (Dynamic Data Mining и Data Stream Mining) [1]. Особенности решения такого рода задач являются высокие требования к скорости обработки данных, что непосредственно связано со скоростью обучения, используемых для их решения

ИНС. В этих условиях существенным преимуществом обладают нейронные сети, чей выходной сигнал линейно зависит от настраиваемых параметров. И в первую очередь к таким сетям можно отнести радиально-базисные нейронные сети (RBFN) [2], для обучения которых традиционно используют быстродействующие гауссовско-ньютоновские (рекуррентный МНК и его модификации или фильтр Калмана), квазиньютоновские (алгоритм Левенберга-Марквардта), градиентные (алгоритмы Качмажа-Уидроу-Хоффа и Гудвина-Рэмеджа-Кэйнеса) или экстремальные (метод ELM) процедуры.

Главной проблемой при синтезе RBFN является определение необходимого количества нейронов, формирующих архитектуру сети, поскольку недостаточное их число «ухудшает» ее аппроксимирующие свойства, а их излишек ведет к возникновению эффекта «проклятия размерности» (экспоненциальному росту числа нейронов с ростом

размерности входного пространства). Решение этой проблемы возможно на основе построения эволюционирующих систем вычислительного интеллекта [3].

Таким образом, актуальным и целесообразным представляется разработка подхода к обучению параметров и формированию архитектуры RBFN, при котором структура нейронной сети может изменяться в процессе обучения, т. е. количество ее нейронов может как увеличиваться (growing mode), так и уменьшаться (pruning mode) в online-режиме по мере поступления на ее вход новой информации.

1. Содержательная постановка задачи

Реализация этого подхода связана с решением задачи построения гибридной эволюционирующей искусственной нейронной сети, которая базируется на RBFN с переменным количеством нейронов и самоорганизующейся карте Т. Кохонена (SOM) [4], отвечающей за управление количеством нейронов RBFN и настройку расположения центров ядерных функций в режиме самообучения.

Процесс функционирования этой сети может быть упрощенно описан так.

При поступлении первого наблюдения оно подается на вход RBFN, где формируется первый нейрон по принципу «нейроны в точках данных» [4], т. е. практически мгновенно. При поступлении последующих наблюдений они, вначале поступают в SOM (обрабатываются в SOM), где происходит сравнение с уже существующими центроидами (ядрами), а затем, если совпадений не обнаружилось формируется новый центроид (ядро) и соответственно новый нейрон в RBFN.

Таким образом на первом этапе формируется структура RBFN на основе SOM, а на втором — производится ее обучение с использованием метода экстремального обучения (ELM) [5].

Рассмотрим более подробно реализацию этих этапов.

2. Управление количеством нейронов в RBFN (формирование структуры нейронной сети)

Исходной информацией для реализации первого этапа является выборка (возможно растущая) наблюдений $x(1), x(2), \dots, x(k), \dots, x(N), \dots$, в которой $x(k) = [x_1(k) \ x_2(k) \ \dots \ x_n(k)]$, $x(k) \in R^n$ и, соответственно, $y(1), y(2), \dots, y(k), \dots, y(N), \dots$, где $y(k) = [y_1(k) \ y_2(k) \ \dots \ y_m(k)]$, $y(k) \in R^m$.

Как было отмечено выше, наиболее сложным моментом формирования структуры RBFN является определение необходимого количества нейронов сети по мере поступления (online) на ее вход новых наблюдений.

Для решения этой задачи воспользуемся идеями, лежащими в основе эволюционирующих систем вычислительного интеллекта, адаптированных к обработке информации в режиме реального масштаба времени.

В рамках развиваемого подхода предлагается следующий алгоритм управления количеством нейронов в RBFN [6].

Шаг 0. Задаются: Δ — порог неразличимости векторов центров c_i активационных ядерных функций $\phi_i(x, c_i, \sigma_i)$, $i = \overline{1, h}$ ($h \leq H$); H — максимально допустимое количество нейронов в RBFN ($H \gg n$ и $H \leq N$) и параметр ширины σ_i , определяющего локальную область входного пространства, на которую реагирует функция $\phi_i(x, c_i, \sigma_i)$ (например, $\sigma_i^2 = 0,33^2 \ \forall i = \overline{1, h}$ при $-1 \leq x_j(k) \leq 1$, $j = \overline{1, n}$, $k = \overline{1, N}$).

Шаг 1. При поступлении наблюдения $x(1)$ формируется первый центр $c_1 = x(1)$ и сама активационная ядерная функция.

Без потери общности, для определенности, в качестве активационной будем рассматривать гауссову функцию вида:

$$\phi_i(x, c_i, \sigma_i) = e^{-\frac{\|x - c_i\|^2}{2\sigma_i^2}}. \quad (1)$$

В итоге формируется функция

$$\phi_1(x, c_1, \sigma_1) = e^{-\frac{\|x - x(1)\|^2}{2\sigma_1^2}}.$$

Таким образом, уже с поступления первого наблюдения начинается обучение сети по принципу «нейроны в точках данных» [4].

Шаг 2. При поступлении наблюдения $x(2)$:

1) если выполняется условие

$$\|x(2) - c_1\| \leq \Delta, \quad (2)$$

то наблюдение $x(2)$ не формирует новый центр и исключается из обучающей выборки;

2) если

$$\Delta < \|x(2) - c_1\| \leq 2\Delta, \quad (3)$$

то c_1 корректируется согласно WTA-правилу самообучения Т. Кохонена [4] следующим образом:

$$c_1 := c_1 + \eta(2)(x(2) - c_1),$$

где $0 < \eta(k) < 1$ — параметр шага настройки;

3) если же

$$2\Delta < \|x(2) - c_1\|, \quad (4)$$

то формируется новая радиально-базисная функция $\phi_2(x, c_2, \sigma_2)$ с центром $c_2 = x(2)$:

$$\phi_2(x, c_2, \sigma_2) = e^{-\frac{\|x - c_2\|^2}{2\sigma_2^2}} \quad \text{или} \quad \phi_2(x, c_2, \sigma_2) = e^{-\frac{\|x - x(2)\|^2}{2\sigma_2^2}}.$$

Шаг N. Пусть к моменту поступления N -го входного образца $x(N)$ сформировано $h < H$

активационных функций $\varphi_i(x, c_i, \sigma_i)$ с центрами c_i , $i = \overline{1, h}$.

При поступлении на вход наблюдения $x(N)$ вначале определяется центр $c^0 = c_i$, ближайший к $x(N)$, т. е. c_i , для которого расстояние $\|x(N) - c_i\|$, $i = \overline{1, h}$ минимально среди всех h имеющихся центров. Далее проверяются условия:

$$\begin{aligned} \|x(N) - c^0\| &\leq \Delta; \\ \Delta < \|x(N) - c^0\| &\leq 2\Delta; \\ 2\Delta < \|x(N) - c^0\|. \end{aligned}$$

В результате $x(N)$ либо игнорируется, либо происходит коррекция центра c^0 , либо создается новая радиально-базисная функция $\varphi_{h+1}(x, c_{h+1}, \sigma_{h+1})$.

Процесс наращивания количества R-нейронов скрытого слоя RBFN может быть продолжен до достижения их количества равного H .

Рекомендованное количество ядер (скрытых нейронов) H для RBFN можно определить из эмпирической формулы:

$$H = \text{floor}[n + 0.618(n - m)],$$

где floor — наибольшее целое число, которое меньше или равно данному; n — количество нейронов входного слоя; m — количество нейронов выходного слоя.

Если в сети сформировано $h = H$ радиально-базисных функций, эволюционирующая RBFN переходит из режима обучения, основанного на памяти по принципу «нейроны в точках данных» в режим обучения, основанного на методе экстремального обучения (ELM).

При этом в процессе корректировки центров сформированных радиально-базисных функций постоянно контролируется расстояние между ними. Если выясняется, что

$$\|c_l - c_t\| \leq \Delta, \quad \forall l = \overline{1, h}, t = \overline{1, h}, l \neq t, h \leq H,$$

то соответствующий R-нейрон с активационной функцией $\varphi_i(x, c_i, \sigma_i)$ исключается из сети.

Таким образом, при формировании данной нейронной сети реализуются принципы эволюционного обучения П. Ангелова — Н. Касабова [3] и самообучения Т. Кохонена [4].

Далее подробно рассмотрим второй этап, связанный с экстремальным обучением сформированной на первом этапе RBFN.

3. Экстремальное обучение RBFN

На рис. 1 приведена стандартная схема радиально-базисной сети с n входами и m — выходами, содержащая h — нейронов (ядер) скрытого слоя, которая реализует нелинейное преобразование вида

$$y_j = F_j(x) = w_{j0} + \sum_{i=1}^h w_{ji} \varphi_i(x, c_i, \sigma_i), \quad j = \overline{1, m},$$

где $\varphi_i(x, c_i, \sigma_i) = \Phi_i(\|x - c_i\|, \sigma_i)$ — радиально-базисная симметричная ядерная функция (выход i -го нейрона), определяющая свойства сети и зависящая от расстояния $\|x - c_i\|$ между входом, который задан матрицей $x = [x_1 \ x_2 \ \dots \ x_n]$ и центром c_i (центр i -го нейрона), а так же параметра ширины σ_i , выделяющего локальную область входного пространства, на которую реагирует эта ядерная функция; w_{ji} — весовой коэффициент связи i -ого ядра (нейрона скрытого слоя) с j -ым выходом (выходным нейроном) сети.

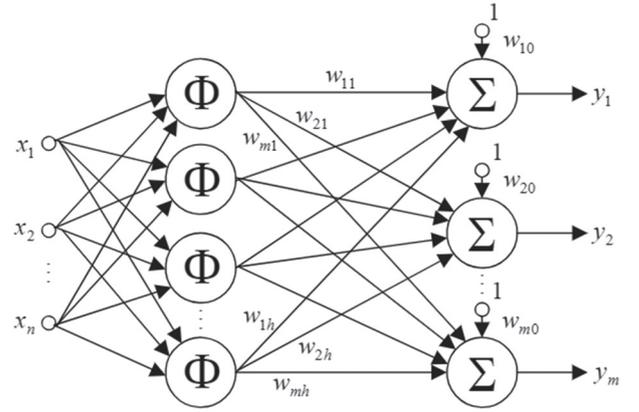


Рис. 1. Радиально-базисная нейронная сеть

Предполагается, что $\varphi_i(x, c_i, \sigma_i)$ является нелинейной, ограниченной, интегрируемой и всюду непрерывной функцией. Такие ядерные функции включают в себя, в том числе, широко используемую гауссову функцию.

Как было отмечено выше, в качестве активационной будем использовать гауссову функцию вида (1).

4. Расширение метода экстремального обучения (ELM) для случая RBFN

ELM может быть линейно расширен на случай SLFN (Single hidden layer feedforward neural network) с ядерными RBF [7].

Пусть заданы матрицы входных признаков (экземпляров)

$$X = \begin{bmatrix} x(1) \\ \dots \\ x(N) \end{bmatrix} = \begin{bmatrix} x_1(1) & x_2(1) & \dots & x_n(1) \\ \dots & \dots & \dots & \dots \\ x_1(N) & x_2(N) & \dots & x_n(N) \end{bmatrix}_{N \times n}$$

и выходных классов (результатов)

$$Y = \begin{bmatrix} y(1) \\ \dots \\ y(N) \end{bmatrix} = \begin{bmatrix} y_1(1) & y_2(1) & \dots & y_m(1) \\ \dots & \dots & \dots & \dots \\ y_1(N) & y_2(N) & \dots & y_m(N) \end{bmatrix}_{N \times m}.$$

Обозначим через w_{ji} — весовой коэффициент связи i -ого ядра (нейрона скрытого слоя) с j -ым выходом (выходным нейроном) сети.

Тогда математическая модель RBFN с h ядрами (см. рис. 1) может быть записана так:

$$\sum_{i=0}^h w_i \varphi_i(x(k), c_i, \sigma_i) = o(k), \quad k = \overline{1, N},$$

где

$$w_0 = [1 \ 1 \dots 1], \quad w_i = [w_{1i} \ w_{2i} \dots w_{mi}],$$

$$\varphi_0(x(k), c_0, \sigma_0) = 1, \quad \varphi_i(x(k), c_i, \sigma_i) = \Phi_i(\|x(k) - c_i\|, \sigma_i),$$

$$x(k) = [x_1(k) \ x_2(k) \dots x_n(k)]$$

и, соответственно $o(k) = [o_1(k) \ o_2(k) \dots o_m(k)]$.

Подобно случаю SLFN, стандартные RBFN с h ядрами могут аппроксимировать эти N данных с нулевой ошибкой. Это означает, что

$$\sum_{k=1}^N \|o(k) - y(k)\| = 0,$$

где $y(k) = [y_1(k) \ y_2(k) \dots y_m(k)]$. Т. е. существуют w_i , c_i и σ_i такие, что

$$\sum_{i=0}^h w_i \varphi_i(x(k), c_i, \sigma_i) = y(k), \quad k = \overline{1, N}. \quad (5)$$

Из (5) следует, что для нахождения значений матрицы синаптических весов W необходимо решить систему линейных уравнений, которая в матричной форме может быть записана так:

$$\Phi W = Y, \quad (6)$$

где

$$\Phi = \begin{bmatrix} 1 & \varphi_1(x(1), c_1, \sigma_1) & \varphi_2(x(1), c_2, \sigma_2) & \dots & \varphi_h(x(1), c_h, \sigma_h) \\ 1 & \varphi_1(x(2), c_1, \sigma_1) & \varphi_2(x(2), c_2, \sigma_2) & \dots & \varphi_h(x(2), c_h, \sigma_h) \\ \dots & \dots & \dots & \dots & \dots \\ 1 & \varphi_1(x(N), c_1, \sigma_1) & \varphi_2(x(N), c_2, \sigma_2) & \dots & \varphi_h(x(N), c_h, \sigma_h) \end{bmatrix}_{N \times (h+1)}$$

$$W = \begin{bmatrix} w_0 \\ w_1 \\ \dots \\ w_h \end{bmatrix} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ w_{11} & w_{21} & \dots & w_{m1} \\ \dots & \dots & \dots & \dots \\ w_{1h} & w_{2h} & \dots & w_{mh} \end{bmatrix}_{(h+1) \times m}$$

матрица выходных весов нейронов скрытого слоя сети;

$$Y = \begin{bmatrix} y(1) \\ \dots \\ y(N) \end{bmatrix} = \begin{bmatrix} y_1(1) & y_2(1) & \dots & y_m(1) \\ \dots & \dots & \dots & \dots \\ y_1(N) & y_2(N) & \dots & y_m(N) \end{bmatrix}_{N \times m}.$$

Подобно SLFN, Φ называется выходной матрицей скрытого слоя сети RBFN где i -й столбец Φ , $i = \overline{1, h}$, является выходом i -го нейрона (ядра) относительно входов $x(1), x(2), \dots, x(N)$.

Поскольку на практике сеть обучается с использованием конечных обучающих выборок $(x(k), y(k))$, $k = \overline{1, N}$, отыскание $\min_W \|f_h - f\|$, эквивалентно нахождению $\min_W \|\Phi W - Y\|$. Для фиксированных центров ядер c_i и параметров ширины σ_i обучение RBFN эквивалентно нахождению решения \tilde{W} с помощью метода наименьших квадратов для системы линейных уравнений $\Phi W = Y$:

$$\|\Phi \tilde{W} - Y\|^2 = \min_W \|\Phi W - Y\|^2.$$

Однако в большинстве практических применений $h \neq N$ (количество нейронов (ядер) может быть намного меньше числа обучающих данных, т. е. $h \ll N$), и для матрицы W может не существовать таких w_i , $i = \overline{0, h}$ что $\Phi W = Y$.

Единственное наилучшее приближенное решение \tilde{W} по методу наименьших квадратов указанной линейной системы можно найти следующим образом:

$$\tilde{W} = \Phi Y,$$

где $\Phi = (\Phi^T \Phi)^{-1} \Phi^T$ — псевдообратная матрица (матрица Мура-Пенроуза) для случая невырожденной матрицы Φ .

Таким образом, подобно SLFN, алгоритм экстремального обучения (ELM) для RBFN может быть сформулирован таким образом.

5. Алгоритм ELM для RBFN

Заданы: обучающий (тренировочный) набор данных $(x(k), y(k))$, $x(k) \in R^n$ и $y(k) \in R^m$, $k = \overline{1, N}$, вид ядерной функции $\varphi_i(x, c_i, \sigma_i)$ и предельное количество нейронов H (ядер) RBFN.

Шаг 1. Определяем центры c_i ядерных функций $\varphi_i(x, c_i, \sigma_i)$ с помощью SOM, а также их параметры ширины σ_i , $i = \overline{1, h}$ (т. е. формируем RBFN с h ядрами в соответствии с подходом описанным выше).

Шаг 2. Вычисляем в скрытом слое сформированной RBFN значения выходной матрицы Φ .

Шаг 3. Вычисляем значения матрицы выходных весов W в соответствии с формулой: $W = \Phi Y$.

В результате получаем обученную RBFN.

6. Иллюстративный пример

Для апробации предложенной архитектуры нейронной сети рассмотрим задачу прогнозирования хаотического ряда, описываемого дифференциальным уравнением Мэки-Гласса [8], которое задается следующим уравнением ($\gamma = 0.1$, $\beta = 0.2$, $\tau = 17$):

$$\frac{\partial x}{\partial t} = \beta \frac{x_\tau}{1 + x_\tau^n} - \gamma x, \quad \gamma, \beta, n > 0, \quad (7)$$

где γ , β , n — некоторые коэффициенты; x_τ — значение переменной x в момент времени $(t - \tau)$.

В зависимости от значений параметров, уравнение воспроизводит ряд периодических и хаотических колебаний. В работе, эти значения были вычислены с помощью метода Рунге-Кутты четвертого порядка. Временной шаг принят равным 0.1, начальные условия $x(0) = 1.2$. График уравнения Мэки-Гласса приведен на рис. 2.

В качестве тестовой традиционно используются задача прогнозирования значения $x(k + 6)$

временного ряда (7) с параметром задержки $k = 17$ по значениям $x(k-18)$, $x(k-12)$, $x(k-6)$ и $x(k)$.

Для обучения нейронных сетей использовались первые 500 значений ряда, а для проверки – следующие 500.

На рис. 3. представлен процесс эволюции рецепторных полей ($c_1, \Sigma_1, c_2, \Sigma_2$) двух активационных функций эволюционирующей RBFN,

для случая online прогнозирования ряда Мэкки-Гласса (7).

Перед началом обработки полученный временной ряд нормировался таким образом, чтобы его значения находились в интервале $[-1; 1]$. После этого начиналось обучение нейронной сети. Результаты аппроксимации, обучения и прогнозирования представлены на рис. 4 – 6.

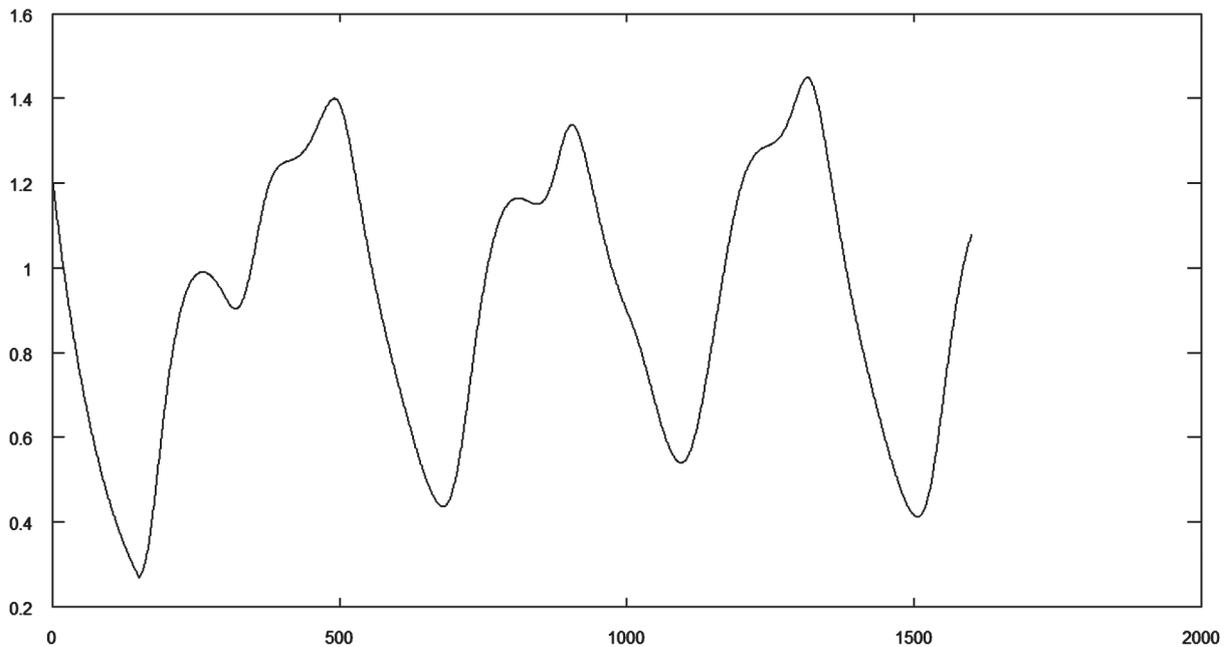


Рис. 2. График уравнения Мэкки-Гласса

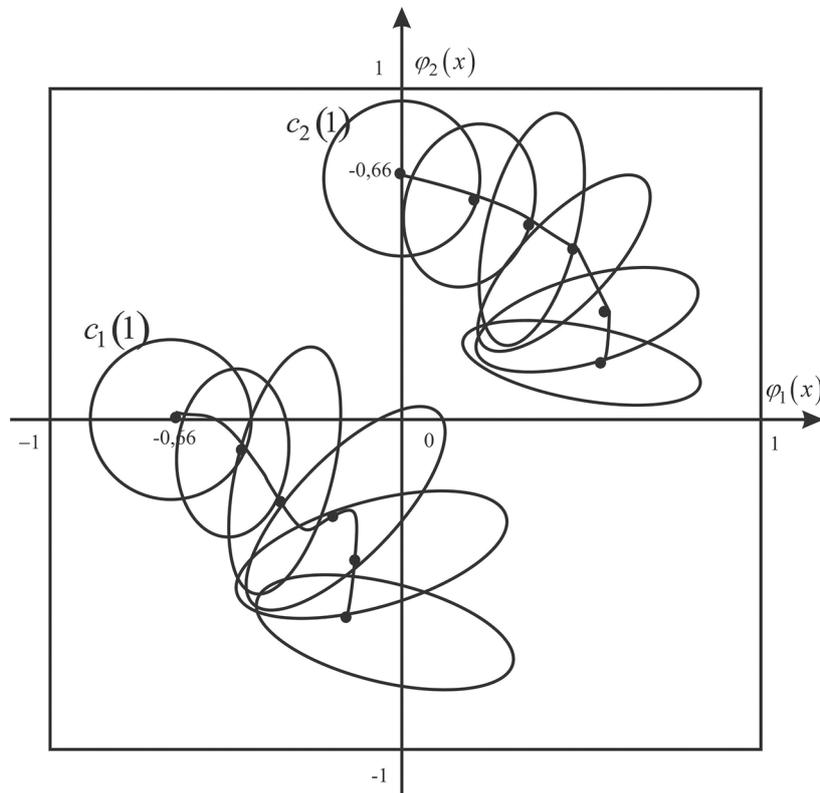


Рис. 3. Процесс эволюции рецепторных полей двух активационных функций эволюционирующей радиально-базисной нейронной сети

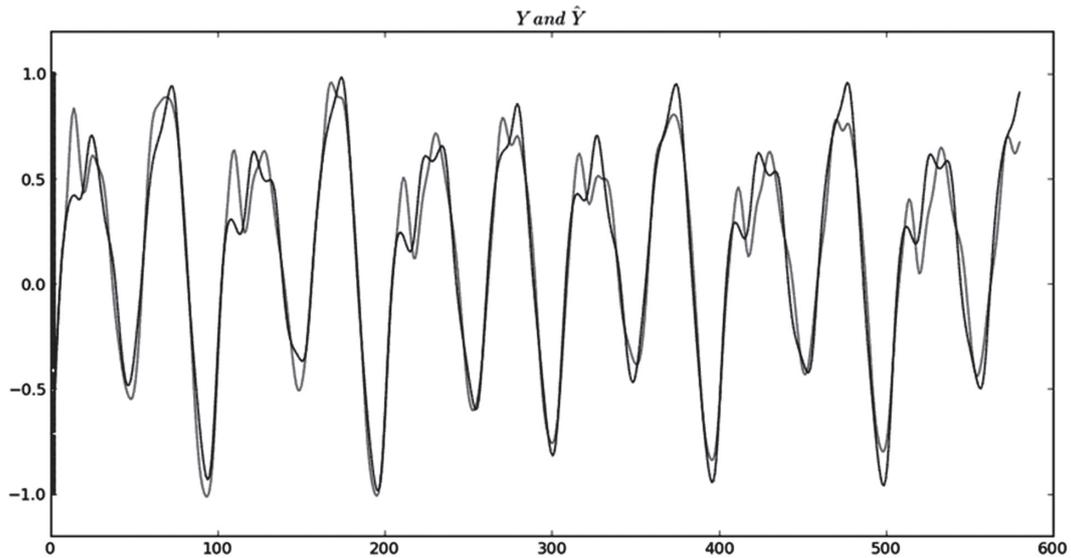


Рис. 4. Результаты аппроксимации временного ряда Мэки-Гласса

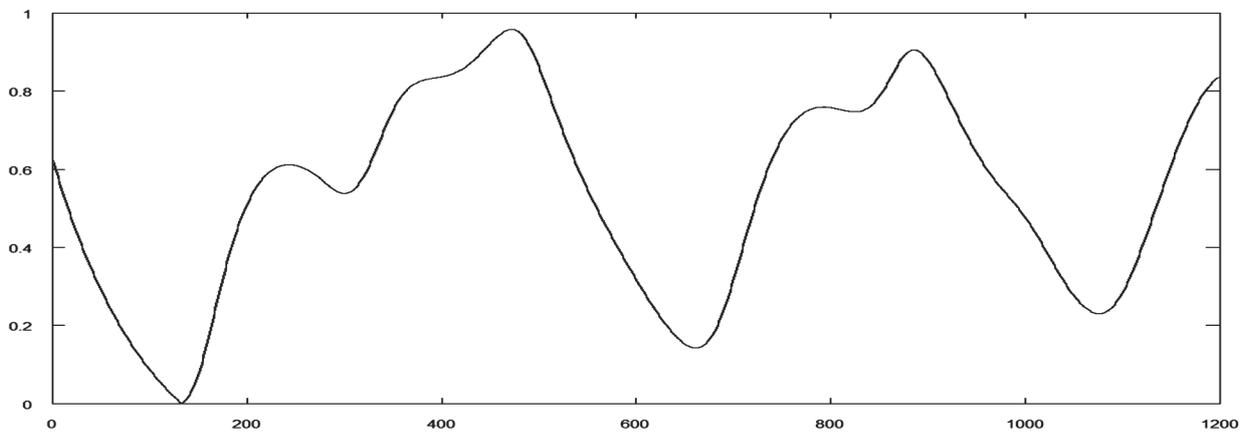


Рис. 5. Результаты обучения нейронной сети

При этом ошибка обучения составила 0.02%. Такие точные результаты обосновываются стационарностью свойств ряда Мэки-Гласса.

Далее система была переведена в режим прогнозирования значений. Прогнозирование значений ряда производилось в online-режиме, причем пересчет настраиваемых параметров сети, а это — центры функций принадлежности и синаптические веса, не производился по причине стационарности ряда Мэки-Гласса. Было принято решение спрогнозировать 700 значений, а затем сравнить с истинными значениями ряда. Результаты прогнозирования приведены на рис. 6 и рис. 7.

Прогнозирование значений было организовано таким образом: на каждой итерации процедуры предсказания прогнозируемое значение при помощи квадратичного критерия ошибки сравнивалось с истинным, а результат сохранялся. Система быстро адаптировалась к подаваемым ей на вход значениям, и в результате разница между истинным и прогнозируемым значением значительно снизилась.

Заключение

Предложенный в работе подход к обучению эволюционирующей радиально-базисной нейронной сети позволяет в режиме реального масштаба времени производить не только настройку выходных весов ИНС и параметров радиально-базисных функций, но и дает возможность изменять саму архитектуру сети в процессе поступления новой информации на ее вход. Это позволило решить проблему «проклятия размерности», которая присуща RBFN.

Использование метода экстремального обучения (ELM) вместо традиционных методов и относительная компактность архитектуры сети обеспечили высокую скорость ее обучения при приемлемой точности результатов, что было подтверждено рядом вычислительных экспериментов в ходе проведения имитационного моделирования.

В перспективе необходимо провести эксперименты связанные с установлением зависимости скорости обучения RBFN и точности результатов от использования различных видов активационных

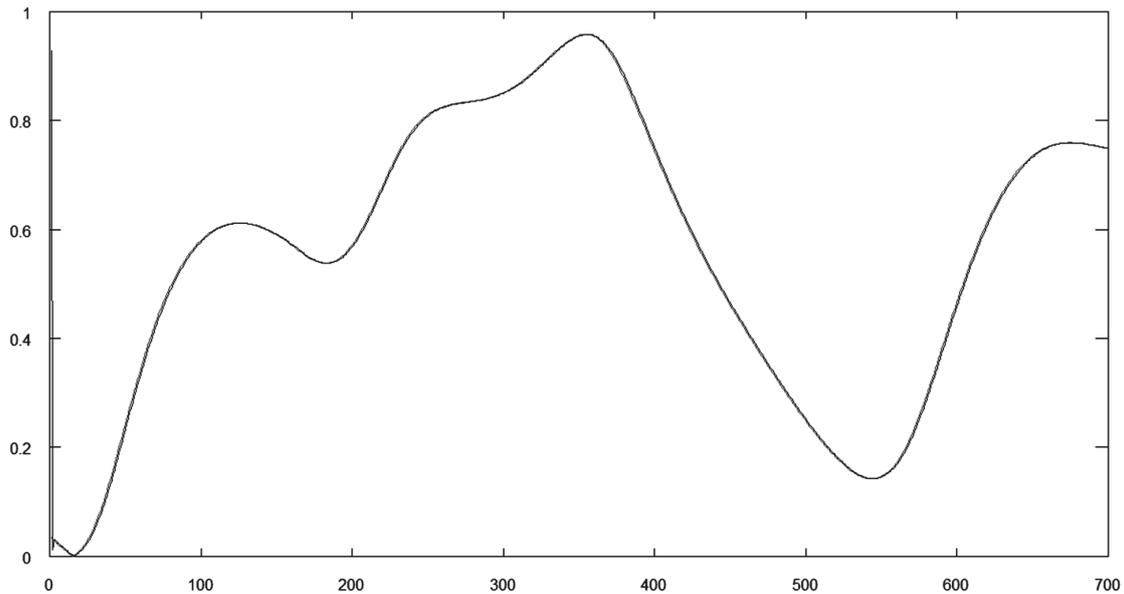


Рис 6. Результаты прогнозирования значений ряда Мэки-Гласса

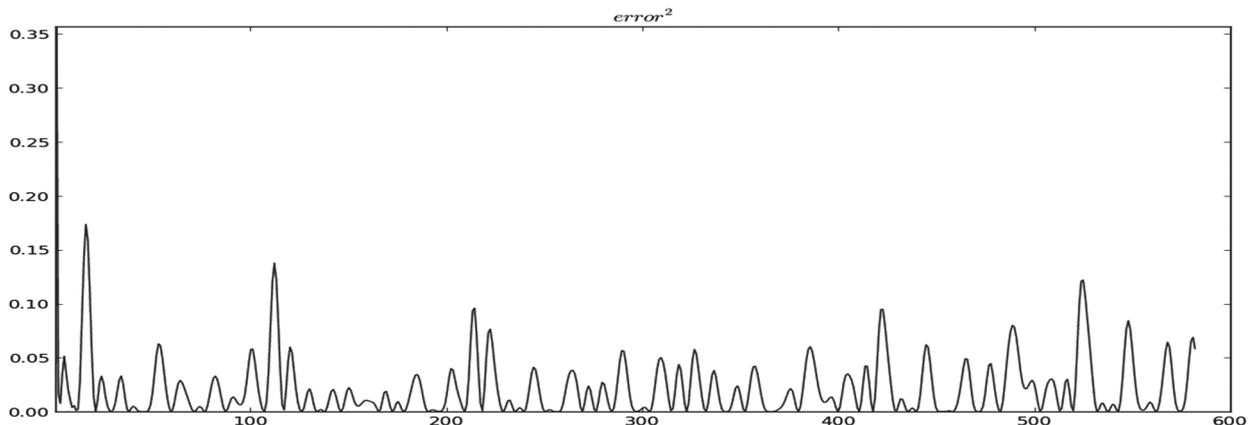


Рис. 7. График изменения ошибки аппроксимации на временном ряде Мэки-Гласса

ядерных функций, таких например, как функции В. Епанечникова [9], Коши, различных потенциальных функций [10], а также оценок Парзена [11] и т. п.

Список литературы:

- [1] Bifet A. Adaptive Stream Mining: Pattern Learning and Mining from Evolving Data Streams / A. Bifet // IOS Press. – 2010. – 224 p.
- [2] Park J. Universal approximation using radial-basis-function networks / J. Park, I.W. Sandberg // Neural Computation. – 1991. Vol. 3. P. 246 –257.
- [3] Angelov P. Evolving computational intelligence systems / P. Angelov, N. Kasabov // Proc. 1st Int. Workshop on Genetic Fuzzy Systems. – Granada, Spain, 2005. P. 76 – 82.
- [4] Kohonen T. Self-Organizing Maps / T. Kohonen // Berlin: Springer-Verlag. – 1995. – 362 p.
- [5] Huang G.-B. Extreme Learning Machine: Theory and applications / G.-B. Huang, Q.-Y. Zhu, C.-K. Siew // Neurocomputing. – 2006. Vol. 70. P. 489 – 501.
- [6] Bodyanskiy Y.V. An evolving radial basis neural network with adaptive learning of its parameters and architecture / Y.V. Bodyanskiy, A.K. Tyshchenko, A.A. Deineko // Automatic Control and Computer Sciences. – 2015. – 49 (5). – P. 255–260.
- [7] Huang G.-B. Extreme Learning Machine with Randomly Assigned RBF Kernels. / G.-B. Huang, C.-K. Siew // International Journal of Information Technology. – 2005. Vol. 11, No. 1. P. 16 – 24.
- [8] Гласс, Л. От часов к хаосу. Ритмы жизни. / Л. Гласс, М. Мэки – М.: Мир. – 1991. – 107 с.
- [9] Епанечников В.А. Непараметрическая оценка многомерной плотности вероятности / В.А. Епанечников // Теория вероятностей и ее применение – 1968 – 14 – №1 – С. 156 – 161.
- [10] Айзерман М. А. Метод потенциальных функций в теории обучения машин. / М. А. Айзерман, Э.М. Браверман, Л.И Розоноэр // М.: Наука, 1970. – 384 с.
- [11] Parzen E. On the estimation of a probability density function and the mode / E. Parzen // Ann. Math. Statist. – 1962. – 38. – P. 1065 – 1076.

Поступила в редколлегию 16.10.2019

УДК 004.032.26:007.052:629.7.072

В.В. Москаленко¹, А.С. Москаленко², А.Г. Коробов³, М.О. Зарецький⁴¹Сумський державний університет, м. Суми, Україна, v.moskalenko@cs.sumdu.edu.ua²Сумський державний університет, м. Суми, Україна, a.moskalenko@cs.sumdu.edu.ua³Сумський державний університет, м. Суми, Україна, a.korobov@cs.sumdu.edu.ua⁴Сумський державний університет, м. Суми, Україна, m.zaretskij@cs.sumdu.edu.ua

МОДЕЛЬ ТА АЛГОРИТМ НАВЧАННЯ СИСТЕМИ АВТОНОМНОЇ НАВІГАЦІЇ ДЛЯ МАЛОГАБАРИТНОГО БЕЗПІЛОТНОГО АПАРАТУ

В статті розглядається нова модель системи автономної навігації малогабаритного безпілотного апарату, що включає згорткову нейронну мережу для виділення візуальних ознак, машину екстремального навчання для оцінки переміщення та інформаційно-екстремальний класифікатор з використанням бустингу для оцінки перешкод. Запропоновано новий алгоритм навчання моделі, що полягає у навчанні екстрактора ознак без вчителя на основі зростаючого розріджено кодуючого нейронного газу, навчанні з учителем вирішальних правил та тонкій настройці моделі за допомогою алгоритму симуляції відпалу. Розглянуто комплексний критерій вибору параметрів моделі екстрактора ознак. Результати фізичного моделювання з оптимальною моделлю на відкритих KITTI-наборах даних підтверджують придатність запропонованих алгоритмів для практичного використання.

НАВІГАЦІЯ, ВІЗУАЛЬНА ОДОМЕТРИЯ, ЗГОРТКОВА НЕЙРОННА МЕРЕЖА, НЕЙРОННИЙ ГАЗ, ІНФОРМАЦІЙНИЙ КРИТЕРІЙ, ЕКСТРЕМАЛЬНЕ НАВЧАННЯ.

В.В. Москаленко, А.С. Москаленко, А.Г. Коробов, Н.А. Зарецький. Модель и алгоритм обучения системы автономной навигации для малогабаритного беспилотного аппарата. В статье рассматривается новая модель системы автономной навигации малогабаритного беспилотного аппарата, которая включает свёрточную нейронную сеть для выделения визуальных признаков, машину экстремального обучения для оценки перемещения и информационно-экстремальный классификатор с использованием бустинга для оценки препятствий. Предложен новый алгоритм обучения модели, который заключается в обучении экстрактора признаков без учителя на основе растущего разреженного кодирующего нейронного газа, обучении с учителем решающих правил и тонкой настройке модели с помощью алгоритма симуляции отжига. Рассмотрен комплексный критерий выбора параметров модели экстрактора признаков. Результаты физического моделирования с оптимальной моделью на открытых KITTI-наборах данных подтверждают пригодность предложенных алгоритмов для практического использования.

НАВІГАЦІЯ, ВІЗУАЛЬНА ОДОМЕТРИЯ, СВЁРТОЧНЫЕ НЕЙРОННЫЕ СЕТИ, НЕЙРОННИЙ ГАЗ, ІНФОРМАЦІОННИЙ КРИТЕРІЙ, ЕКСТРЕМАЛЬНЕ ОБУЧЕННЯ.

V.V. Moskalenko, A.S. Moskalenko, A.G. Korobov, M.O. Zaretskiy. Aerial compact drone autonomous navigation system – the model and training algorithms. The paper presents a novel model of convolutional neural network for visual feature extraction, extreme-learning machine for position displacement estimation and boosted information-extreme classifier for obstacle prediction with new training algorithms to build decision rules of autonomous navigation system for compact drones are presented in the paper. Growing sparse-coding neural gas algorithm for unsupervised training of the convolution filters, supervised learning algorithms for training decision rules and simulated annealing search algorithm for fine tuning are proposed. The complex criterion for choosing parameter of feature extractor model is considered. Simulation results with optimal model on open KITTI-datasets confirm the suitability of proposed algorithms for practical usage.

NAVIGATION, VISUAL ODOMETRY, CONVOLUTIONAL NEURAL NETWORK, NEURAL GAS, INFORMATION CRITERION, EXTREME LEARNING.

Вступ

Автономна навігація має важливе значення у пошукових та рятувальних задачах, оскільки ручне дистанційне керування польотом за даними відеопотоку дуже проблематичне поблизу будівель, дерев та всередині приміщень [1–3]. Крім того, система глобального позиціонування може бути не надійною в умовах низького рівня супутникового покриття та внаслідок багатопроменевого поширення сигналу. Альтернативним рішенням може бути використання компактного лазерного сканера. Проте таке рішення дороге, і лазер має низьку

частоту [2]. Рішення на основі машинного зору найбільш придатне для автономної навігації з точки зору ваги, ціни та інформації яку можна отримати. За допомогою бортової камери літальний апарат може оцінювати власне переміщення та одночасно одержувати інформацію про навколишнє середовище польоту.

Геометричні методи візуальної одометрії, такі як прямі, напівпрямі та з використанням локальних ознак, набули найбільшого поширення [4, 5]. Проте геометричні методи не мають стійкості до ряду дуже поширених ефектів, таких як зміна

освітлення, рух динамічних об'єктів в полі зору камери, різні калібрування камери, слаботекстурованість поверхонь в полі зору, шум та розмитість кадрів при русі.

Використання методів глибокого навчання та недорогих камер загального призначення є найбільш перспективним підходом для візуальної одометрії. В рамках даного підходу є широкі можливості тренування системи для забезпечення стійкості до параметрів камери та складних умов середовища [5, 6]. Однак глибокі моделі споживають великі обсяги обчислювальних ресурсів як в режимі навчання, так і в режимі екзамону. Крім того, для успішного навчання, глибокі моделі потребують великий обсяг розміченого набору даних. Зниження складності моделі може бути досягнуте за рахунок звуження області застосування, до певної доменної області. Але навіть невелика глибока модель є ресурсоємкою і вимагає великого обсягу розмічених навчальних даних, що призводить до значних накладних витрат та уповільнення адаптації до нових умов чи середовища функціонування.

Традиційна згорткова нейронна мережа, що складається з багатошарового згорткового екстрактора ознак та правил прийняття рішень у вигляді повнозв'язних нейронних шарів, є безперечним лідером серед моделей аналізу зображень [7, 8]. Однак, основними недоліками традиційних згорткових нейронних мереж є їх нездатність до аналізу процесів, що відбуваються в часі, а також висока обчислювальна складність алгоритму навчання на основі зворотного поширення помилки, що ускладнює адаптацію до змін під час експлуатації [9]. Крім того, важко заздалегідь оцінити необхідну кількість нейронів у кожному згортковому шарі. Тому підходи, основані на використанні алгоритму зростаючого нейронного газу та його модифікацій, для навчання нейронних мереж без вчителя є перспективними [1, 10]. Крім того, згортковий екстрактор ознак у поєднанні з розрідженим кодуванням залишається працездатним і за умов малих розмірів навчальних вибірок. При цьому значна незбалансованість цільових класів у нерозміченому наборі даних може бути врахована шляхом точного настроювання екстрактора ознак за критерієм ефективності вирішальних правил системи навігації [8]. Комплексне налаштування в гібридних моделях часто обмежується багатьма неузгодженостями, недиференційованістю цільових функцій чи проміжних ланок та іншим. У цьому випадку, точна настройка може бути ефективно виконана будь-яким пошуковим метаевристичним алгоритмом. Однак швидкість машинного навчання залежатиме від складності побудови оптимальних вирішальних правил.

Завдання оцінки власного переміщення є найважливішою складовою процесу візуальної навігації, і його можна сформулювати як регресійний аналіз для визначення зсуву позиції. Метод опорних векторів та машина екстремального навчання (Extreme Learning Machine, ELM) широко використовуються як моделі регресії в гібридних інтелектуальних системах, особливо з наявністю обмежень на обчислювальні ресурси та розмір розміченого набору даних. У цьому випадку, ELM характеризується найбільш швидким навчанням для отримання рішення регресійної задачі згідно методу найменших квадратів [11]. Для того, щоб уникнути перенавчання, яке відбувається при великій кількості нейронів прихованого шару, важливо дослідити поступове навчання шляхом послідовного додавання прихованих нейронів.

Інформаційно-екстремальний класифікатор високорівневих ознак є одним з перспективних підходів для класифікаційного аналізу зображень на предмет перешкод. Інформаційно-екстремальне навчання дозволяє уникнути проблем пов'язаних з ефективністю обчислень та забезпеченням узагальнюючих властивостей. Основна ідея інформаційно-екстремального класифікатора полягає у перетворенні вхідного простору первинних ознак у бінарний простір Хеммінга з радіально-базисними вирішальними правилами. Цей підхід забезпечує високу обчислювальну ефективність, оскільки використовуються лише прості операції, такі як порівняння з порогами та розрахунок відстані Хеммінга на основі логічної операції XOR та підрахунку ненульових бітів. Проте, методи швидкої оптимізації порогів, для кодування ознак все ще не запропоновані. При цьому індукція ознак на основі алгоритмів випадкового лісу та бустінгу є двома особливо перспективними підходами до прискорення оптимізації порогів для бінарного кодування ознак [12-14].

Для здійснення автономної навігації та ідентифікації перешкод в умовах обмежених обчислювальних ресурсів у даній статті пропонується модель багатошарового згорткового розріджено кодуєчого екстрактора ознак з просторово-часових візуальних даних. Крім того, пропонуються алгоритми тренування моделі, основані на попередньому навчанні без вчителя та тонкій настройці з учителем екстрактора ознак з метою максимізації ефективності інформаційно-екстремального класифікатора та регресійної моделі у вигляді машини екстремального навчання.

1. Опис даних та архітектура моделі

Для формування вхідного математичного опису інтелектуальної інформаційної системи, використовувались набори даних KITTI VO / SLAM 07 та

09, що містять як послідовність кадрів зображення з рухомої відеокамери, так і реальні дані про рух по трьох координатних осях, отриманих з GPS та LiDaR (Набір даних: http://www.cvlibs.net/datasets/kitti/raw_data.php?type=residential. Ліцензія на набір даних: авторська ліцензія Creative Commons Attribution-NonCommercial-ShareAlike 3.0), що знаходяться на платформі Kaggle [8]. Цей набір даних був записаний з відносно низькою частотою кадрів (10 кадрів в секунду), в міських районах з багатьма динамічними об'єктами та швидкістю руху до 90 км / год. Попередня обробка зображень виконувалася із застосуванням функцій зміни розміру, перетворення у відтінки сірого та корекції спотворень внаслідок дисторсії [9]. Для цього необхідні були параметри калібрування камери, однак висока точність не потрібна. Використовувалися як навчальні, так і тестові відеокліпи набору даних KITTI Vision, без врахування анотацій. Щоб зменшити обчислювану складність алгоритмів, зображення були стиснуті до розміру 200x200 пікселів.

Навчена модель візуальної навігації повинна забезпечувати ухилення від перешкод та контроль власної позиції за допомогою одометрії. Запропонована модель візуальної навігації може бути навчена або як самостійна система, або як допоміжна система у вигляді паралельного коректора помилок для геометричної одометрії, основаної на використанні локальних ознак. У випадку допоміжного варіанту модель геометричної одометрії та коректор помилок повинні бути синхронізовані.

Схема інтелектуальної навігаційної системи для малогабаритного літального апарату показана на рис. 1.

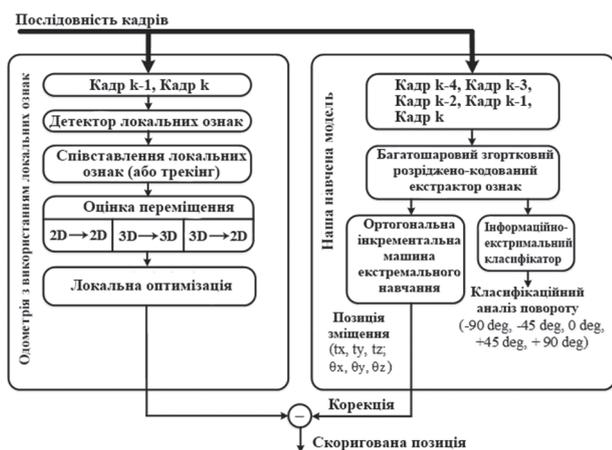


Рис. 1. Модель геометричної візуальної одометрії з запропонованою моделлю паралельного коректора помилок

Для того, щоб виділити ознаки візуальних спостережень, пропонується використовувати згорткову нейронну мережу, на вхід якої надходить багатоканальне зображення, сформоване із серії послідовних відео кадрів у градації сірого. Згорткова

нейронна мережа має багат шарову структуру, на виході якої формується високорівневий ознаковий опис візуальних спостережень. При цьому згорткові фільтри навчаються без вчителя, послідовно шар за шаром. Для виявлення перешкод та формування відповідної реакції використовується інформаційно-екстремальний класифікатор, що навчається з учителем на навчальних зразках, закодованих відповідними високорівневими ознаками. Регресійна модель в рамках ELM використовується для відображення візуальних ознак у відповідну оцінку зміщення відеокамери в просторі.

На рис. 2 показано 4-х шарову архітектуру згорткової нейронної мережі, в першому шарі якої знаходяться 3D фільтри різних розмірів: $5 \times 5 \times K_1$, $3 \times 3 \times K_1$ та $1 \times 1 \times K_1$. Кількість фільтрів регулюється параметром K_2 . Для збереження одного і того ж розміру карт ознак, створених багатомірними фільтрами, використовується техніка заповнення нулями [8]. У другому та третьому шарах параметр відступу для сканування карти ознак з багатомірними фільтрами становить 3 та 2, відповідно. При цьому пропонується використовувати алгоритм ортогонального узгодженого переслідування Orthogonal Matching Pursuit (OMP) [10] та функцію RELU, $y = \max(0, x)$, для обчислення багатоканальних пікселів карти ознак, проте для уникнення втрати інформації, можна подвоїти карту ознак за допомогою функції $y = \{\max(0, x), \max(0, -x)\}$. На рис. 2 не показано функцій обчислення пікселів карти ознак.

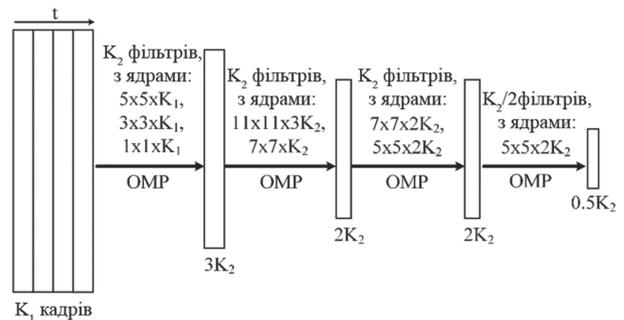


Рис. 2. Архітектура згорткової розріджено кодуючої моделі для екстракції візуальних ознак в навігаційній системі

Для навчання регресійної моделі $y = f(x)$ у вигляді мережі прямого поширення з одним прихованим шаром (МПОП) використовується набір навчальних даних, закодованих візуальними ознаками

$$\{(x_j, y_j) | x_j \in R^N, y_j \in R^M, 1 \leq j \leq n\},$$

де $x_j = (x_{j1}, x_{j2}, \dots, x_{jN})^T$. При цьому вихідна змінна $y_j \in R^M$ відповідає вектору зміщення положення, t , та вектору повороту, \bar{R} , камери $y = (y_1, y_2, \dots, y_n)^T = (t, \bar{R})^T$. МПОП з R адитивними

прихованими вузлами та функцією активації $\varphi(x)$ може бути подана у вигляді системи рівнянь

$$\sum_{r=1}^R \beta_r \varphi(w_r^T x_j + b_r) = o_j, 1 \leq j \leq n$$

де $w_r = (w_{r1}, w_{r2}, \dots, w_{rN})^T$ – вектор вагових коефіцієнтів, який зв’язує вхідний шар з r -м прихованим вузлом; b_r – зміщення r -го прихованого вузла; $\beta_r = (\beta_{r1}, \beta_{r2}, \dots, \beta_{rM})$ – вектор вагових коефіцієнтів, що зв’язує вихідний шар з r -м прихованим вузлом; o_j – вихід мережі для вхідного вектора x_j ; $\varphi(x)$ – функція активації.

Мережа з R прихованими вузлами може відтворити ці N зразків з нульовою помилкою, якщо всі параметри можуть вільно коригуватись, тобто існують, β_r , w_r і b_r . Вищевказані виразів можуть бути компактно переписані як матрична рівність

$$H\beta = Y,$$

де

$$H = \begin{bmatrix} \varphi(w_1^T x_1 + b_1) & \dots & \varphi(w_R^T x_1 + b_R) \\ \dots & \dots & \dots \\ \varphi(w_1^T x_n + b_1) & \dots & \varphi(w_R^T x_n + b_R) \end{bmatrix}_{n \times R},$$

$$\beta = \begin{bmatrix} \beta_1^M \\ \dots \\ \beta_R^M \end{bmatrix}_{R \times M}, \quad Y = \begin{bmatrix} y_1^M \\ \dots \\ y_n^M \end{bmatrix}_{n \times M}$$

де H – матриця виходу прихованого шару.

Інформаційно-екстремальний класифікатор в режимі екзамену приймає рішення про належність вхідного вектора x з відповідним двійковим поданням b до одного з класів повороту $\{X_z^o \mid z = \overline{1, Z}\}$ відповідно до максимального значення функції належності $\mu_z(b)$ згідно виразу $\arg \max \{\mu_z(b)\}$. У такому випадку функція належності $\mu_z(b)$ до класу X_z^o , оптимальний контейнер якого має опорний вектор b_z^* та радіус d_z^* , розраховується за формулою

$$\mu_z(b) = \exp\left(-\sum_{i=1}^{N_z} b_i \oplus b_{z,i}^* / d_z^*\right).$$

Двійкове кодування зразків даних x_j здійснюється шляхом конкатенації шляхів рішень у деревах з ансамблю T_1, \dots, T_k , де шлях подається у вигляді бінарного коду. При цьому кожен ненульовий біт закодованого шляху відповідає номеру вузлу дерева, через який проходив шлях класифікації вхідного зразка x_j відповідним деревом [13, 14].

2. Структура алгоритму навчання

Процес машинного навчання навігаційної системи орієнтований на визначення оптимального вектора параметрів g , які забезпечують максимум комплексного критерію

$$J = \frac{\bar{E}}{E_{\max}} \cdot \frac{\varepsilon_{\min}}{\varepsilon} \cdot \frac{C_{\min}}{C}, \quad (1)$$

$$g^* = \arg \max_G \{J(g)\}, \quad (2)$$

де \bar{E} – інформаційний критерій ефективності навчання для розпізнавання перешкод, усереднений за набором класів; ε – значення середньо квадратичної похибки регресії при визначенні зміни координат камери у просторі; C – критерій обчислювальної трудомісткості моделі екстракції ознак; E_{\max} , ε_{\min} , C_{\min} – максимальне можливе значення інформаційного критерію ефективності навчання класифікатора, мінімально допустиме значення похибки регресійної моделі та критерію обчислювальної трудомісткості екстрактора ознак, відповідно; G – допустима область значень параметрів, які впливають на екстракцію ознак і прийняття рішень.

Пропонується провести навчання екстрактора ознак без вчителя, з використанням зростаючого розріджено кодуємого нейронного газу, основаного на принципах зростаючого нейронного газу та розрідженого кодування. У цьому випадку набір даних для навчання згорткових фільтрів формується шляхом розбиття вхідних зображень або карт ознак на патчі. Ці патчі трансформуються до 1D-векторів та надходять на вхід зростаючого розріджено кодуємого нейронного газу, основні етапи якого наведені нижче:

1. ініціалізація лічильника навчальних векторів $t := 0$;
2. два початкових вузли (нейрони) w_a і w_b ініціалізуються шляхом випадкового вибору векторів з навчальних даних. Вузли w_a і w_b з’єднуються ребром, вік якого встановлюється нульовим. Ці вузли вважаються нефіксованими;
3. обирається наступний вектор x , що нормалізується шляхом приведення до одиничної довжини (L2-нормування);
4. нормалізується кожний базисний вектор, $w_k, k = \overline{1, M}$, шляхом приведення до одиничної довжини (L2-нормалізація);
5. розраховується міра схожості вхідного вектора x до базисних векторів $w_{s_k} \in W$ для сортування $-(w_{s_0}^T x)^2 \leq \dots \leq -(w_{s_k}^T x)^2 \leq \dots \leq -(w_{s_{M-1}}^T x)^2$;
6. визначається найближчий вузол w_{s_0} та другий за близькістю вузол w_{s_1} ;
7. збільшується на одиницю вік всіх вузлів, інцидентних до w_{s_0} ;
8. якщо вузол w_{s_0} фіксований, то перехід до кроку 9, в інакше – крок 10;
9. якщо $(w_{s_0}^T x)^2 \geq \nu$, то перехід до кроку 12. В іншому випадку додаються новий нефіксований нейрон $w_r = x$ та нове ребро, яке з’єднує w_r і w_{s_0} , потім перехід до кроку 13;
10. вузол w_{s_0} та його топологічні сусіди (вузли, зв’язані з ним ребрами) зміщуються у напрямку

вихідного вектора x відповідно до правила Ойа [14] за формулами

$$\begin{aligned} \Delta w_{s_0} &= \varepsilon_b \eta_t y_0 (x - y_0 w_{s_0}), \quad y_0 := w_{s_0}^T x, \\ \Delta w_{s_n} &= \varepsilon_n \eta_t y_n (x - y_n w_{s_n}), \quad y_n := w_{s_n}^T x, \\ 0 < \varepsilon_b &\ll 1, \quad 0 < \varepsilon_n \ll \varepsilon_b, \\ \eta_t &:= \eta_0 (\eta_{final} / \eta_0)^{t/t_{max}}, \end{aligned}$$

де $\Delta w_{s_0}, \Delta w_{s_n}$ – вектори корекції вагових коефіцієнтів нейрона-переможця та його топологічних сусідів відповідно; $\varepsilon_b, \varepsilon_n$ – константи міри оновлення вагових коефіцієнтів нейрона-переможця та його топологічних сусідів відповідно; $\eta_0, \eta_t, \eta_{final}$ – початкове, поточне та кінцеве значення швидкості навчання відповідно;

11. якщо $(w_{s_0}^T E)^2 \geq \nu$, то нейрон w_{s_0} позначається як фіксований;

12. якщо w_{s_0} і w_{s_1} з'єднані ребром, їхній вік зменшується, в іншому випадку – між w_{s_0} і w_{s_1} формується нове ребро з нульовим віком;

13. всі ребра графа з віком більше, ніж a_{max} видаляються. У тому випадку, коли у деяких вузлах немає ребер (вузли стають ізольованими), ці вузли також видаляються;

14. якщо $t < t_{max}$, то переходимо до кроку 15, інакше – збільшуємо лічильник $t := t + 1$ і переходимо до кроку 3;

15. якщо всі нейрони фіксовані, то виконання алгоритму припиняється, інакше перехід до кроку 3 і починається нова епоха навчання (повторення набору даних навчання).

Для класифікаційного аналізу вихідної карти ознак формується навчальний набір даних $D = \{x_j, y_j \mid j = \overline{1, n}\}$, де n – обсяг вибірки, а y_j – мітка j -го пікселя, яка відповідає одному з класів алфавіту розпізнавання $\{X_z^o \mid z = \overline{1, Z}\}$. При цьому інформаційно-екстремальний класифікатор оцінює належність j -го елемента даних (пікселя на карті ознак) за його N_1 ознаками до одного з Z класів. Це здійснюється шляхом кодування ознак на основі дерев рішень та вирішальних правил, побудованих в радіальному базисі бінарного простору Хемінга. Навчання інформаційно-екстремального класифікатора відбувається відповідно до наступних кроків:

1. Ініціалізація вагових коефіцієнтів $w_j = 1/n$.

2. Для $k = 1 \dots K$ виконуються наступні кроки:

3. Генерація D_k набору навчальних даних з усього набору D з використанням функції розподілу ймовірності $P(X = x_j) = w_j$.

4. Навчання дерева рішень T_k на наборі D_k із застосуванням ентропійного критерію збурдженості вузлів.

5. Виконання бінарного кодування вектора x_j з набору даних D шляхом об'єднання шляхів

рішення в деревах T_1, \dots, T_k . Кожне дерево формує двійковий код, в якому одиниці знаходяться у позиціях, що відповідають номерам вузлів, через які проходить шлях вектора x_j від кореня до термінального вузла. Нарешті, формування результуючого коду вектора x_j відбувається шляхом об'єднання кодів, сформованих кожним із дерев [12]. Результатом роботи цього етапу є бінарна матриця $\{b_{z,s,i} \mid i = \overline{1, N_2}; s = \overline{1, n_z}; z = \overline{1, Z}\}$, де N_2 кількість індукованих бінарних ознак та n_z кількість реалізацій відповідного класу X_z^o , що задовольняє рівність $n = \sum_z n_z$.

6. Побудова інформаційно-екстремальних вирішальних правил в радіальному базисі бінарного простору Хемінга та обчислення інформаційного критерію

$$E_z^* = \max_{\{d\}} E_z(d),$$

де $\{d\} = \{0, 1, \dots, \left(\sum_i b_{z,i} \oplus b_{c,i} - 1\right)\}$ – набір концентричних радіусів з центром b_z ; b_z – опорний вектор розподілу даних в класі X_z^o , який розраховується за правилом

$$b_{z,i} = \begin{cases} 1, & \text{if } \frac{1}{n_z} \sum_{s=1}^{n_z} b_{z,s,i} > \frac{1}{Z} \sum_{A=1}^Z \frac{1}{n_A} \sum_{s=1}^{n_A} b_{A,s,i}; \\ 0, & \text{otherwise.} \end{cases}$$

де E_z – критерій ефективності побудови вирішальних правил в режимі навчання для класу X_z^o , який розраховується як нормалізована модифікація інформаційної міри Кульбака [11]:

$$E_z = \frac{1 - (\alpha_z + \beta_z)}{\log_2(2 + \zeta) - \log_2 \zeta} \cdot \log_2 \left[\frac{2 - (\alpha_z + \beta_z) + \zeta}{(\alpha_z + \beta_z) + \zeta} \right], \quad (3)$$

де α_z, β_z – частота хибно позитивних та хибно негативних спрацювань вирішальних правил класу X_z^o ; ζ – будь-яке мале знакододатне число, яке вводиться, для уникнення невизначеності при діленні на нуль.

З метою підвищення ефективності навчання, загальноприйнятим є зведення проблеми багатокласової класифікації до серії двокласових класифікацій за принципом «один проти всіх». Для уникнення проблеми незбалансованості класів розпізнавання, обумовленої переважанням у навчальному наборі даних негативних зразків, здійснюється введення синтетичного класу, який є альтернативним для X_z^o . Синтетичний клас представлено n_z векторами з інших класів, які найбільш близькі до опорного вектора b_z , де n_z – обсяг навчального набору даних класу X_z^o .

7. Тестування отриманих інформаційно-екстремальних вирішальних правил на наборі даних D та розрахунок коефіцієнту помилок для кожної

реалізації з D . При цьому, в режимі розпізнавання прийняття рішення про належність вектора b до одного з класів розпізнавання з алфавіту $\{X_z^o | z = \overline{1, Z}\}$ здійснюється за максимальним значенням функції належності $\mu_z(b)$ згідно виразу $\arg \max\{\mu_z(b)\}$. У цьому випадку функція належності $\mu_z(b)$ бінарного подання b вхідного вектора даних x до класу X_z^o , оптимальний контейнер якого має опорний вектор b_z^* та радіус d_z^* , розраховується за формулою

$$\mu_z(b) = \exp\left(-\sum_{i=1}^{N_2} b_i \oplus b_{z,i}^* / d_z^*\right).$$

8. Оновлення коефіцієнтів $\{w_j\}$, пропорційне отриманій помилці для вектора x_j .

9. Якщо $|E_k^* - E_{k-1}^*| < \varepsilon$ то вийти з циклу.

Ортогональна інкрементальна машина екстремального навчання пропонується використовувати як регресійну модель. Вона уникає надлишкових вузлів і отримує рішення рівнянь найменших квадратів $H\beta = Y$ на основі методу ортогоналізації Грем-Шмідта. Теоретичні докази збіжності ортогонального інкрементального екстремального навчання дано в праці [11]. Метод навчання ортогональної інкрементальної ELM відбувається відповідно до наступних кроків.

1. Встановлюється максимальна кількість ітерацій L_{\max} та бажана точність навчання E_0 .

2. Для $L = 1 \dots L_{\max}$ виконуються наступні кроки.

3. Збільшити кількість прихованих вузлів на один: $r = r + 1$.

4. Випадковим чином згенерувати один прихований нейрон та розрахувати його вихідний вектор h_r .

5. Якщо $r = 1$ тоді $v_r = h_r$ інакше

$$v_r = h_r - \frac{\langle v_1, h_r \rangle}{\langle v_1, v_1 \rangle} v_1 - \frac{\langle v_2, h_r \rangle}{\langle v_2, v_2 \rangle} v_2 - \dots - \frac{\langle v_{r-1}, h_r \rangle}{\langle v_{r-1}, v_{r-1} \rangle} v_{r-1}$$

6. Якщо $\|v_r\| \geq \varepsilon$, то виконується обчислення вихідної ваги для нового прихованого вузла $\beta_r = v_r^T E / (v_r^T v_r)$ та розраховується нова залишкова помилка $E = E - v_r \beta_r$ інакше $r = r - 1$.

7. Якщо $\|E\| \geq E_0$ відбувається вихід з основного циклу роботи.

Для апроксимації глобального оптимуму комплексного критерію під час навчання вирішальних правил та тонкої настройки екстрактора ознак ми пропонуємо використовувати метаевристичний алгоритм імітації відпалу [14]. Ефективність алгоритму симуляції відпалу залежить від реалізації процедури пошуку сусідів `create_neighbor_solution` для формування нового рішення s_i на i -ій ітерації алгоритму. На рис. 3 показано псевдо код алгоритму симуляції відпалу, на кожній ітерації якого виконується обчислення функції критерію $f()$, яка

обчислюється шляхом пропускання розміченого навчального набору даних через модель детектора та розрахунку комплексного критерію (1) [14].

```

s_current ← create_initial_solution()
s_best ← s_current
T ← T_0
c ← ε, 0 < ε < 1
for(i = 1 to epochs_max)
    s_i ← create_neighbor_solution(s_current)
    if f(s_i) ≥ f(s_current)
        s_current ← s_i
        if f(s_i) ≥ f(s_best)
            s_best ← s_i
        end if
    elseif exp((f(s_current) - f(s_i)) / T) > uniform_random(0,1)
        s_current ← s_i
    end if
    T ← c × T
end for
return(s_best)
    
```

Рис. 3. Псевдокод алгоритму симуляції відпалу

Аналіз псевдокоду на рис. 3 показує, що поточне рішення $s_{current}$, відносно якого відбувається пошук нових кращих рішень s_{best} , оновлюється у випадку знаходження нового рішення, що збільшує критерій (1) або випадковим чином з розподілу Гіббса. При цьому початкове рішення утворюється за допомогою процедури `create_initial_solution`. Для формування нового рішення пропонується використовувати найпростіший неадаптивний алгоритм, який можна представити у вигляді формули [14]

$$s_{current} = s_{current} + \text{uniform_random}(-1,1) \cdot \text{step_size},$$

де `uniform_random` – функція генерації випадкових чисел з рівномірного розподілу з заданого діапазону; `step_size` – це розмір діапазону пошуку нових рішень, сусідніх з $s_{current}$.

3. Результати моделювання та обговорення

Різні значення параметрів K_1 та K_2 впливають як на інформативність ознак, так і на обчислювальну трудомісткість. Ми пропонуємо вимірювати трудомісткість за кількістю операцій множення (Mul) та додавання (Add), що виконуються під час згорткових операцій з зображенням або картою ознак. Для архітектури мережі, показаної на рис. 2, трудомісткість може бути розрахована за формулою

$$C = K_2(2706472K_1 + 4438784K_2) \quad (4)$$

Для класифікатора та регресійної моделі оптимальна конфігурація згорткового екстрактора може бути різною, оскільки вона відповідає за різні завдання. Тому комплексний критерій (1) пропонує компроміс з точки зору точності правил прийняття рішень та обчислювальної складності екстрактора візуальних ознак.

Множина класів розпізнавання $\{X_z^o\}$ описує характерні перешкоди та відповідні команди реакції і має потужність $Z = 5$. Перший клас розпізнавання X_1^o характеризує рух вперед без повороту. Класи X_2^o і X_3^o відповідають лівому повороту 45 і 90 градусів відповідно. Класи X_4^o і X_5^o відповідають правому повороту 45 і 90 градусів відповідно. Обсяг навчальних та тестових даних кожного класу $n_z = 300$.

Спочатку пропонується навчати вирішальні правила моделі, використовуючи попередньо навчений без вчителя екстрактор ознак за допомогою зростаючого розріджено кодуєчого нейронного газу без тонкої настройки. При цьому під час навчання без вчителя використовуються три фіксованих значення параметра реконструкції набору навчальних даних v . Цей параметр безпосередньо впливає на кількість каналів карт ознак K_2 .

Для покращення результатів машинного навчання візуальної навігаційної системи підвищують інформативність ознакового опису шляхом тонкої настройки навчених без вчителя згорткових фільтрів. У цьому випадку використовувались наступні параметри алгоритму симуляції відпалу: $c = 0.98$, $T_0 = 10$, $epochs_max = 5000$, $step_size = 0.001$. На кожному кроці для тонкої настройки передбачається перепідготовка моделі регресії та класифікатора.

Для виявлення тенденції зміни середніх значень часткових та комплексного критеріїв при зростанні параметрів K_1 та K_2 , що впливають на розмір екстрактора ознак (рис. 2), було виконано симуляцію для трьох фіксованих значень кожного з цих параметрів (Табл. 1). При цьому оптимальні значення цих параметрів визначаються для відкритого набору даних КІТГІ-07.

Таблиця 1

Залежність часткових та комплексного критерію від параметрів екстрактора ознакового опису K_1 і K_2

v	K_1	K_2	\bar{E} / E_{max}	$\epsilon_{min} / \epsilon$	C_{min} / C	J
0.6	3	18	0,083	0,112	1,000	0,009296
0.6	5	18	0,101	0,188	0,827	0,015703
0.6	7	18	0,098	0,200	0,705	0,013818
0.7	3	21	0,28	0,688	0,297	0,057214
0.7	5	21	0,29	0,756	0,264	0,057879
0.7	7	21	0,29	0,775	0,238	0,053491
0.8	3	46	0,39	0,968	0,082	0,030957
0.8	5	46	0,55	1,000	0,077	0,04235
0.8	7	46	0,51	1,000	0,072	0,03672

Аналіз табл. 1 показує, що збільшення значень параметрів K_1 і K_2 загалом призводить до збільшення достовірності та обчислювальної складності (4) вирішальних правил класифікатора та

регресійної моделі. У той же час, збільшення параметра K_1 мало впливає на ефективність класифікатора внаслідок зниження ефективності пошукового алгоритму зі значним збільшенням розміру простору пошуку, тоді як помилка регресії однаково чутлива до значення параметрів K_1 і K_2 .

Однак, з огляду на те, що при зростанні K_1 та K_2 достовірність правил прийняття рішень зростає повільніше, ніж обчислювальна складність, то використання комплексного критерію J забезпечує компромісне рішення. Тобто ми вважаємо наступні значення параметрів $K_1^* = 5$ та $K_2^* = 21$ оптимальними.

У оптимальній конфігурації екстрактора ознак середнє значення інформаційного критерію функціональної ефективності рівне $\bar{E} = 0.29$. Це відповідає точності 95,2% для навчального набору та 94% для тестового набору. На рис. 4 показано графік зміни середнього значення інформаційного критерію ефективності (4) відносно кількості ітерацій алгоритму імітації відпалу.

Аналіз рис. 4 показує, що після 1000-ї ітерації зростання інформаційного критерію (4) почало сповільнюватися, а після 2500-ї ітерації практично не змінилося. Така зміна критерію вказує на те, що подальше збільшення інформаційного критерію досягне лише за рахунок збільшення інформативності ознакового опису або шляхом збільшення значень K_1 та K_2 , або шляхом вдосконалення структури екстрактора (рис. 2).

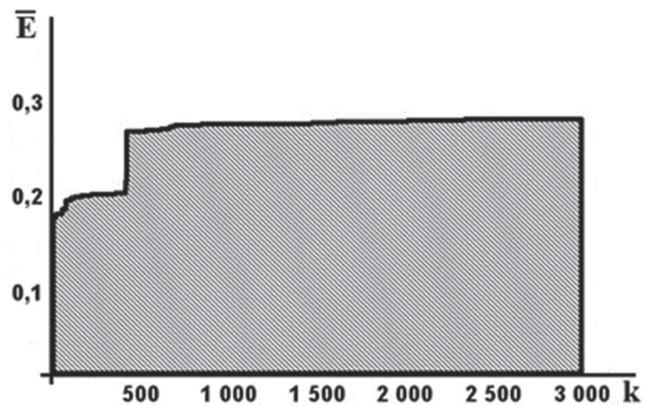


Рис. 4. Графік зміни усередненого значення інформаційного критерію ефективності (3) залежно від кількості ітерацій алгоритму імітації відпалу

Для візуальної оцінки ефективності машинного навчання навігаційної системи, реальна траєкторія, виміряна за допомогою GPS та LiDaR, може бути порівняна з реконструйованою траєкторією, отриманою з використанням навченої моделі. На рис. 5а показано результат використання одометрії на основі локальних ознак з системи ORB-SLAM [5, 6]. Максимальна кількість ключових точок обмежена до 100, щоб забезпечити необхідну частоту

кадрів у випадку бортової системи без застосування обчислювальних можливостей графічного процесора.

Аналіз рис. 5а показує, що реконструйована траєкторія має значну похибку. На рисунку 5б показано результат використання запропонованої моделі, навченої та перевіреної на тих самих наборах даних. У цьому випадку помилка реконструкції набагато менша. На рисунку 5в показано результат використання одометрії на основі локальних ознак від ORB-SLAM з паралельною корекцією на основі запропонованої моделі. Як бачите, точність реконструкції висока.

На рис. 6а показано, що навчена модель на наборі даних КІТТІ-07 має низьку продуктивність при тестуванні на іншому наборі даних. На рис. 6б показано, що повторне навчання моделі на новому наборі даних значно підвищило точність реконструкції траєкторії з набору даних.

Таким чином, результати, отримані на відкритих наборах даних, свідчать про придатність моделі для практичного використання. Щоб уникнути перенавчання, оптимізована складність моделі. Але обмежена ємність моделі викликає потребу повторного навчання при зміні доменної області використання.

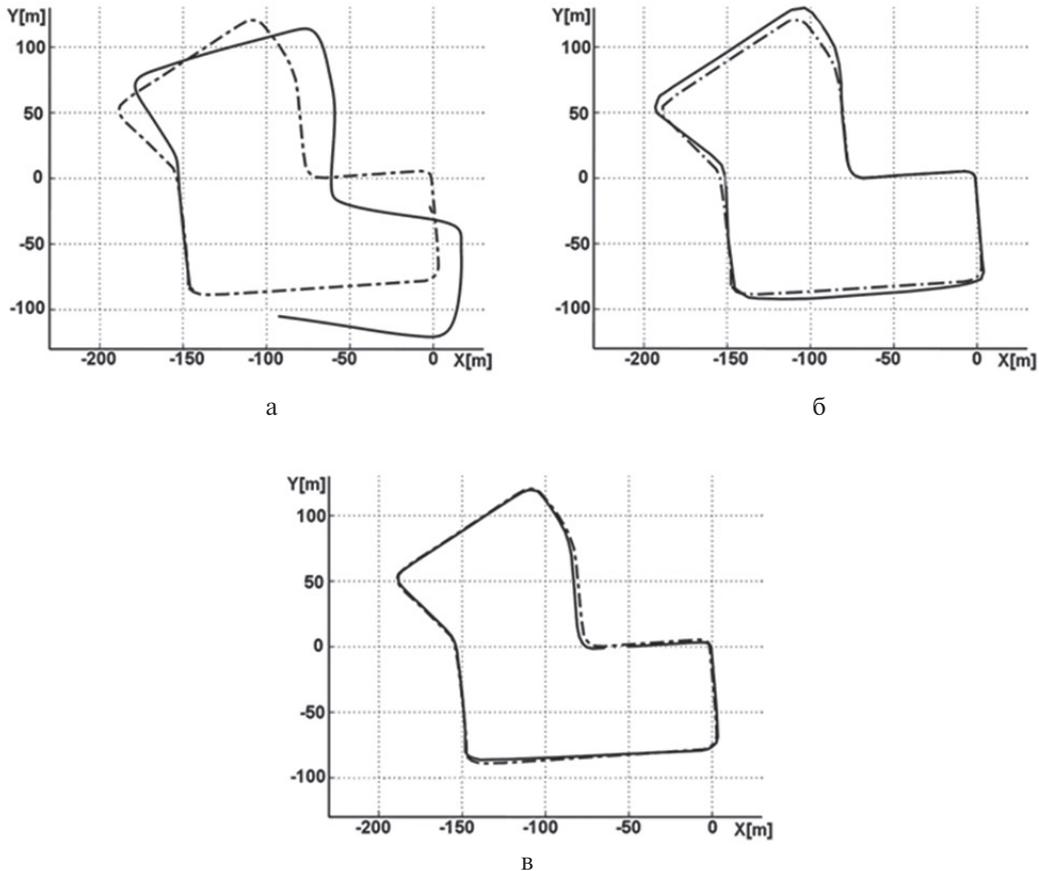


Рис. 5. Реальна траєкторія (пунктирна лінія) та реконструйована (суцільна лінія) для КІТТІ-07:

(а) – монокулярна одометрія на основі локальних ознак; (б) – запропонована модель; (в) – паралельна корекція помилок для монокулярної одометрії на основі локальних ознак з використанням запропонованої моделі

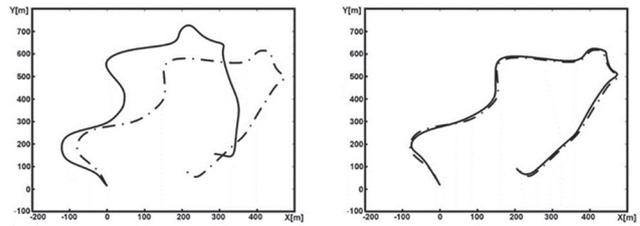


Рис. 6. Реальна траєкторія (пунктирна лінія) і реконструйована (суцільна лінія) траєкторії для КІТТІ-09:

(а) – навчена модель на КІТТІ-07;
(б) – після перенавчання на КІТТІ-09

Висновки

1. Наукова новизна результатів полягає в наступному:

– запропоновано нову модель автономної навігаційної системи для малогабаритного літального апарату. Модель складається з 4-х шарової згорткової розріджено кодууючої мережі, що складається з різномасштабних 3D згорткових фільтрів і приймає на вхід серію послідовних кадрів, машини екстремального навчання для оцінки переміщення і інформаційно-екстремальний класифікатор для прогнозування повороту;

– розроблено комплексний критерій оцінки ефективності моделі з урахуванням обчислювальної трудомісткості екстрактора ознак та точності вирішальних правил моделі;

– розроблено алгоритм навчання, що складається з 3-х частин. Перша частина – це навчання екстрактора ознак без вчителя на основі розтушеного розріджено кодуючого нейронного газу. Друга частина – навчання з учителем інформаційно-екстремального класифікатора та машини екстремально-го навчання. Остання частина – тонка настройка екстрактора ознак на основі метаевристичного алгоритму симуляції відпалу для підвищення ефективності вирішальних правил.

2. Практичне значення отриманих результатів для безпілотної авіації полягає у формуванні сучасної наукової та методологічної основи для проектування здатних до навчання систем автономної навігації для малогабаритних літальних апаратів, що мають ресурсні та інформаційні обмеження. У той же час результати симуляції моделі підтверджують високу ефективність отриманих вирішальних правил для визначення координат у просторі та розпізнавання перешкод на основі відеопотоку.

Подяки

Робота виконана на базі лабораторії інтелектуальних систем кафедри комп'ютерних наук Сумського державного університету при фінансовій підтримці МОН України в рамках держбюджетної науково-дослідної роботи ДР №0117U003934

Список літератури:

- [1] Moskalenko, V.; Moskalenko, A.; Korobov, A.; Boiko, O.; Martynenko, S.; Borovenskyi, O. Model and Training Methods of Autonomous Navigation System for Compact Drones. Proceedings of the 2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP), Lviv, Ukraine, 21–25 August 2018; Lviv Polytechnic Publishing House: Lviv, Ukraine, 2018; pp. 503 – 508.
- [2] Wang, S.; Deng, Z.; Yin G. An Accurate GPS-IMU/DR Data Fusion Method for Driverless Car Based on a Set of Predictive Models and Grid Constraints. Sensors 2016, 16(3), pp. 280–293, DOI:10.3390/s16030280. Available online: <https://www.mdpi.com/1424-8220/16/3/280>.
- [3] Suwandi, B.; Kitasuka, T.; Aritsugi, M. Low-cost IMU and GPS fusion strategy for apron vehicle positioning. Proceedings of the 2017 IEEE Region 10 Conference (TENCON), Penang, Malaysia, 5–8 Nov. 2017; pp. 449–454.
- [4] Mary, B. A.; Gerhard, P. H. Pose Estimation of a Mobile Robot Based on Fusion of IMU Data and Vision Data Using an Extended Kalman Filter. Sensors (Basel) 2017, 17(10), 2164, DOI:10.3390/s17102164. Available online: <https://www.mdpi.com/1424-8220/17/10/2164> (accessed on 04 November 2018).
- [5] Folkesson, J.; Leederkerken, J.; Williams, R.; Patrikakis, A.; Leonard J. A. A Feature Based Navigation System for an Autonomous Underwater Robot. Proceedings of the sixth edition of Field and Service Robotics, Chamonix, France, 9–12 July 2007; Laugier C., Siegwart R. Eds; Springer Tracts in Advanced Robotics Springer: Berlin, Germany, 2008; pp.105–114, DOI:10.1007/978-3-540-75404-6_10. Available online: https://link.springer.com/chapter/10.1007/978-3-540-75404-6_10 (accessed on 04 November 2018).
- [6] Cadena, C.; Carlone, L.; Carrillo, H.; Latif, Y.; Scaramuzza, D.; Neira, J.; Reid, I.; Leonard, J. Past, Present, and Future of Simultaneous Localization And Mapping: Towards the Robust-Perception Age. IEEE Transactions on Robotics 2016, 32(6), 1309–1332, DOI: 10.1109/TRO.2016.2624754.
- [7] Dorian, G.-L.; Marta, S.; Juan, D. T.; Montiel, J.M.M. Real-time monocular object SLAM. Robotics and Autonomous Systems 2016, 75, 435–449, DOI:10.1016/j.robot.2015.08.009.
- [8] Ayoul, Th.; Buckley, T.; Crevier, F.; UAV Navigation above Roads Using Convolutional Neural Networks. Available online: <http://cs231n.stanford.edu/reports/2017/pdfs/553.pdf> (accessed on 04 November 2018)
- [9] Mohanty, V. DeepVO: A Deep Learning approach for Monocular Visual Odometry. Available online: <https://arxiv.org/pdf/1611.06069.pdf> (accessed on 04 November 2018).
- [10] Labusch, K.; Barth, E.; Martinetz, T. Sparse Coding Neural Gas: Learning of Overcomplete Data Representations. Neurocomputing 2009, 72(7–9), 1547–1555, DOI: 10.1016/j.neucom.2008.11.027.
- [11] Zou, W.; Xia, Y.; Li, H. Fault Diagnosis of Tennessee-Eastman Process Using Orthogonal Incremental Extreme Learning Machine Based on Driving Amount, IEEE Transactions on Cybernetics 2018, 1–8. DOI: 10.1109/TCYB.2018.2830338.
- [12] Москаленко, В. В.; Коробов, А. Г. Інформаційно-екстремальний алгоритм функціонування системи розпізнавання об'єктів на місцевості з оптимізацією параметрів екстрактора ознак. Радіоелектроніка, інформатика, управління 2017, 2, 38–45, 10.15588/1607-3274-2017-2-7.
- [13] Vens, C.; Costa, F. Random Forest Based Feature Induction. Proceedings of the IEEE 11th International Conference on Data Mining, Vancouver, Canada, 11–14 Dec, 2011; Publisher: Piscataway, NJ, 2011; pp. 744–753. DOI: 10.1109/ICDM.2011.121. Available online: <https://ieeexplore.ieee.org/document/6137279>
- [14] Moskalenko, V.; Dovbysh, S.; Naumenko, I.; Moskalenko, A.; Korobov, A. Improving the effectiveness of training the on-board object detection system for a compact unmanned aerial vehicle. Eastern-European Journal of Enterprise Technologies 2018, 4/9 (94), pp. 19–26. DOI: 10.15587/1729-4061.2018.139923

Надійшло до редакції 29.10.2018

УДК 004.032

С.Г. Удовенко¹, Д.В. Келембет²¹ ХНЕУ, м. Харків, Україна, serhiy.udovenko@hneu.net;² ХНУРЕ, м. Харків, Україна, kelembet@gmail.com

КЕРУВАННЯ АГРЕГАТАМИ КОМПРЕСОРНОЇ СТАНЦІЇ З ВИКОРИСТАННЯМ НЕЧІТКИХ ІНВЕРСНИХ МОДЕЛЕЙ

Розглянуто можливість і доцільність застосування для управління агрегатами компресорної станції нечітких інверсних моделей. Запропоновано модифікований алгоритм побудови таких моделей. Сутність модифікації полягає в декомпозиції глобальної нечіткої MISO-системи на елементарні підсистеми і попередньої інверсії елементарних комірок нечіткої моделі з подальшим переходом до інверсії нечіткої глобальної моделі.

НЕЧІТКА ІНВЕРСНА МОДЕЛЬ, КОМПРЕСОРНА СТАНЦІЯ, ДЕКОМПОЗИЦІЯ, НЕЧІТКИЙ РЕГУЛЯТОР

С.Г. Удовенко, Д.В. Келембет. Управление агрегатами компресорной станции с применением нечетких инверсных моделей. Рассмотрена возможность и целесообразность применения для управления агрегатами компресорной станции нечетких инверсных моделей. Предложен модифицированный алгоритм построения таких моделей. Суть модификации заключается в декомпозиции глобальной нечеткой MISO-системы на элементарные подсистемы и предварительной инверсии ячеек нечеткой модели с переходом к инверсии нечеткой глобальной модели.

НЕЧЕТКАЯ ИНВЕРСНАЯ МОДЕЛЬ, КОМПРЕССОРНАЯ СТАНЦИЯ, ДЕКОМПОЗИЦИЯ, НЕЧЕТКИЙ РЕГУЛЯТОР

S.G. Udovenko, D.V. Kelembet. Control of compressor station units using a fuzzy inverse model. The possibility and expediency of application for control of aggregates of the compressor station of fuzzy inverse models is considered. A modified algorithm for constructing such models is proposed. The essence of the modification is the decomposition of the global fuzzy MISO-system on the elementary subsystem and the previous inversion of the elementary cells of the fuzzy model with the subsequent transition to the inversion of the fuzzy global model.

FUZZY INVERSE MODEL, COMPRESSOR STATION, DECOMPOSITION, FUZZY REGULATOR

Вступ

Оптимальний режим експлуатації магістральних газопроводів визначається роботою компресорних станцій. Завдання управління технологічними процесами (ТП) компресорної станції (КС) складаються в регулюванні витрати, тиску і температури газу. Управління газоперекачуючими агрегатами (ГПА) здійснюється автономними автоматичними системами регулювання. Існують регулятори тиску газу, багатоканальні вимірники і регулятори температури. В даний час регулювання здійснюється переважно з використанням цифрових ПІД-регуляторів [1]. Для організації ефективного функціонування систем управління компресорною станцією доцільно застосовувати системи з елементами штучного інтелекту [2]. Модель нечіткого керування компресорною станцією, як і модель будь-якої системи управління з нечітким описом параметрів, будується на основі формалізації суб'єктивних знань експертів.

Розглянемо можливість і доцільність застосування для управління ТП компресорної станції моделей з нечітким описом вихідних даних і нечітким логічним висновком. Режим роботи компресорної станції залежить від типів, схеми з'єднання, оборотів ГПА, температури і складу газу.

Припустимо, що існує інверсна система, здатна відповідно до заданої бажаної траєкторією

виробляти керуючий сигнал, який дозволяє домогтися збіжності керованого виходу ГПА (заданого тиску газу) до бажаної траєкторії. В цьому випадку синтез регулятора можна звести до проблеми визначення інверсної системи. Ідея використання інверсних моделей для класу лінійних і нелінійних систем отримала розвиток в роботах [3, 4]. Це призводить до необхідності розглянути можливість використання нечітких систем Такагі-Сугено як інверсних моделей ТП КС. Однак при наявності невизначеностей і збурень реалізація подібної схеми управління у відкритому контурі може привести до низької якості роботи, що синтезується системи і навіть до нестійким режимам. Метою даної роботи є дослідження можливості і доцільності використання інверсних нечітких компенсаційних регуляторів в замкнених системах цифрового керування з внутрішньою нечіткою моделлю, що відбиває динамічні властивості технологічних процесів компресорної станції газоперекачувальної системи.

1. Характеристика задач керування компресорною станцією

Призначення лінійної КС – забезпечення транспорту газу через кожну чергову ділянку газопроводу, до наступної КС. Лінійні КС використовують газоперекачувальні агрегати (ГПА) з газотурбінним

або електромоторним приводом і працюють у режимі стабілізації вихідного тиску. Кожна станція повинна компенсувати втрати тиску на попередній лінійній ділянці, що виникають через тертя газу по стінках труби, відновлюючи тиск на виході КС до рівня уставки, яка задається оперативно-диспетчерською службою.

Лінійна КС, стабілізуючи тиск на вході лінійної ділянки, тим самим не допускає поширення хвиль тиску (результат дії збурень) на інші ділянки МГ, підвищуючи пропускну здатність МГ і знижуючи непродуктивні втрати потужності в агрегатах інших КС, викликані змінами їх частот обертання. Проаналізуємо коротко режим газопроводу, використовуючи газодинамічні залежності, що описують встановлене протікання газу у достатньо протяжній трубі. Витрата газу q , млн.м³/доб, через газопровід довжиною L , км, визначається формулою

$$q = 0,3 \cdot 10^{-6} \cdot D^{2,5} \sqrt{(P_{\text{вих}}^2 - P_{\text{вх}}^2) / (\lambda \Delta T_{\text{cp}} LZ)}$$

де: $P_{\text{вих}}$, $P_{\text{вх}}$ – початковий та кінцевий тиск на ділянці МГ, кгс/см²; $\Delta = R_{\text{в}}/R_{\text{г}}$ – відносна густина газу (за повітрям); $R_{\text{в}}$, $R_{\text{г}}$ – газова стала повітря та газу при атмосферному тиску; λ – коефіцієнт гідравлічного опору; T_{cp} – середня температура газу на даному МГ, К; D – діаметр труби, мм; Z – коефіцієнт стиснення газу.

Значення сумарної потужності КС для конкретного МГ дозволяє вибрати необхідне число ГПА (з урахуванням необхідного резерву). Спрощений графік розподілу тиску за довжиною газопроводу наведено на рис. 1. Функція, що описує залежність падіння тиску від відстані до КС, близька до лінійної (максимальне відхилення Δ не перевищує 7%).

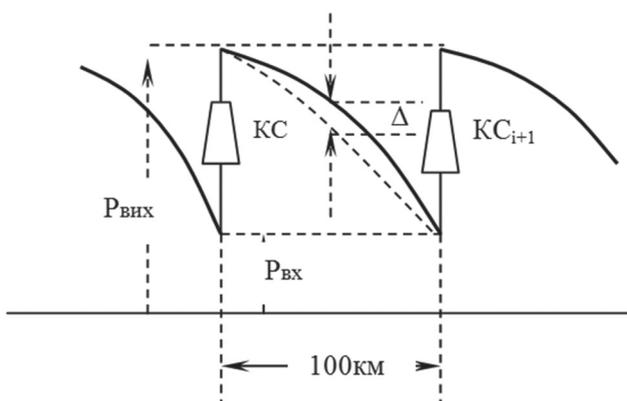


Рис. 1. Розподіл тиску за довжиною МГ

При керуванні технологічним режимом необхідно враховувати специфіку КС як об'єкта керування. Вона полягає в наступному:

– зміна частоти обертання одного з ГПА викликає зміну витрати та потужності всіх ГПА КС (тобто працює схема “кожний впливає на всіх”, (рис. 2);

– коефіцієнти взаємовпливу агрегатів невідомі і непостійні, залежать від режиму ГПА і фактичних витратно-напірних характеристик нагнітачів;
– взаємовплив реалізується з запізненням (вплив відстані між агрегатами “за газом”).

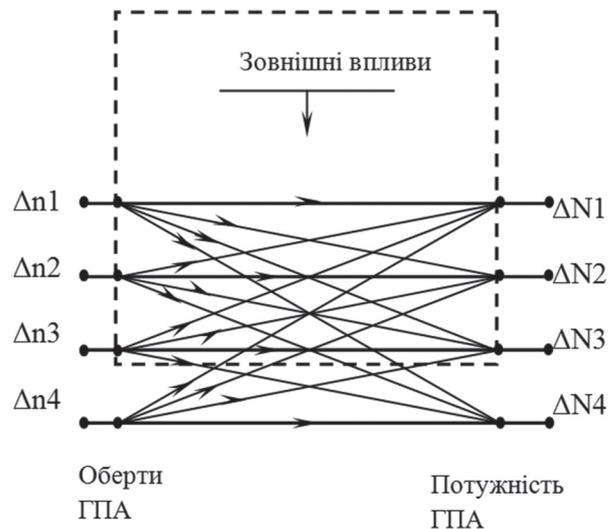


Рис. 2. Залежність потужності КС від обертів ГПА

Після зміни завдання з вихідного тиску оператор для підвищення тиску покроково збільшує частоти обертання агрегатів та оцінює після закінчення перехідних процесів для кожного ГПА запаси регулювання за параметрами САК ГПА, які контролюються. При практичному обнулінні сигналу неузгодженості за будь-яким параметром ГПА, подальше навантаження цього ГПА неможливо, тобто необхідно перерозподіляти навантаження між агрегатами, або вводити резервний агрегат, або змінити уставку вихідного тиску.

Цей процес зручно аналізувати та ілюструвати з використанням режимної площини, що має координати – витрата (Q) та ступінь стиснення (ϵ). Особливість цієї площини – кожній її точці відповідає єдине значення корисної потужності $N_{\text{п}}$, яка віддається ГПА у зовнішню мережу.

Виділимо ту частину площини ϵ , Q , усі точки якої відповідають режимам, при яких параметри ГПА, що функціонально зв'язані з потужністю, не досягають рівнів обмеження, тобто виконується умова

$$N_{\text{min}} < N < N_{\text{max}}$$

Цю частину режимної площини називають областю допустимих режимів (ОДР) агрегату. Оскільки ГПА складається з газотурбінної установки (ГТУ) та відцентрового нагнітача (ВЦН), ОДР ГПА може бути отримана накладенням областей припустимих режимів ГТУ та ЦБН і виділенням тих ділянок, що одночасно належать зазначеним ОДР (рис. 3).

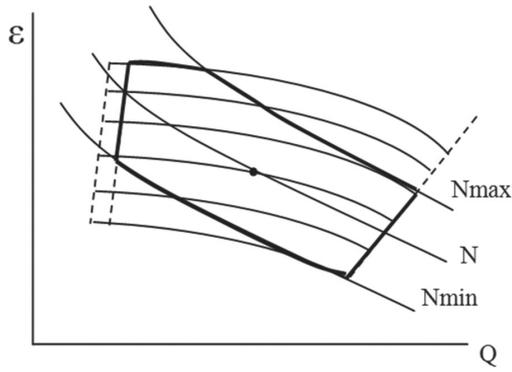


Рис. 3. Розподіл тиску за довжиною МГ

Таким чином, в системах керування агрегатами КС мають враховуватися нелінійності, що згідно з рис. 1 та рис. 3 є притаманними режимам роботи компресорних станцій.

Кінцева мета створення системи керування технологічними процесами транспортування газу полягає у підвищенні ефективності роботи газотранспортної системи за рахунок оперативного планування, централізованого контролю і управління режимами роботи основних технологічних об'єктів газопроводу в реальному часі.

Зокрема, завдання керування технологічними процесами (ТП) компресорної станції складаються в регулюванні витрати, тиску і температури газу. Управління газоперекачувальними агрегатами здійснюється автономними автоматичними системами регулювання, де застосовуються регулятори тиску газу, багатоканальні вимірники і регулятори температури. В даний час регулювання здійснюється переважно з використанням ПІ та ПІД-регуляторів [5].

Для організації ефективного функціонування систем керування компресорною станцією доцільно застосовувати системи з елементами штучного інтелекту. Схема нечіткого керування компресорною станцією, як і модель будь-якої системи керування з нечітким описом параметрів, будується на основі формалізації суб'єктивних знань експертів [6].

Визначення нелінійної моделі об'єктів цифрового керування може бути важким завданням, особливо для складних процесів (зокрема, для ТП КС). Альтернатива вирішення цієї проблеми полягає у використанні апроксимуючих можливостей нечітких систем для подання нелінійних динамічних процесів. Синтез регулятора при цьому може базуватися на інверсії нечіткої моделі.

2. Принцип інверсії та декомпозиції нечіткої моделі

Для цифрового керування перспективним є застосування нечітких систем Такагі-Сугено з постійним набором правил, що дозволяють отримувати формалізоване уявлення про чисельні процедур. Розглянемо систему Такагі-Сугено

\sum_f^n з n -вимірним входом (e_1, \dots, e_n) і скалярним виходом s (MISO - систему). Позначення \sum_f^n відповідає постійному набору наступних правил:

$R^{(i_1, \dots, i_n)}$: якщо e_1 дорівнює $A_1^{i_1}$ і ... і e_n дорівнює $A_n^{i_n}$, $s = \varphi(i_1, \dots, i_n)$, де $A_k^{i_k}$ ($k = \overline{1, n}$) – i -й символ, пов'язаний зі входом e_k ; $\varphi(i_1, \dots, i_n)$ – чисельне значення виходу, пов'язане з набором (i_1, \dots, i_n) .

Рис. 4 ілюструє принцип інверсії MISO-моделі Такагі-Сугено, застосовуваний для ідентифікації нелінійних систем.



Рис. 4. Принцип інверсії нечіткої моделі

Згідно з рис. 4, якщо існує відповідність виходу (Out) і входів (I_n) нечіткої моделі, то можна поставити задачу відновлення входів (I'_n) по відомим виходам нечіткої моделі.

В роботі [3] пропонується метод інверсії нечіткої SISO-моделі. Поставимо задачу модифікації цього методу стосовно до задачі цифрового нечіткого MISO-керування. Суть модифікації полягає в декомпозиції глобальної нечіткої MISO-системи на елементарні підсистеми і попередньої інверсії елементарних осередків нечіткої моделі з подальшим переходом до інверсії нечіткої глобальної моделі. Розглянемо процедуру декомпозиції глобальної нечіткої системи на підсистеми на прикладі системи Такагі-Сугено \sum_f^2 з двома входами e_1 і e_2 , задану набором правил $R^{(i_1, i_2)}$ наступного вигляду: якщо e_1 дорівнює $A_1^{i_1}$ і e_2 дорівнює $A_2^{i_2}$, то $S = \varphi(i_1, i_2)$.

Вихід, що генерується такою системою, дорівнює:

$$s = \sum_{(i_1, i_2) \in I^2} \xi^{(i_1, i_2)}(e) \varphi(i_1, i_2); e = [e_1, e_2] \text{ і } I^2 = I_1 \times I_2$$

Ця система визначена на наступній множині:

$$E^2 = E_1 \times E_2 = [\alpha_1^{e_1}, \beta_1^{e_1}] \times [\alpha_2^{e_2}, \beta_2^{e_2}], \quad (1)$$

де $\alpha_p^{e_p}, \beta_p^{e_p}; p = \overline{1, 2}$ – відповідно мінімальне і максимальне значення інтервалів, що визначають елементи підмножин E_p .

Значення E_1 та E_2 можуть бути декомпозовані наступним чином:

$$E_1 = [\alpha_1^{e_1}, \beta_1^{e_1}] = \bigcup_{i_1=1}^{N_1-1} [a_1^{i_1}, a_1^{i_1+1}], \quad (2)$$

$$E_2 = [\alpha_2^{e_2}, \beta_2^{e_2}] = \bigcup_{i_2=1}^{N_2-1} [a_2^{i_2}, a_2^{i_2+1}]$$

где $a_1^{i_1}$ и $a_2^{i_2}$ — модальные значения нечетких символов $A_1^{i_1}$ и $A_2^{i_2}$.

Очевидно, що:

$$E_{(i_1, i_2)}^2 = [a_1^{i_1}, a_1^{i_1+1}] \times [a_2^{i_2}, a_2^{i_2+1}]; (i_1, i_2) \in I_*^2, \quad (3)$$

$$I_*^2 = \{(i_1, i_2) | i_p = 1, \dots, N_p - 1; p = 1, 2\}.$$

Кожне значення $E_{(i_1, i_2)}^2$ відповідає прямокутній елементарній нечіткій комірці, для якої активізуються 4 нечітких правила. В цьому випадку область E^2 утворюється об'єднанням двох комірок, що їй відповідають:

$$E^2 = \bigcup_{(i_1, i_2) \in I_*^2} E_{(i_1, i_2)}^2. \quad (4)$$

Таким чином, глобальна нечітка система, що розглядається, складається з $(N_1 - 1) \times (N_2 - 1)$ нечітких підсистем $\sum_f^{(i_1, i_2)}$, кожна з яких визначається сукупністю наступних правил:

$$s_{(i_1, i_2)} = \sum_{(v_1, v_2) \in \{0,1\}^2} (e) \phi(i_1 + v_1, i_2 + v_2). \quad (5)$$

Кожна нечітка підсистема $\sum_f^{(i_1, i_2)}$ визначається однією елементарною коміркою $E_{(i_1, i_2)}^2$, що містить 2^2 вузлів.

Таким чином, вихід, що генерується глобальною нечіткою системою, може бути однозначно представлений для сукупності входів, що відповідають кожній елементарній комірці $E_{(i_1, i_2)}^2$, за допомогою правил (5).

Звідси випливає, що в кожен момент часу для будь-якого входу вихід глобальної системи дорівнює виходу, що генерується підсистемою, яка визначається відповідною коміркою.

Узагальнюючи отримані результати на глобальну MISO - систему з n входами, можна показати, що в цьому випадку кожна підсистема визначається набором 2^n правил $R_{(i_1, \dots, i_n)}^{(v_1, \dots, v_n)}$: наступного вигляду: якщо e_1 дорівнює $A_1^{i_1+v_1}$ і e_n дорівнює $A_n^{i_n+v_n}$, то:

$$s_{(i_1, \dots, i_n)} = \phi(i_1 + v_1, \dots, i_n + v_n); \quad (6)$$

$$v_p = 0, 1; p = 1, \dots, n.$$

Таким чином, можна зробити висновок, що кожному входу глобальної нечіткої системи за допомогою набору правил типу (6) можна однозначно поставити у відповідність вихід нечіткої підсистеми.

Розглянемо як приклад нечітку модель динамічного керованого процесу, що представлена в [7]. Вона описується системою Такагі-Сугено, що базується на наборі правил $R^{(i_1, i_2)}$ наступного

вигляду: якщо вихід $y(k)$ дорівнює $A_1^{i_1}$ і керування $U(k)$ дорівнює $A_2^{i_2}$, то $S = \phi(i_1, i_2)$.

$$y(k+1) = \phi(i_1, i_2); i_1 = \overline{1,4}; i_2 = \overline{1,4}. \quad (7)$$

Повний набір правил системи Такагі-Сугено наведено в табл. 1. У цьому прикладі виходи e_1 і e_2 позначені відповідно як $y(k)$ і $u(k)$, а вихід системи S як $y(k+1)$.

Таблиця 1

Набір правил нечіткої системи (7)

$y \setminus u$	A_1^1	A_1^2	A_1^3	A_1^4
A_1^1	1.06	1.05	1.07	1.11
A_1^2	1.13	1.19	1.23	1.33
A_1^3	1.46	1.51	1.63	1.79
A_1^4	1,84	2.03	2.12	2.34

Декомпована нечітка система складається з 9 нечітких підсистем, що визначаються за допомогою 9 нечітких елементарних комірок, як це показано на рис.5. Вихід кожної з цих 9 підсистем визначається наступною залежністю:

$$s_{(i_1, i_2)} = \sum_{(v_1, v_2) \in \{0,1\}^2} \xi^{(i_1+v_1, i_2+v_2)} * \phi(i_1 + v_1, i_2 + v_2); \quad (8)$$

де $i_1 = \overline{1,3}$; $i_2 = \overline{1,3}$.

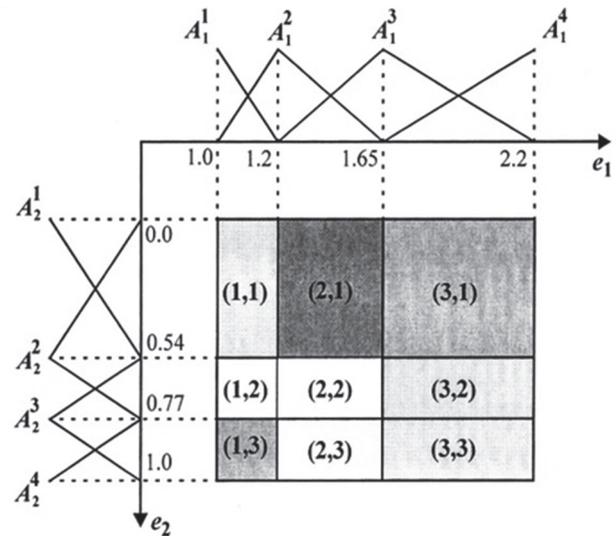


Рис. 5. Елементарні комірки нечіткої системи (7)

Наприклад, вихід підсистеми, яка визначається елементарною коміркою (1.1), відповідає такому керуванню:

$$s_{(1,1)} = \Psi^{(1,1)} [y(k), u(k)] = \sum_{(v_1, v_2) \in \{0,1\}^2} \xi^{(1+v_1, 1+v_2)} \phi(1+v_1, 1+v_2) = \quad (9)$$

$$= (0.350y(k) + 0.710) + (0.648y(k) - 0.666)u(k)$$

Виходи інших підсистем наведені в таблиці 2.

Очевидно, що кожен з цих виходів визначається однозначно відповідним правилом залежності (8).

Таблиця 2

Виходи підсистем нечіткої системи (7)

Нечітка комірка	Вихід нечіткої підсистеми
(1,1)	$0.710+0.350 y(k)+$ $(-0.666+0.648 y(k)) u(k)$
(1,2)	$0.537+0.465 y(k)+$ $(-0.347+0.434 y(k)) u(k)$
(1,3)	$1.140-0.204 y(k)+(-1.130+1.304$ $y(k)) u(k)$
(2,1)	$0.250+0.733 y(k)+(0.160-0.041$ $y(k)) u(k)$
(2,2)	$0.743+0.293 y(k)+$ $(-0.753+0.722 y(k)) u(k)$
(2,3)	$0.364+0.422 y(k)+$ $(-0.260+0.579 y(k)) u(k)$
(3,1)	$0.320+0.690 y(k)+$ $(-0.685+0.471 y(k)) u(k)$
(3,2)	$-0.543+1.073 y(k)+(0.913-0.237$ $y(k)) u(k)$
(3,3)	$0.226+0.525 y(k)+$ $(-0.086+0.474 y(k)) u(k)$

3. Алгоритм інверсії нечіткої моделі

Для отримання стратегії інверсії елементарних нечітких комірок важливою є правильна інтерпретація нечітких символів відповідних правил вибору. Для системи з двома входами кожне правило записується у наступному вигляді: «Якщо e_2 відповідає $A_2^{i_2}$, то $S_{(i_1, i_2)} = \varphi_{(i_1, i_2)}(i_2, e_1)$ ». Це правило можна інтерпретувати в такий спосіб: «Якщо вхід системи дорівнює $A_2^{i_2}$, то вихід системи буде дорівнювати $\varphi_{(i_1, i_2)}(i_2, e_1)$ ».

Інверсія цього правила відповідає такому твердженню: «Якщо потрібно, щоб вихід системи дорівнював $\varphi_{(i_1, i_2)}(i_2, e_1)$, то достатньо подати на вхід системи сигнал $A_2^{i_2}$ ». Звідси випливає наступне правило: «Якщо $S_{(i_1, i_2)}$ відповідає $C^{\varphi_{(i_1, i_2)}(i_2, e_1)}$, то $e_2 = \text{Mod}(A_2^{i_2}) = a_2^{i_2}$ », тобто маємо трикутну функцію належності з модальним значенням $\varphi_{(i_1, i_2)}(i_2, e_1)$. Значимо, що модальне значення $\text{Mod}(A_2^{i_2})$ відповідає нечіткому значенню $A_2^{i_2}$.

Можна показати, що при виконанні наведеного нечіткого правила (13) слушним є такий набір інверсних правил: «Якщо $S_{(i_1, i_2)}$ відповідає $C^{\varphi_{(i_1, i_2)}(i_2, e_1)}$, то $e_2 = a_2^{i_2}$; якщо $S_{(i_1, i_2)}$ відповідає $C^{\varphi_{(i_1, i_2)}(i_2+1, e_1)}$ то $e_2 = a_2^{i_2+1}$ ».

Інверсія, здійснювана на наведеній схемі, дозволяє перейти до етапу побудови глобальної нечіткої інверсної моделі. Для інверсії глобальної нечіткої системи, представленої сукупністю елементних нечітких підсистем, необхідно здійснити інверсію кожної з цих підсистем. Це завдання може мати безліч рішень, серед яких необхідно відповідно до деякого критерію вибрати найкраще.

З практичної точки зору важливо також визначити, чи всі підсистеми необхідно інвертувати або деякі з них можна виключити з розгляду. Обчислення

за розглянутим вище алгоритмом припускають, що для кожної підсистеми, що містить одну вхідну змінну, відомі значення інших $(n-1)$ входів.

Іншими словами, для інверсії системи по входах e_n мають бути визначені індекси комірок i_1, i_2, \dots, i_{n-1} . Нехай ці індекси дорівнюють $i_1^*, i_2^*, \dots, i_{n-1}^*$. Інверсія в цьому випадку передбачає розгляд підсистем $(i_1^*, i_2^*, \dots, i_{n-1}^*, i_n)$ для $i_n \in [1, N_n - 1]$, де N_n – число символів, що використовуються для створення змінної e_n . Таким чином, якщо $N_n = 2$, необхідно інвертувати тільки одну комірку. У загальному випадку, число інвертованих підсистем зменшиться до $N_n - 1$.

Визначимо умову існування єдиного рішення завдання побудови інверсної нечіткої моделі. Нехай вихідна система з n входами трансформована до нечітких SISO- підсистем згідно з набором наступних N_n правил: «Якщо e_n відповідає A_n^i , то $S_{(i_1^*, \dots, i_{n-1}^*, i)} = \varphi_{(i_1^*, \dots, i_{n-1}^*, i)}(1, z_1)$; якщо e_n відповідає A_n^2 , то $S_{(i_1^*, \dots, i_{n-1}^*, i)} = \varphi_{(i_1^*, \dots, i_{n-1}^*, i)}(2, z_1)$; якщо e_n відповідає A_n^n , то $S_{(i_1^*, \dots, i_{n-1}^*, i)} = \varphi_{(i_1^*, \dots, i_{n-1}^*, i)}(n, z_1)$, де $Z_1 = (e_1, \dots, e_{n-1})$ ».

Набір цих N_n правил будемо вважати монотонним, якщо виконуються умови:

$$\varphi_{(i_1^*, \dots, i_{n-1}^*, 1)}(1, Z_1) < \varphi_{(i_1^*, \dots, i_{n-1}^*, 2)}(2, Z_1) < \dots < \varphi_{(i_1^*, \dots, i_{n-1}^*, N_n)}(N_n, Z_1) \quad (10)$$

або

$$\varphi_{(i_1^*, \dots, i_{n-1}^*, 1)}(z, Z_1) > \varphi_{(i_1^*, \dots, i_{n-1}^*, 2)}(2, Z_1) > \dots > \varphi_{(i_1^*, \dots, i_{n-1}^*, N_n)}(N_n, Z_1).$$

У разі монотонності набору правил рішення задачі інверсії є єдиним. Дійсно, якщо S^* – значення виходу, для якого визначається інверсне рішення, то існує лише одне значення $i_n \in [1, N_{n-1}]$, для якого виконується нерівність:

$$\varphi_{(i_1^*, \dots, i_{n-1}^*, i_n)}(i_n, Z_1) \leq s^* \leq \varphi_{(i_1^*, \dots, i_{n-1}^*, i_{n+1})}(i_n + 1, Z_1).$$

Звідси випливає, що тільки підсистема $(i_1^*, \dots, i_{n-1}^*, i_n)$ може генерувати вихід S^* , який необхідно інвертувати.

Для розглянутої вище нечіткої системи (7) нечітка комірка (1,1) моделі може бути трансформована в SISO-модель за наступними правилами: «Якщо $U(k)$ відповідає A_2^1 , то $S_{(1,1)} = \varphi_{(1,1)}(2, y(k)) = 0.7y(k) + 0.35$ ».

Побудуємо набір інверсних правил для моделі (10), виходячи з того, що вихід моделі $y(k+1)$ повинен бути рівним бажаному значенню $y_{des}(k+1)$. Відповідно до запропонованої вище процедури інверсії цей набір прийме наступний вигляд: «Якщо $y_{des}(k+1)$ відповідає $C^{\varphi_{(1,1)}(1, y(k))}$, то $C^{\varphi_{(1,1)}(1, y(k))} = 0,0$; якщо $y_{des}(k+1)$ відповідає $C^{\varphi_{(1,1)}(2, y(k))}$, то $U(k) = 0,54$ ».

Вихід відповідної системи визначається наступним чином:

$$u(k) = \frac{(0.350y(k) + 0.710)}{(0.648y(k) - 0.666)} + \frac{y_{des}(k+1)}{(0.648y(k) - 0.666)}, \quad (11)$$

де $y(k) \neq 1.0278$.

Узагальнення отриманих результатів на систему з n входами є тривіальним, оскільки форма SISO-моделі нечіткої підсистеми залишається незмінною для будь-якого числа входів глобальної системи.

4. Аналіз результатів моделювання

Для моделювання системи нечіткого керування ГПА КС з використанням інверсної моделі використовувалися можливості програмного середовища MATLAB, зокрема, пакета для моделювання динамічних систем Simulink і пакета синтезу нечітких систем управління Fuzzy Logic Toolbox.

Для різних діапазонів вихідного тиску (або ступеня стиснення газу) динамічні властивості ГПА можуть змінюватися (через нелінійність статичної залежності по каналу «швидкість обертів вала ГПА - вихідний тиск»). У зв'язку з цим було проведено поділ загального діапазону зміни вихідного тиску на 5 інтервалів, що дозволило після вибору функцій належності сформувати 5 лінійних підсистем (і, відповідно, 5 елементарних нечітких комірок). Правила визначення виходів для кожної з цих підсистем наведені в табл. 1.

Таблиця 3

Таблиця виходів підсистем моделі

№ п/с	Комірка	Вихід $y(k+1)$
1	(1,1)	$0,22y(k) + 0,87u(k) + 1,53$
2	(2,1)	$-0,18y(k) + 0,87u(k) + 1,1$
3	(3,1)	$-0,48y(k) + 0,87u(k) + 0,92$
4	(4,1)	$0,17y(k) + 0,87u(k) + 0,41$
5	(5,1)	$0,12y(k) + 0,87u(k) + 0,46$

Загальна схема моделювання системи цифрового керування ГПА з нечітким регулятором за допомогою засобів пакету Simulink представлена на рис. 6. У запропонованому цифровому SISO-регуляторі реалізований описаний вище метод нечіткого керування динамічним процесом, заснований на використанні інверсії внутрішньої нечіткої моделі.

Результати моделювання роботи системи при стрибкоподібній зміні уставки від 8,0 МПа до 8,5 МПа (діапазон елементарної комірки (3,1)) і подальшому її зниженні до 8,0 МПа підтвердив переваги запропонованої схеми нечіткого керування в порівнянні з ПІД-керуванням. Зміна режиму роботи ГПА здійснюється на верхньому рівні автоматизованої системи управління технологічними процесами КС.

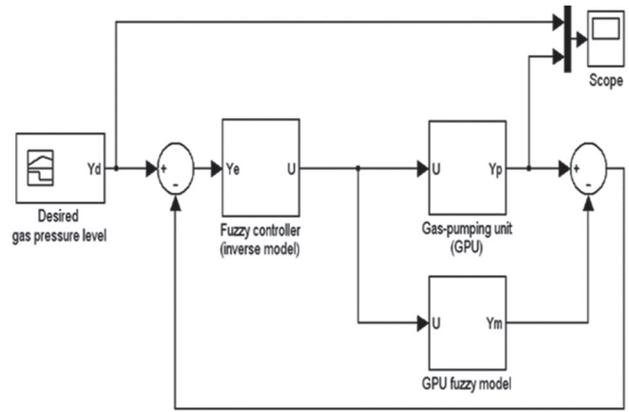


Рис. 6. Схема моделювання системи цифрового керування ГПА з інверсною моделлю

Висновки

У статті був розглянутий підхід до керування газоперекачувальними агрегатами компресорних станцій газотранспортної системи з використанням нечітких регуляторів. Запропоновано структуру MISO - системи цифрового керування, яка базується на застосуванні внутрішніх нечітких інверсних моделей. Для інверсії глобальної нечіткої системи, представлені сукупністю елементарних нечітких підсистем, згідно з наведеним алгоритмом здійснюється інверсія кожної з цих підсистем. В схемі компенсаційного регулювання вихідного тиску ГПА використано інверсну нечітку модель об'єкта керування. Наведено результати моделювання системи нечіткого керування агрегатами компресорної станції, які підтверджують ефективність застосування інверсних нечітких моделей для підтримки заданого тиску газу на виході компресорної станції.

Список літератури:

- [1] Ісаков А.Т., Експлуатація і технічне обслуговування газорозподільних станцій магістральних газопроводів (довідник) / А.Т. Ісаков, М.В. Хохлаєв, Б.С. Фланчик та ін. – Київ: "Росток". – 2003. – 411 с.
- [2] Удовенко С. Интеллектуальное управление компрессорной станцией / С. Удовенко, Д. Келембет // Матеріали XIV міжнародної конференції «Контроль і управління в складних системах (КУСС-2018)», м. Вінниця, 15-17 жовтня 2018 р. – С.21.
- [3] Hao J. Inverse identification for control of nonlinear dynamical systems / J. Hao, J. Vandewalle // Proc. World Congress on Neural Networks '98, Portland 3. – 1998. – P.317–320.
- [4] Удовенко С.Г. Модифицированный метод построения инверсной нечеткой модели объекта цифрового управления / С.Г. Удовенко, Р. Альхайек // Системи управління, навігації та зв'язку. – 2009. – Вип.4(12). – С.130–134
- [5] Боженюк А.В. Нечеткая классификация ситуаций и принятие решений в системах магистрального транспорта / А.В. Боженюк, В.В. Шадрин // Известия ТРТУ – Таганрог: Изд-во ТРТУ. – 2006. – № 10 (65). – С. 9–12.
- [6] Пегат А. Нечеткое моделирование и управление / А. Пегат. – М.: БИНОМ. Лаборатория знаний. – 2011. – 798 с.
- [7] Nauck D. Neuro-Fuzzy Systems for Function Approximation / D. Nauck, R. Kruse // Fuzzy Sets and Systems. – 1999. – Vol. 101, №2. – P. 261-271.
- [8] Коржик М. В. Моделювання об'єктів та систем керування засобами MatLab: / М. В. Коржик. – Київ: НТУУ "КПІ", 2016. – 174 с.

Надійшла до редколегії 05.11.2018

УДК 004.852

О.Г. Руденко¹, О.О. Безсонов², О. Г. Лебедєв³, О. С. Романюк⁴¹ ХНЕУ, м. Харків, Україна, oleg.rudenko@hneu.net;² ХНЕУ, м. Харків, Україна, oleksandr.bezsonov@hneu.net;³ ХНУРЕ, м. Харків, Україна, oleg.lebediev@nure.ua;⁴ ХНУРЕ, м. Харків, Україна, romanyk@gmail.com

КРИТЕРІЇ ВИБОРУ ПЕРСЕПТРОННОЇ МОДЕЛІ ДЛЯ ПРОГНОЗУВАННЯ: АНАЛІЗ І ПРАКТИЧНІ РЕКОМЕНДАЦІЇ ЩОДО ЇХ ВИКОРИСТАННЯ

Аналізуються існуючі і досить поширені статистичні (RMSE, MAPE, MAE та інші) і деякі інформаційні критерії (AIC, KIC, HQ тощо) (метрики) вибору персептронної моделі для прогнозування часових рядів, аналізуються переваги та недоліки цих критеріїв. Крім того, розглядаються запропоновані останнім часом нові критерії, які ще не набули широкого поширення, однак, є такими, що володіють, на думку авторів, істотними перевагами. До числа таких критеріїв належать критерій точності напрямку та його модифікація, спрямована симетрія та її модифікація, зважений інформаційний критерій та його адаптивна модифікація.

Розглянуті критерії вибору моделі аналізуються для прогнозування об'єму пасажиропотоку у аеропортах Німеччини. Отримані результати свідчать про те, що критерії вибору моделі на основі медіани прогнозів мають меншу мінливість, ніж інші, в той час як критерії, що базуються на геометричному середньому, мають більшу мінливість.

ПЕРСЕПТРОННА МОДЕЛЬ, ЧАСОВИЙ РЯД, КРИТЕРІЇ ОЦІНЮВАННЯ, ПОМИЛКА ПРОГНОЗУ, НЕСТАЦІОНАРНІСТЬ, СПРЯМОВАНА СИМЕТРІЯ, СИМЕТРИЧНИЙ КРИТЕРІЙ

Руденко О.Г., Безсонов А.А., Лебедєв О.Г., Романюк А.С. Критерии выбора персептронной модели для прогнозирования: анализ и практические рекомендации по их использованию. Анализируются существующие и достаточно распространенные статистические (RMSE, MAPE, MAE и другие) и некоторые информационные критерии (AIC, KIC, HQ и т.п.) (метрики) выбора персептронной модели для прогнозирования, анализируются преимущества и недостатки этих критериев. Кроме того, рассматриваются предложенные в последнее время новые критерии, еще не получившие широкого распространения, однако, обладающие, по мнению авторов, существенными достоинствами. К числу таких критериев относятся критерий точности направления и его модификация, направленная симметрия и ее модификация, взвешенный информационный критерий и его адаптивная модификация.

Рассмотренные критерии выбора модели анализируются для прогнозирования объема пассажиропотока в аэропортах Германии. Полученные результаты свидетельствуют о том, что критерии выбора модели на основе медианы прогнозов имеют меньшую изменчивость, чем другие, в то время как критерии, основанные на геометрическом среднем, имеют большую изменчивость.

ПЕРСЕПТРОННАЯ МОДЕЛЬ, ВРЕМЕННОЙ РЯД, КРИТЕРИЙ ОЦЕНИВАНИЯ, ОШИБКА ПРОГНОЗА, НЕСТАЦИОНАРНОСТЬ, НАПРАВЛЕННАЯ СИММЕТРИЯ, СИММЕТРИЧНИЙ КРИТЕРИЙ

Rudenko O.G., Bezsonov O.O., Lebediev O.G., Romanyk O.S. Criterias for the selection of the perseptronic model for forecasting: analysys and practical recomendations regarding their usage. Existing and common statistical (RMSE, MAPE, MAE and others) and some information criteria (AIC, KIC, HQ, etc.) for selecting a perceptron model for forecasting are analyzed. Advantages and disadvantages of these criteria are discussed. In addition, the recently proposed new criteria, which have not yet become widespread, but which, according to the authors, have significant advantages are considered. Such criteria include direction accuracy criterion and its modification, directional symmetry and its modification, weighted information criterion and its adaptive modification.

The considered model selection criteria are utilized for prediction the volume of passenger traffic at airports in Germany. The results suggest that the criteria for selecting a model based on the median of the forecasts have less variability than others, while the criteria based on the geometric mean have greater variability.

PERSEPTRONIC MODEL, TIME SERIES, EVALUATION CRITERIA, FORECAST ERROR, NON-STATIONARITY, DIRECTED SYMMETRY, SYMMETRIC CRITERIA

Вступ

Прогнозування часових рядів (ЧР) є важливою науково-технічною проблемою, яка дозволяє передбачити поведінку різних факторів в екологічних, економічних, соціальних та інших системах. Актуальність задачі прогнозування ЧР особливо зростає останнім часом. Це обумовлено доступністю потужних обчислювальних засобів збору та обробки інформації [1–7].

Метою будь-якого прогнозування є створення моделі, яка дозволяє передбачити майбутнє і оцінити тенденції в змінах того чи іншого фактора. Якість прогнозу в такому випадку залежить від наявності передісторії змінюваного чинника, похибок вимірювання даної величини і інших чинників. Для оцінки якості прогнозування використовуються різні критерії: оцінки помилок прогнозування, швидкість розрахунку, інтерпретованість тощо.

Оцінка помилки прогнозування або точності прогнозу є найбільш важливими при вирішенні практичних завдань.

Слід зазначити, що жорсткі статистичні припущення про властивості часових рядів часто обмежують можливості класичних методів прогнозування. Застосування методів обчислювального інтелекту, зокрема штучних нейронних мереж (ШНМ) в даній задачі обумовлено наявністю в більшості ЧР складних закономірностей, апіорної та поточної невизначеності, нестаціонарностей, викидів тощо [7–12].

Інтелектуальні методи прогнозування, як і класичні, потребують побудови математичної моделі, якість якої і обумовлює точність прогнозування. Необхідно зазначити, що при цьому для побудови моделі (вибору її структури та оцінці параметрів) використовуються ті ж критерії, які характеризують точність прогнозування [13–15].

При дослідженні ЧР отримати абсолютно точний прогноз майже неможливо, з цієї причини важливим завданням вважається оцінка різних моделей прогнозування з точки зору певних критеріїв якості. При цьому надійність обраної моделі прогнозування оцінюється періодичним порівнянням фактичних і прогнозованих значень ряду. При вирішенні практичних задач більш значущими вважається аналіз помилки прогнозування або точності прогнозу [16].

Як вже зазначалося, абсолютну точність прогнозу оцінювати не прийнято, тому що для кожного конкретного завдання вона може кардинально відрізнитися, тому оцінюють саме помилку прогнозу, яка характеризується деякою метрикою.

З урахуванням цих труднощів і помилок в прогнозах, виникає необхідність розробки рекомендацій щодо вибору критеріїв оцінки якості прогнозування, зроблених на підставі та розрахунку помилок прогнозування.

Метою роботи є аналіз існуючих і досить поширених оцінок структури нейромережевої моделі і помилок (метрик) прогнозування та розробка рекомендацій щодо їх використання.

1. Багатошаровий перцептрон (БП)

БП представляє собою нейронну мережу з декількома шарами, кожен з яких складається з комп'ютерних вузлів (нейронів) [17–18]. Топологія багатошарового перцептрона показана на рис. 1. Входи мережі підключені до кожного нейрона в першому шарі. Виходи нейронів першого шару потім становляться входами для нейронів другого шару і так далі. Останній шар є вихідним, всі інші шари між вхідними та вихідними шарами називаються прихованими. Архітектура БП може бути

зручно записана як $n_0 - n_1 - \dots - n_l$, де n_0 є розмір вхідного вектора мережі, а n_i , $1 \leq i \leq l$ позначає номери вузлів у відповідних шарах. Таким чином, БП використовує наступну апроксимацію нелінійного оператора:

$$\hat{y}(x) = \hat{f}(x) = f^q \left[(W^q)^T f^{q-1} \left[(W^{q-1})^T f^{q-2} \left[\dots f^1 \left[(W^1 x + b_1)^T \right] \dots \right] \right] \right] + b_q, \quad (1)$$

де W^i — вектор вагових параметрів нейронів i -го шару мережі; $f^i[\cdot]$ — активаційна функція (АФ) i -го шару, b_i — зміщення i -го нейрону.

При використанні ШНМ, як і при традиційному підході, також виникають задачі структурної та параметричної оптимізації, відповідні вибору оптимальної топології мережі і її навчання (налаштування параметрів). Якщо завдання визначення структури є дискретною оптимізаційною (комбінаторною), то пошук оптимальних параметрів здійснюється в безперервному просторі за допомогою класичних методів оптимізації.

Традиційні методи визначення структури мережі полягають або в послідовному її ускладненні шляхом введення нових нейронів і нових зв'язків між ними, або в послідовному її спрощенні, починаючи з деякої досить складної топології.

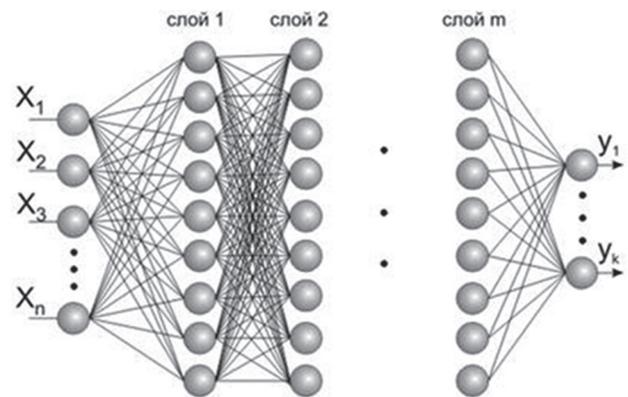


Рис. 1. Топологія багатошарового перцептрона

Навчання мережі полягає у визначенні вектора її параметрів розміру $q \times 1$ і зводиться до мінімізації деякого функціоналу (критерія навчання) від помилок

$$J = e^2(k) = M \left\{ \left[y(k) - \hat{y}(k) \right]^2 \right\} \quad (2)$$

де $y(k)$ — реальна змінна, що прогнозується; $\hat{y}(k)$ — вихідний сигнал мережі.

Більшість відомих в даний час алгоритмів навчання нейронних мереж заснована на використанні жорстких і важко перевірених умов, пов'язаних з гіпотезою нормальності закону розподілу завдань і обґрунтованих посиленнями на

центральною граничною теоремою, і представляє деякі модифікації методу найменших квадратів (МНК).

Задача структурної оптимізації є більш складною і її рішення повинне задовольняти деякому/ деяким критеріям. У зв'язку з тим, що визначення найкращої архітектури ШНМ є надто важливим для рішення поставленої задачі, а на цей час не існує загальних рекомендацій щодо вибору архітектури мережі, слід розглянути існуючі критерії оцінки ефективності та виявити їх переваги та недоліки.

2. Критерії, засновані на абсолютних помилках

Середня помилка прогнозу (Mean Forecast Error, MFE):

$$MFE = \frac{1}{N} \sum_{i=1}^N e_i. \quad (3)$$

Середня абсолютна помилка (Mean Absolute Error, MAE):

$$MAE = \frac{1}{N} \sum_{i=1}^N |e_i| = \text{mean} |e_i| \quad (4)$$

визначає величину помилки прогнозу в цілому.

Як і MFE, дана метрика також залежить від шкали вимірювання і використання перетворень вихідних даних.

У метрики MAE є два недоліки, які проявляються тільки в разі аналізу прогнозу точності моделі:

- 1) її значення важко інтерпретувати;
- 2) за допомогою цієї метрики можна проводити порівняння тільки між моделями з одного ряду даних. На практиці ж часто стоїть завдання зрозуміти, як себе веде модель з кількох рядів даних в порівнянні з іншими моделями. Однак отримані з різних рядів значення MAE складати один з одним не можна.

Середня квадратична помилка (Mean Squared Error, MSE):

$$MSE = \frac{1}{N} \sum_{i=1}^N e_i^2. \quad (5)$$

Дана метрика враховує наявність рідкісних екстремальних помилок.

Залишкова квадратична помилка (Residual Sum of Squares, RSS, Sum of Squared Error, SSE):

$$RSS = SSE = \sum_{i=1}^N e_i^2. \quad (6)$$

Дана метрика має схожі з MSE характеристики.

Знакова середня квадратична помилка (Signed Mean Squared Error, SMSE):

$$SMSE = \frac{1}{N} \sum_{i=1}^N \left(\frac{e_i}{|e_i|} \right) e_i^2. \quad (7)$$

Дана метрика залежить від шкали вимірювання і перетворень вхідних даних і нормалізує отримане значення MSE.

Root Mean Squared Error, RMSE:

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N e_i^2}. \quad (8)$$

Дана метрика має схожі з MSE характеристики.

Корінь четвертого ступеня помилки в четвертому ступені (Fourth Root Mean Quadrupled Error, FRMQE, R4MS4E) [19]:

$$FRMQE = R4MS4E = \sqrt[4]{\sum_{i=1}^N e_i^4}. \quad (9)$$

Нормалізована середня квадратична помилка (Normalized Mean Squared Error, NMSE):

$$NMSE = \frac{MSE}{\sigma^2} = \frac{1}{N\sigma^2} \sum_{i=1}^N e_i^2, \quad (10)$$

де σ — дисперсія прогнозу.

Дана метрика нормалізує отримане значення MSE.

Медіанна абсолютна помилка (Median Absolute Error, MdAE):

$$MdAE = \text{median} |e_i|_{i=1,N} \quad (11)$$

Стандартне відхилення (Standard deviation, SD):

$$SD = \sum_{i=1}^N \sqrt{\frac{(y_i - ME)^2}{N}}. \quad (12)$$

Абсолютні вимірювання помилок мають наступні недоліки:

- 1) залежність від масштабу [4] (тому якщо задача прогнозу включає в себе об'єкти з різними масштабами або величинами, абсолютна помилка в якості метрики не може застосовуватися);
- 2) високий вплив викидів в даних на оцінку ефективності прогнозу [19, 20] (тому якщо дані містять викиди, то абсолютні значення помилок є неефективними).

RMSE, MSE мають низьку надійність (робастність): результати можуть бути різними залежно від різної частки даних [14].

3. Критерії, засновані на обчисленні процентних відношень

Середня абсолютна процентна помилка (Mean Absolute Percentage Error, MAPE):

$$MAPE = \frac{1}{N} \sum_{i=1}^N \left| \frac{e_i}{y_i} \right| \times 100. \quad (13)$$

Як видно, MAPE не має розмірності, з дуже простою інтерпретацією. Її можна вимірювати в частках або відсотках, легко складати з різних рядів. Можна навіть розрахувати MAPE і вивчити його розподіл, використовуючи інструменти статистичного аналізу.

Дана метрика застосовується для часових рядів, фактичні значення яких значно більше 1

(наприклад, оцінки помилки прогнозування енергоспоживання майже у всіх статтях наводяться як значення MAPE). Ця метрика по-різному ставиться до позитивних і негативних помилок: жорсткіше до випадків завищених прогнозів, ніж занижених. У випадку з агрегуванням прогнозів по різних значеннях це призводить до викривлення – дослідник не отримує достовірної інформації про якість його прогнозу.

В результаті детального порівняння MAPE з іншими метриками при виборі моделі прогнозування в роботі [21] показано, що MAPE є ненадійним критерієм при використанні цієї метрики для вибору між конкуруючими прогнозними моделями. Зокрема, існує висока ймовірність того, що MAPE вибере модель, яка дає оцінку нижче середнього (тобто «недооцінює») моделі, що прогнозує середнє значення.

Аналогічний результат отримано в [22], де показано, що MAPE систематично і послідовно високо оцінює моделі, які дають занадто оптимістичний прогноз. Через поділ на фактичні значення ця метрика виявилася чутливою до масштабу, і при малому значенні фактичного у значення MAPE може бути дуже великим.

Середня абсолютна відсоткова помилка типу арктангенс (*Mean Arctangent Absolute Percentage Error (MAAPE)*) [23, 24]:

$$MAAPE = \frac{1}{N} \sum_{i=1}^N AAPE_i, \quad (14)$$

$$\text{де } AAPE_i = \arctg \left(\left| \frac{y_i - \hat{y}_i}{y_i} \right| \right).$$

Ця метрика дозволяє дещо поліпшити властивості MAPE.

Середня процентна помилка (*Mean Percentage Error, MPE*):

$$MPE = \frac{1}{N} \sum_{i=1}^N \left(\frac{e_i}{y_i} \right) \times 100 \quad (15)$$

покликана оцінювати не точність прогнозу, а його зміщення.

Можна помітити, що вона схожа на MAPE, має ті ж переваги і недоліки, що і MAPE, але відрізняється лише відсутністю модулів в чисельнику. MPE показує відсоток зсуву прогнозу. Позитивні значення цього критерію вказують на систематичне заниження прогнозу, а негативні – на завищення. Значення MPE так само, як і MAPE може варіюватися у відсотках.

Дана метрика є досить хорошим показником повної помилки прогнозу, але вона не настільки інтуїтивно зрозуміла, на відміну від інших метрик.

Медіанна абсолютна процентна помилка (*Median Absolute Percentage Error, MdAPE*):

$$MdAPE = \text{median}_{i=1,N} |\bar{e}_i|, \quad (16)$$

де $\bar{e}_i = e_i / y_i$.

Квадратний корінь середньоквадратичної процентної помилки (*Root Median Square Percentage Error, RMSPE*):

$$RMSPE = \sqrt{\text{mean}_{i=1,N} (\bar{e}_i)^2} \quad (17)$$

Квадратний корінь медіанної процентної помилки (*Root Median Square Percentage Error, RMSPE*):

$$RMdSPE = \sqrt{\text{median}_{i=1,N} (\bar{e}_i)^2}. \quad (18)$$

Для даної групи критеріїв характерними є наступні недоліки:

1) з'являється поділ на нуль, коли фактичне значення дорівнює нулю;

2) несиметричність – значення помилки відрізняються, якщо прогнозоване значення більше або менше фактичного [16, 17];

3) викиди значно впливають на результат, особливо, якщо відхилення має значно більшу величину, ніж максимальне значення «нормальних» випадків [15];

4) обчислення помилки є необ'єктивним, що може призвести до неправильної оцінки моделі прогнозування [21].

4. Симетричні критерії

Симетрична MAPE (*Symmetric MAPE, SMAPE*):

$$SMAPE = \frac{1}{N} \sum_{i=1}^N \frac{2|e_i|}{|y_i| + |\hat{y}_i|} \quad (19)$$

використовує середнє арифметичне фактичного і прогнозованого значень, застосовується для часових рядів, фактичні значення яких значно більше 1.

Незважаючи на те, що метрика повинна бути симетричною, але такою вона до кінця не є. SMAPE демонструє зміщення тільки цього разу в бік завищених прогнозів: завищені прогнози призводять до меншої помилки, ніж занижені.

Симетрична медіанна APE (*Symmetric Median APE, SMdAPE*):

$$SMdAPE = \text{median}_{i=1,N} \frac{|y_i - \hat{y}_i|}{|y_i + \hat{y}_i|}. \quad (20)$$

Модифікована MAPE (*Modified sMAPE, msMAPE*):

$$msMAPE = \frac{1}{N} \sum_{i=1}^N \frac{|y_i - \hat{y}_i|}{\frac{|y_i| + |\hat{y}_i|}{2} + s_i}, \quad (21)$$

$$\text{де } s_i = \frac{1}{i-1} \sum_{k=1}^{i-1} |y_i - \bar{y}_k|, \bar{y}_k = \frac{1}{i-1} \sum_{k=1}^{i-1} y_k.$$

Дана група критеріїв має такі недоліки:

- 1) незважаючи на назву, вони є несиметричними [18];
- 2) якщо фактичне значення дорівнює прогнозованому, але з протилежним знаком, або обидва ці значення дорівнюють нулю, тоді відбувається поділ на нуль;
- 3) на значення критеріїв суттєво впливають викиди аналогічно з відсотковими помилками;
- 4) існує проблема інтерпретації результатів.

5. Відносні критерії

Середня відносна абсолютна помилка (*Mean Relative Absolute Error, MRAE*):

$$MRAE = \text{mean} \frac{|e_i|}{(y_i - \hat{y}_i^*)}, \quad (22)$$

де \hat{y}_i^* – прогнозна величина базової (еталонної) моделі (на практиці, як правило, обирають $\hat{y}_i^* = y_{i-1}$).

Медіанна відносна абсолютна помилка (*Median Relative Absolute Error, MdRAE*):

$$MdRAE = \text{median}_{i=1, N} \left| \frac{|e_i|}{(y_i - \hat{y}_i^*)} \right|. \quad (23)$$

Геометрична середня відносна помилка (*Geometric Mean Relative Absolute Error, GMRAE*) [25, 26]:

$$GMRAE = \sqrt[m]{\prod_{i=1}^m \left| \frac{y_i - \hat{y}_i}{y_i - \tilde{y}_i} \right|} \quad (24)$$

де $y_i - \tilde{y}_i$ – помилка за деякою другою моделлю.

GMRAE показує, у скільки разів побудована модель виявилася гірше (або краще) за обрану для порівняння (модель-бенчмарк). Якщо GMRAE > 1, то обрана модель виявилася менш точною, в протилежній ситуації – більш точною. Однак питання про вибір другої моделі залишається. GMRAE не може використовуватися в ситуаціях, коли хоча б одна з помилок однієї з моделей виявилася рівною нулю – в цьому випадку GMRAE стає рівним або нулю або нескінченності.

Виходячи з наведених формул, поділ на нульову помилку відбувається, якщо прогностичне значення отримане за допомогою моделі, дорівнює дійсному значенню.

Якщо обрана наївна модель, то поділ на нуль помилка виникає у випадку безперервної послідовності ідентичні значення часового ряду.

6. Масштабовані помилки

Середня абсолютна масштабована помилка (*Mean Absolute Scaled Error, MASE*):

$$MASE = \frac{T-1}{N} \frac{\sum_{i=1}^N |e_{T+i}|}{\sum_{j=2}^T |y_j - y_{j-1}|}, \quad (25)$$

де T – інтервал прогнозування.

Сума в чисельнику, відповідає тестовій вибірці, а в знаменнику – навчальній. Друга фактично являє собою середню абсолютну помилку прогнозу за методом Naive. Ця ж сума відповідає середньому абсолютному відхиленню ряду в перших різницях. Ця величина, по суті, показує, наскільки навчальна вибірка передбачувана і може дорівнювати нулю тільки в тому випадку, коли всі значення в навчальній вибірці рівні один одному. Це відповідає відсутності будь-яких змін в ряді даних, тобто ситуації, на практиці майже неможливі.

Слід зазначити, що модуль в знаменнику (25) потрібен для ситуацій, коли при прогнозуванні присутні як позитивні, так і негативні значення змінних, середня величина яких може бути близька до нуля. Якщо в такій ситуації брати просто середнє, виникає все та ж проблема масштабу.

Середня абсолютна масштабована помилка (*sMAE - scaled MAE, іноді позначається як MAE / mean*):

$$sMAE = \frac{1}{N} \sum_{i=1}^N \frac{|e_i|}{y_i}. \quad (26)$$

Зауважимо, знаменник sMAE може дорівнювати нулю тільки в одному випадку – якщо всі фактичні значення в навчальній вибірці дорівнюють нулю. Критерій sMAE – практично ідеальний показник. У нього є лише один невеликий недолік – він не обмежений зверху. У загальному випадку значення sMAE буде показувати, який відсоток в середньому становлять помилки прогнозу. Ця величина буде незміщеною (в порівнянні з MAPE і SMAPE) і може легко складатися з іншими такими ж величинами.

Корінь квадратний середньої абсолютної масштабованої помилки (*Root Mean Square Scaled Error, RMSSE*):

$$RMSSE = \sqrt{\text{mean} \left(\frac{|e_i|}{\frac{1}{T-1} \sum_{j=2}^T |y_j - y_{j-1}|} \right)^2}. \quad (27)$$

Відносні критерії є симетричними та стійкими до викидів, однак мають наступні недоліки:

- якщо у прогностичному горизонті реальні значення є рівними, тоді відбувається поділ на нуль;
- можливе невелике зміщення оцінок [23].

7. Критерії, засновані на обчисленні відносних величин

Відносна середня абсолютна помилка (*Relative Mean Absolute Error, RelMAE*):

$$RelMAE = \frac{MAE}{MAE^*}, \quad (28)$$

де MAE і MAE* – середні абсолютні помилки для аналізованої моделі прогнозування та базової

(еталонної) моделі відповідно, які обчислюють за формулою (4).

Відносний квадратний корінь середньої квадратичної помилки (*Relative Root Mean Squared Error, RRMSE*):

$$RRMSE = \frac{RMSE}{RMSE^*}. \quad (29)$$

Логарифм відносного середньоквадратичного відхилення (*Log mean squared error ratio, LMR*):

$$LMR = \log\left(\frac{RMSE}{RMSE^*}\right). \quad (30)$$

Ці критерії визначають кількість випадків, коли аналізована модель прогнозування перевершує базову, але не оцінюють значення різниці.

Недоліками цих критеріїв є можливість поділу на нульову помилку, якщо базова прогнозна помилка дорівнює нулю.

8. Інформаційні критерії

Інформаційний критерій Акаїке (Akaike Information Criteria, AIC) [27]:

$$AIC = \log MSE + \frac{2q}{N}, \quad (31)$$

де q – загальна кількість параметрів моделі; N – кількість спостережень.

Особливості AIC:

1) залежить від припущення нормальності даних;

2) цей критерій не є спроможним; тобто при збільшенні даних він схильний до так званої перепідгонці;

3) критерій AIC в загальному випадку є асимптотично ефективним.

Бассівський інформаційний критерій (Bayesian Information Criteria, BIC) [28]:

$$BIC = \log MSE + q \frac{\log N}{N}. \quad (32)$$

Особливостями даного критерію є наступне:

1) цей критерій вводить більш строгі штрафні характеристики, ніж AIC; тобто BIC «штрафує» моделі з великою кількістю параметрів більшою мірою, ніж це робить AIC;

2) критерій BIC є строго слухним критерієм: якщо дані в дійсності згенеровані моделлю, адекватною реальній, то BIC вибере справжню модель з одиначною ймовірністю з урахуванням того, що розмір вхідних даних буде прагнути до нескінченності;

3) BIC менш ефективний, ніж AIC.

Інформаційний критерій Кульбака (Kullback Information Criterion, KIC) [29]

$$KIC = N \ln \left[\frac{1}{N} \sum_{i=1}^N e_i^2 \right] + 3S \ln N. \quad (33)$$

Цей критерій є менш використовуваним за інші.

Інформаційний критерій Хеннана-Куїнна (HQ) [30]

$$HQ = N \ln \left[\frac{1}{N} \sum_{i=1}^N e_i^2 \right] + 2S \ln(\ln N). \quad (34)$$

Слід зазначити, що в окремому випадку нормального розподілу критерій AIC приймає вид

$$AIC(S) = RSS(S) + 2S, \quad (35)$$

званий критерієм Акаїке-Меллоуса.

Зважений інформаційний критерій (Weighted Information Criteria, WIC)

В роботі [31] було описано спроби поєднання статистичних та інформаційних критеріїв в одному зваженому критерії (Weighted Information Criterion, WIC)

$$WIC = 0,1(AIC + BIC) + 0,2(RMSE + MAPE) + 0,2((1 - DA) + MDA). \quad (36)$$

Слід зазначити, що коефіцієнти, які використовувалися в цьому критерії задавалися евристично не зрозуміло з яких міркувань.

Адаптивний зважений інформаційний критерій (adaptive weighted information criterion, AWIC).

В роботі [32] замість фіксованих використовувалися коефіцієнти, які обчислювалися шляхом вирішення деякої оптимізаційної задачі (при цьому критерій AIC та BIC бралися з вагою 0,1). Відповідний критерій було названо адаптивним (Adaptive WIC, AWIC)

$$AWIC = 0,1(AIC + BIC) + w_1 RMSE + w_2 MAPE + w_3(1 - DA) + w_4 MD. \quad (37)$$

Слід, однак, зазначити, що при цьому

1) не визначено критерій для оцінки вагових коефіцієнтів AIC и BIC;

2) немає жодних вказівок або рекомендацій щодо використання конкретних статистичних критеріїв, які б враховували поведінку часового ряду;

3) евристичні методи не були успішними в оцінці вагів для комбінованих методів;

4) цей критерій не враховує характеристик часових рядів.

9. U-статистика Тейла

Ця важлива міра розраховується за формулою [33]:

$$U = \frac{\sqrt{\sum_{i=1}^N (FPE_i - APE_i)^2}}{\sqrt{APE_i^2}}, \quad (38)$$

де $FPE_i = \frac{\hat{y}_i - y_i}{y_i}$ – відносна зміна прогнозних даних;

$APE_i = \frac{y_{i+1} - y_i}{y_i}$ – відносна зміна фактичних даних,

або

$$U = \frac{\sqrt{\frac{1}{N} \sum_{i=1}^N e_i^2}}{\sqrt{\frac{1}{N} \sum_{i=1}^N \hat{y}_i^2} \sqrt{\frac{1}{N} \sum_{i=1}^N y_i^2}}. \quad (39)$$

Цей показник дозволяє здійснювати порівняння формальних методів прогнозування з найпростішими («наївними») методами.

Як видно з виразів (38), (39) шляхом піднесення відносних похибок до квадрата ліквідується ефект взаємної компенсації помилок і забезпечується ефект, коли значні помилки мають більшу вагу, ніж дрібні.

Як базис для порівняння використовуються два прості методи:

а) метод «випадкового блукання» (random walk) – це математична модель процесу випадкових змін – кроків у дискретні моменти часу. При цьому передбачається, що зміна на кожному кроці не залежить від попередніх змін і від часу;

б) метод урахування сезонності – цей метод розглядає можливість наявності в часових рядах сезонності. Оскільки високий відсоток флуктуацій даних у межах часового ряду зумовлений саме сезонністю, цей метод може мати кращі результати, ніж перший, при цьому залишаючись достатньо простим.

U-статистика Тейла коливається в межах діапазону $[0; \infty]$ і є простою для інтерпретації:

– якщо $U = 1$, найпростіші базові методи є настільки точними, як і формальний метод прогнозування, що застосовується;

– якщо $U < 1$, формальний метод прогнозування, який застосовується, є кращим від базових методів. Наближення значення U-статистики до 0 свідчить про підвищення якості прогнозування порівняно з базовими методами;

– якщо $U > 1$, немає сенсу використовувати відповідний формальний метод прогнозування, оскільки базові методи дають кращі результати.

U-статистика Тейла може використовуватися при оцінці якості прогнозів, отриманих на основі різних методів та моделей.

є залежною від зміни масштабу та перетворення даних.

Може використовуватися при оцінці якості прогнозів, отриманих на основі різних методів та моделей.

Недоліки:

1) є залежною від зміни масштабу та перетворення даних;

2) її значення суттєво залежать від дисперсії ряду, що прогнозується (при прогнозуванні процесів з низькою дисперсією при явно невдалому

прогнозі U-статистика може мати мале значення, що обумовлюється низькою варіацією признаку).

Модифікована U-статистика Тейла [34]:

$$U = \frac{\sqrt{\frac{1}{N} \sum_{i=1}^N e_i^2}}{\sqrt{\frac{1}{N} \sum_{i=1}^N \hat{y}_i^2} \sqrt{\frac{1}{N} \sum_{i=1}^N y_i^2}} \frac{\sigma_y^2}{\sigma_{\hat{y}}^2}, \quad \text{якщо } \sigma_y^2 > \sigma_{\hat{y}}^2, \quad (40)$$

$$U = \frac{\sqrt{\frac{1}{N} \sum_{i=1}^N e_i^2}}{\sqrt{\frac{1}{N} \sum_{i=1}^N \hat{y}_i^2} \sqrt{\frac{1}{N} \sum_{i=1}^N y_i^2}} \frac{\sigma_{\hat{y}}^2}{\sigma_y^2}, \quad \text{якщо } \sigma_y^2 < \sigma_{\hat{y}}^2,$$

де $\sigma_y^2, \sigma_{\hat{y}}^2$ – дисперсії прогнозованого процесу та прогнозу відповідно, має за мету ліквідувати недолік традиційної статистики.

Ми розглянули деякі важливі міри (метрики) оцінки точності прогнозу. Кожна з цих мір має деякі унікальні властивості, відмінні від інших. У експериментах краще використовувати кілька критеріїв ефективності, що допоможе отримати знання про суму, величину та напрям загальної прогнозованої похибки. З цієї причини при прогнозуванні часових рядів аналітики зазвичай і використовують більше ніж одну міру для судження.

Крім того, існує ще достатньо велика кількість метрик, що представляють собою деякі модифікації розглянутих вище, наприклад, середня відносна абсолютна помилка (Mean Relative Absolute Error, MRAE), медіанна абсолютна процентна помилка (Median Absolute Percentage Error, MdAPE), медіанна відносна абсолютна помилка (Median Relative Absolute Error, MdRAE), процентна середньоквадратична помилка (Root Mean Square Percentage Error, RMSPE), відносна середня абсолютна помилка (Relative Mean Absolute Error, RelMAE), відносна середньоквадратична помилка (Relative Root Mean Square Error, RelRMSE), симетрична медіанна APE (Symmetric Median Mean Absolute Percentage Error, sMdAPE) та ін.]

10. Моделювання

Розглядалась задача прогнозування об'єму пасажиропотоку у аеропортах Німеччини. Для цього було використано данні з офіційного сайту Eurostat [35] – статистичної служба Європейського союзу, що займається збором статистичної інформації по країнам-членам ЄС і гармонізацією статистичних методів використовуваних даними країнами.

Слід зазначити, що часові ряди, які відображають пасажиропотік будь-якого транспорту є надзвичайно популярні при тестуванні різноманітних систем прогнозування, оскільки вони мають усі необхідні складові часового ряду – тренд, сезонність та циклічність. Отже, для прогнозування об'єму

пасажиропотоку у аеропортах Німеччини використовувався багатошаровий перцептрон (БП) з трьома входами та одним виходом (для прогнозування кількості пасажирів у наступному місяці використовувались данні за три минулих місяця). Для навчання нейронної мережі було сформовано навчаючу вибірку, у якій було відображено зміни пасажиропотоку Німеччини за останні 16 років. Результати моделювання з використанням критерію вибору структури MSE мережі наведено у рис. 2. В цьому випадку БП мав один прихований шар з 16 нейронами (архітектура 3-16-1).

В якості активаційної функції (АФ) нейронів вибиралася функція гіперболічного тангенсу

$$f_{th}(z) = \tanh(\alpha z) = \frac{e^{\alpha z} - e^{-\alpha z}}{e^{\alpha z} + e^{-\alpha z}}, \quad (41)$$

де α – деякий коефіцієнт, що впливає на вид функції активації.

Навчання (налаштування параметрів) перцептрону здійснювалося за допомогою алгоритму Левенберга-Марквардта.

На рисунку сполушною лінією показано реальний пасажиропотік, а пунктирною – спрогнозований за допомогою БП. Результати моделювання зі застосуванням усіх можливих помилок прогнозу зведено у табл.1.

Таблиця 1

Результати моделювання

Критерій	Архітектура мережі	Середня помилка прогнозу (у відсотках)
MAE	3-16-1	3,7794
MdAE	3-13-1	2,4964
MSE	3-16-1	1.3643
RMSE	3-14-1	3,3389
FRMQE	3-19-1	4,2612
MAPE	3-17-1	5,7794
MdAPE	3-18-1	2,8400
RMSPE	3-16-1	5,8643
RMdSPE	3-12-1	2,0651
SMAPE	3-15-1	4,5794
SMdAPE	3-12-1	4,9700
MRAE	3-18-1	2,1794
MdRAE	3-13-1	1,5964
GMRAE	3-15-1	1,5330
MASE	3-19-1	7,2145
RMSSE	3-20-1	6,6005
AIC	3-16-1	0,7693
BIC	3-16-1	0,7797
U-статистика Тейла	3-19-1	0,6572 (абсолютне значення)

11. Обговорення результатів дослідження та рекомендації щодо вибору критеріїв прогнозування

Як видно з результатів моделювання, використання різних критеріїв часто приводить до практично однакових результатів (однакової структури моделі). В даному випадку це можна пояснити зменшенням можливостей вибору структури моделей за рахунок використання в них однакового числа вхідних сигналів (триох).

Якщо ж розглядати задачу визначення оптимального числа вхідних сигналів, а потім вибору структури перцептронної моделі, то результати можуть бути зовсім іншими.

Слід зазначити, що на цей час існує досить багато досліджень, присвячених аналізу критеріїв якості прогнозування та розробці рекомендацій їх ефективного вибору. Розглянемо деякі з них.

Якщо оцінка прогнозу здійснюється для часових рядів з одним масштабом і були виконані

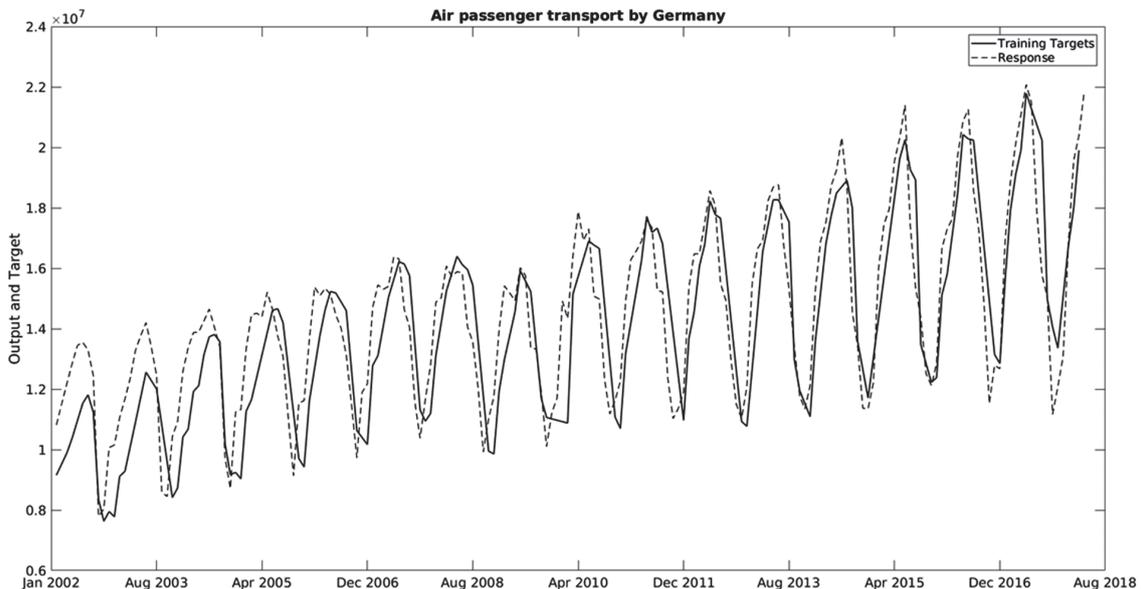


Рис. 2. Результати моделювання

процедури попередньої обробки даних (очищення даних, виявлення аномалій), розумно вибрати MAE, MdAE, RMSE. У разі використання різних масштабів ці міри критерії не застосовуються [23].

Щодо порівняння критеріїв MSE та MAE слід зауважити, що є тільки одна ситуація, в якій помилкам на основі MSE можна віддати перевагу щодо помилок на основі MAE – це прогнозування цілочисельного попиту. У цьому випадку число нулів у вибірці може бути настільки великим (може легко перевалити за 50 %), що всі помилки, засновані на модулях, будуть віддавати перевагу нульовому прогнозу. Це означає, наприклад, що при продажу нічого не продамо, тому нічого запасати – депресивний прогноз. Це пояснюється тим, що мінімум MAE відповідає медіанному значенню, в той час як мінімум MSE відповідає середньому.

Обидві ці оцінки MAE і MSE добре підходять, наприклад, для визначення точності прогнозу однієї і тієї ж послідовності при різних параметрах моделі або при використанні різних моделей, але ці оцінки виявляють помилок прогнозу менше, ніж при найвигіднішому методі. Тобто, точність результатів прогнозу перевершує точність найвигіднішого методу. І навпаки, якщо значення відносних оцінок більше одиниці, то точність результатів прогнозу в середньому гірше, ніж при найвигіднішому методі прогнозування.

Ці оцінки так само придатні для визначення точності прогнозу на два або більше кроків вперед. Для цього потрібно лише замість помилки прогнозу на один крок при їх розрахунку використовувати значення помилок прогнозу на відповідну кількість кроків вперед.

Якщо дані набору не містять викидів, і часовий проміжок, на якому проводилися розрахунки, не є настільки великим, щоб не мати близьких за значенням даних, доцільним є використання масштабованих оцінок.

Якщо дані мають сезонні або циклічні закономірності, рекомендується використовувати нормовані помилки, коли можуть бути розраховані коефіцієнти нормування в межах інтервалу, рівного циклу або сезону.

При оцінці точності декількох прогнозів потрібен спосіб агрегування. Якщо прийняти середнє арифметичне, тоді слід використовувати середню абсолютну процентну похибку (MAPE).

Для різномасштабних випадків можна надати наступні рекомендації.

Незважаючи на те, що процентні помилки зазвичай використовуються при прогнозуванні в реальних задачах, внаслідок несиметрії застосовувати їх не рекомендують. Якщо діапазон значень лежить в позитивній напівплощині, і в даних немає викидів, доцільно використовувати симетричні критерії.

Якщо дані «забруднені», тобто містять викиди, доцільно застосовувати масштабні міри, такі як MASE, inRSE. У цьому випадку необхідно [23], щоб

- 1) горизонт прогнозування був достатнім;
- 2) не було однакових значень прогнозуємого ряду;
- 3) значення нормалізованого коефіцієнта не повинно дорівнювати нулю.

Після розрахунку результати аналізуються щодо поділу на нульові помилки і наявності протиріч.

У зв'язку з тим, що на різних наборах даних виходять різні результати оцінок помилок прогнозування, при відсутності результатів попереднього аналізу і необхідності узагальнення методів замість однієї метрики прогнозування розумно використовувати певний набір метрик помилок.

Слід зазначити, що іноді в літературі зустрічається порівняння моделей на основі RMSE. Таке порівняння в загальному випадку некоректно [15] і зазвичай призводить до вибору неправильної, тобто не самої точної, моделі. Це обумовлено тим, що RMSE сильно схильна до впливу викидів (через квадрат у формулі). Тому в результаті використання цього критерію перевага віддається менш точним моделям, які посередньо прогнозують дані, але при цьому не допускають жодної великої помилки. Очевидно, що будь-які коефіцієнти, засновані на RMSE, також будуть приводити до спотворених результатів.

У зв'язку з тим, що на різних наборах даних виходять різні результати оцінок помилок прогнозування, а кожен критерій має недоліки, які можуть призводити до неточного визначення якості прогнозування, неможливо виділити один універсальний критерій. Тому розумним є використання певного набору метрик помилок.

Висновки

На цей час існує багато критеріїв оцінки ефективності вибору моделі для прогнозування часових рядів. Все ці критерії мають як переваги, так і недоліки, серед яких в першу чергу слід відзначити послідовність / непослідовність, чутливість до викидів, обчислювальний час тощо.

Аналіз традиційних критеріїв вибору моделі та помилок прогнозу показав, що вибір якісного критерію залежить від мети прогнозування та наявності інформації щодо статистичних властивостей даних та завад.

Якщо використання інформаційних критеріїв AIC, BIC є досить ефективним при визначенні структури нейромережевої моделі [36, 37], то запропоновані в останній час нові критерії оцінювання, які поєднують статистичні та інформаційні критерії в одному зваженому, мають ряд суттєвих

недоліків, зокрема не враховують характеристик часових рядів та містять суб'єктивний підхід до визначення вагових коефіцієнтів.

Розглянуті критерії вибору моделі аналізуються для прогнозування об'єму пасажиропотоку у аеропортах Німеччини. Отримані результати свідчать про те, що критерії вибору моделі на основі медіани прогнозів мають меншу мінливість, ніж інші, в той час як з критерії, що базуються на геометричному середньому, мають більшу мінливість.

При прогнозуванні економічних рядів з використання даних з високою волатильністю доцільним є застосування критеріїв, що базуються на медіані (MdAE, MdAPE, SMdAPE, MdRAE), і не використовувати інформаційні критерії, які вводять штрафи за складність моделі та неточність прогнозування (AIC, BIC).

Список літератури:

- [1] Bowerman B. Forecasting: methods and applications. 4th ed. / B. Bowerman, R. O'Connell, A. Koehler. – Belmont, CA.: Thomson Brooks, 2005. – 384 p.
- [2] Hyndman R.J. Forecasting: principles and practice. / R. J. Hyndman, G. Athanasopoulos. // 2nd ed, 2017. OTexts. <http://otexts.com/fpp>
- [3] Hanke J. E. Business forecasting (5th ed.) / J. E. Hanke, A.G. Reitsch. – NJ: Prentice-Hall, 1995. – 606 p.
- [4] Hyndman R. J. Another look at measures of forecast accuracy / R. J. Hyndman, A. B. Koehler. / Int. J. of Forecasting. – 2006. – 22 (4). – Pp. 679–688.
- [5] Gooijer J.G.D. 25 Years of Time Series Forecasting. / J.G.D. Gooijer, R.J.Hyndman / Int. Journal of Forecasting.- 2006.- 22(4).- Pp. 679-688.
- [6] Adhikari R. An Introductory Study on Time series Modeling and Forecasting // R. Adhikari, R.K. Agrawal. – LAP LAMBERT Academic Publishing, 2013 – 67 p.
- [7] Allende H. Artificial neural networks in time series forecasting: a comparative analysis. / H. Allende, C. Moraga. R. Salas // Kybernetika. – 2002. – 6. – Pp.685–707.
- [8] Cao L.J. Support Vector Machine with Adaptive Parameters in Financial Time Series Forecasting // L.J. Cao, F.E.H. Tay // IEEE Transaction on Neural Networks-2003.- Vol. 14.- No. 6.- P: 1506-1518.
- [9] Tay F.E.H. Application of support vector machines in financial time series forecasting / F.E.H.Tay, L. Cao // Omega. -2001.- 29.- P. 309–317.
- [10] Mahmoud E. Accuracy in forecasting: A respect to division by zero errors and contradiction survey. / E. Mahmoud // J. of Forecasting. – 1984. – 3(2). – Pp. 139–159.
- [11] Flores B.E. A pragmatic view of accuracy measurement in forecasting / B.E. Flores // Omega. –1986. –Vol. 14. – Issue 2. – Pp. 93–98.
- [12] Armstrong, J.S. Error measures for generalizing about forecasting methods: empirical comparisons. / J.S. Armstrong, F. Collopy. // Int. J. Forecasting. – 1992. –8.–Pp. 69–80.
- [13] Makridakis S. Accuracy measures: theoretical and practical concerns. / S. Makridakis // Int. J Forecasting.- 1993.- 9.-Pp. 527- 529.
- [14] Goodwin, P. On the asymmetry of the symmetric MAPE. / P. Goodwin, R. Lawton // Int. J. of Forecasting.- 1999.- 15.- Pp. 405-408.
- [15] Koehler A.B. The asymmetry of the sAPE measure and other comments on the M3-competition. // A. B. Koehler.// Int. J. of Forecasting.- 2001.- 17.- Pp. 570-574.
- [16] Tofallis C. A better measure of relative prediction accuracy for model selection and model estimation / C. Tofallis // J. of the Operational Research Society.-2015.- 66.- Pp. 1352 – 1362.
- [17] Хайкин С. Нейронные сети: полный курс. / С. Хайкин – М.: Изд. дом «Вильямс», 2006. – 1104 с.
- [18] Бодянский Е.В. Искусственные нейронные сети: архитектуры, обучение, применения. / Е.В. Бодянский, О.Г. Руденко. – Харків: ТЕЛЕТЕХ, 2004. – 369 с.
- [19] Myrtveit I. Validity and reliability of evaluation procedures in comparative studies of effort prediction models. // I. Myrtveit, E. Stensrud //Empirical Software Engineering.- 2012.- 17(1-2).- P.23-33.
- [20] Foss T. A simulation study of the model evaluation criterion MMRE. / T. Foss, E. Stensrud, B. Kitchenham, I. Myrtveit. // IEEE Transactions on Software Engineering,- 2003.-29.- Pp. 985 -995.
- [21] Green K. Percentage error: What denominator? / K. Green, L. Tashman// Int. Journal of Applied Forecasting.- 2009.-12.- Pp. 36-40.
- [22] Kolassa S. Percentage Errors Can Ruin Your Day (and Rolling the Dice Shows How) / S. Kolassa, R. Martin // Int. Journal of Applied Forecasting. - 2011.- issue 23.- Pp.21-27.
- [23] Shcherbakov M.V. A Survey of Forecast Error Measures / M.V. Shcherbakov, A.Brebels, N.L. Shcherbakova, A.P. Tyukov, T.A. Janovsky, V. Kamaev. // World Applied Sciences Journal 24 (Information Technologies in Modern Industry, Education & Society), –2013. – Pp.171–176.
- [24] Kim S. A new metric of absolute percentage error for intermittent demand forecasts / S. Kim, H. Kim // Int. J. of Forecasting. – 2016. –32.– Pp. 669–679.
- [25] Bal G. A comparison of different model selection criteria for forecasting Euro/USD exchange rates by feed forward neural network / C. Bal, S. Demir, C.H. Aladag // Int. J. of Computing, Communications & Instrumentation Engg. (IJCCIE).- 2016.- Vol. 3,- Issue 2.- P. 271-275]
- [26] Fildes R. The evaluation of extrapolative forecasting methods. / R. Fildes. // Int. J. of Forecasting. - 1992.- 8.- P.81–98.
- [27] Akaike H.A. A new look at statistical model identification / H.A. Akaike // IEEE Trans. on Automatic Control. - 1974. – 19. – P.716-723.
- [28] Schwarz G. Estimating the dimension of model / G. Schwarz // Ann. Statist. – 1978. – 6. – P.461-464.
- [29] Kullback S. Information theory and statistics. // S. Kullback /- New York, NY:Wiley, 1959.- 409 p.
- [30] Hannan E.J. The determination of the order of an autoregression / E.J. Hannan, B.G. Quinn / J. of the Royal Statistical Society. –1979. – 41. –P. 190–195.
- [31] Egrioglu E. A New Model Selection Strategy in Artificial Neural Network // E. Egrioglu, C.H. Aladag, S. Gunay / Applied Mathematics and Computation. –2008. – 195. –Pp. 591–597.
- [32] Aladag C.H. Improving weighted information criterion by using optimization / C.H. Aladag, E. Egrioglu, S. Gunay, M.A. Basaran / Journal of Computational and Applied Mathematics. –2010. – 233. –Pp. 2683–2687.
- [33] Тэйл Г. Экономическое прогнозирование и принятие решений. – М. Статистика, 1977.-282 с.
- [34] Пискунов Е.Ю. Модификация коэффициента Тэйла / Е.Ю. Пискунов // Математическое моделирование, системный анализ. –2012.- № 5. – С. 14-18. **35**.
- [35] http://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=avia_паос
- [36] Руденко О.Г. Нейросетевая аппроксимация нелинейных зашумленных функций на основе коэволюционного кооперативно- конкурентного подхода. / О.Г. Руденко, А.А. Бессонов / Проблемы управления и информатики –2018.- №3.- С. 5-14.
- [37] Руденко О.Г. Многокритериальная оптимизация эволюционирующих сетей прямого распространения / О.Г. Руденко, А.А. Бессонов / Проблемы управления и информатики. –2014.- № 6.- С. 29-41.

Поступила в редколлегию 14.11.2018

УДК 004.94: 519.688



М.С. Широкопетлева¹, О.А. Пономаренко², З.В. Дудар³

¹ХНУРЕ, м. Харків, Україна, marija.shirokopetleva@nure.ua;

²ХНУРЕ, м. Харків, Україна, oleksandr.ponomarenko1@nure.ua;

³ХНУРЕ, м. Харків, Україна, zoya.dudar@nure.ua

ПОРІВНЯННЯ МЕТОДІВ ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ

Стаття присвячена опису і порівнянню моделей прогнозування часових рядів і виявленню можливостей застосування різних моделей для вирішення задач прогнозування з різними вихідними даними: часові інтервали, наявність сезонності та/або трендів. Крім цього, серед двох популярних методів прогнозування ARIMA і ANN проведені більш детальний огляд і практичне порівняння на прикладі реальних часових рядів вартості житнього хліба в Україні, визначені похибки прогнозування на короткостроковий і довгостроковий періоди. Оцінка похибки проводилася з використанням програмної системи, представленої в даній статті, шляхом порівняння відхилення результатів прогнозування від реальних даних за останній період, що становить 1/4 вихідних даних. Згідно з отриманими результатами зроблено висновки щодо ефективності обох методів і потенційних можливостей їх майбутнього використання. Також пропонується використовувати представлену програмну систему для екстраполяції рядів в різних областях, таких як економіка, технічні системи, освіту, природні та соціальні системи.

ЧАСОВІ РЯДИ, ПРОГНОЗУВАННЯ, МОДЕЛЬ ПРОГНОЗУВАННЯ, ІНТЕГРОВАНА МОДЕЛЬ АВТОРЕГРЕСІЇ – КОВЗНОГО СЕРЕДНЬОГО, ШТУЧНІ НЕЙРОННІ МЕРЕЖІ, ПОРІВНЯННЯ

М.С. Широкопетлева, А.А. Пономаренко, З.В. Дудар. Сравнение методов прогнозирования временных рядов. Статья посвящена описанию и сравнению моделей прогнозирования временных рядов и выявлению возможностей применения различных моделей для решения задач прогнозирования с различными исходными данными: временные интервалы, наличие сезонности и/или трендов. Кроме этого, среди двух популярных методов прогнозирования ARIMA и ANN проведены более подробный обзор и практическое сравнение на примере реальных временных рядов стоимости ржаного хлеба в Украине, определены погрешности прогнозирования на краткосрочный и долгосрочный периоды. Оценка погрешности проводилась с использованием программной системы, представленной в данной статье, путем сравнения отклонения результатов прогнозирования от реальных данных за последний период, составляющий 1/4 исходных данных. Согласно полученным результатам сделаны выводы относительно эффективности обоих методов и потенциальных возможностей их будущего использования. Также предлагается использовать представленную программную систему для экстраполяции рядов в различных областях, таких как экономика, технические системы, образование, природные и социальные системы.

ВРЕМЕННЫЕ РЯДЫ, ПРОГНОЗИРОВАНИЕ, МОДЕЛЬ ПРОГНОЗИРОВАНИЯ, ИНТЕГРИРОВАННАЯ МОДЕЛЬ АВТОРЕГРЕССИИ – СКОЛЬЗЯЩЕГО СРЕДНЕГО, ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ, СРАВНЕНИЕ

Mariya Shirokopetleva, Oleksandr Ponomarenko, Zoya Dudar. Comparison of time series forecasting methods. The article is devoted to the description and comparison of time series forecasting models and identifying the possibilities of using various models for solving forecasting problems with different initial data: time intervals, presence of seasonality and / or trends. In addition, among the two popular ARIMA and ANN forecasting methods, a more detailed review and practical comparison was made using the example of real time series of the cost of rye bread in Ukraine, forecast errors for the short and long term were identified. The estimation of the error was carried out using the software system presented in this article by comparing the deviation of the prediction results from the real data for the last period, which is 1/4 of the original data. According to the results obtained, conclusions were drawn regarding the effectiveness of both methods and the potential possibilities of their future use. It is also proposed to use the presented software system for extrapolating series in various fields, such as economics, technical systems, education, natural and social systems.

TIME RANGES, FORECASTING, MODEL OF FORECASTING, AUTOREGRESSIVE INTEGRATED MOVING AVERAGE, ARTIFICIAL NEURAL NETWORKS, COMPARISON

Вступ

Прогнозування передбачає майбутні події на основі даних минулого чи теперішнього часу. Цей процес описує наукове дослідження перспектив розвитку для будь-якого процесу. І найчастіший спосіб представляти ці процеси — використання часових рядів.

Часові ряди — це послідовність деяких значень вимірювань, упорядкованих за їхнім виглядом в інтервалі часу. У часовому ряді є два обов'язкові

елементи: відмітка часу і значення, пов'язане з ним.

1. Класифікація часових рядів

Існує багато типів класифікації часових рядів. Вони можуть бути розділені за часом, за частотою отримання значень часових рядів, за балансом між стабільним середнім рівнем тощо.

Також часові ряди можна розділити за часовим горизонтом прогнозування. Прогнози можуть бути

короткостроковими, середньостроковими або довгостроковими. Однак у цьому питанні є важливе уточнення. Діапазони тривалості прогнозів і навіть їх кількість відрізнятимуться від завдання до завдання. Це означає, що різні часові ряди матимуть власний, відмінний від інших, класифікацію тривалості дії.

Говорячи про часові ряди, слід розрізнити концепцію методу прогнозування та модель прогнозування, які є подібними через їх співвідношення.

Метод прогнозування описує порядок дій, що необхідно зробити, щоб отримати модель прогнозування часових рядів. Також метод має виконувати операції, щоб оцінити якість прогнозних значень.

У той же час модель прогнозування є функціональним представленням часових рядів. Це наша основа для прогнозування значень майбутнього процесу. Загальний ітеративний спосіб побудови моделі прогнозування складається з наступних кроків:

1) Вибір класу загальних моделей для прогнозування часових рядів для вибраного часового горизонту. Зазвичай цей клас вибирається на підставі минулого досвіду.

2) Вибір моделі підкласу. Для більшості моделей прогнозування існує безліч варіантів підмоделей. Цей крок вимагає більш глибокого аналізу часових рядів та вимог прогнозування.

3) Аналіз параметрів моделі. Для цього кроку потрібні дані, які дозволяють вказати атрибути моделі прогнозування. Зазвичай це найбільш трудомістка частина роботи, тому що цей крок враховує багато історичних параметрів часових рядів.

4) Верифікація отриманої моделі. Часто, потрібно взяти одну або декілька областей, які будуть використовуватися для тестування прогнозу та оцінки точності. Елементи, які використовуються для перевірки, називаються контрольно-пропускними пунктами.

5) Аналіз результатів. Якщо точність отриманих результатів на попередньому кроці є достатньою, то вважається, що модель готова до використання. В іншому випадку, можна послідовно повторити попередні кроки, починаючи з першого.

Існує два варіанти постановки проблеми прогнозування часових рядів. Перший варіант враховує дані, доступні тільки для цього часового ряду. Другий варіант дозволяє використовувати набори даних, які співвідносяться з існуючими часовими рядами, але взяті ззовні розглянутих рядів. За звичайним підходом, ці додаткові дані також повинні бути представлені як часові ряди. У даній роботі увага буде зосереджена насамперед на часових рядах без впливу зовнішніх чинників.

Як згадувалося вище, часові ряди прогнозування вимагають визначення моделі прогнозування. Метою створення моделі прогнозування є створення моделі з мінімальним середнім абсолютним відхиленням реальної вартості від прогнозу. Наші розрахунки обмежені горизонтом, який також можна назвати часом виконання.

2. Порівняння моделей прогнозування

На основі роботи [1] можна побудувати приbliżну класифікацію методів і моделей прогнозування.

По-перше, ми розподіляємо методи прогнозування на дві групи: інтуїтивні та формалізовані методи.

Інтуїтивні методи засновані на експертних оцінках. Ці методи застосовуються до систем, опис яких погано оформлений або не може бути формалізований.

Формалізовані методи визначають математичну залежність, яка дозволяє прогнозувати майбутнє значення процесу, тобто побудувати модель прогнозування.

Оскільки інтуїтивні методи не можуть бути описані функціонально, будуть розглянуті лише формалізовані. Тепер ми можемо описати моделі прогнозування. Вони також умовно можна розділити на дві категорії: моделі доменів та моделі часових рядів.

Перша група описує моделі, прив'язані до певної теми, і які не можуть бути повністю переведені в інші предметні області.

У той же час друга група описує моделі, які шукають залежності в рамках самого процесу. Ці моделі є універсальними для різних предметних областей, тобто їх загальний вигляд не змінюється в залежності від характеру часових рядів.

Тепер ми можемо вилучити з розгляду ще одну категорію і не враховувати модель предметних областей. Це безглуздо, оскільки їх кількість еквівалентна кількості предметних областей. Тим не менше, ми можемо класифікувати моделі часових рядів. Вони знову будуть розділені на дві групи: статистичні моделі та структурні моделі.

Статистичні моделі описують взаємозв'язок між минулим та майбутнім значенням за допомогою рівнянь. До них відносяться: моделі регресії (лінійна та нелінійна регресія), авторегресійні моделі (ARIMAX, GARCH та ін.), Модель експоненціального згладжування тощо.

Структурні моделі визначають однакові залежності у вигляді певної структури та правил переходу на неї. До них відносяться: моделі штучних нейронних мереж, моделі, засновані на ланцюгах Маркова, моделі моделей класифікації та регресії

тощо. У ряді робіт [2], [3], [4] зазначено, що на сьогоднішній день найбільш поширеними моделями прогнозування є авторегресійні моделі (ARIMAX), а також нейромережеві моделі (ANN). Крім цього, в дослідженнях [5], [6] наведені ідеї, що дозволяють використовувати обидва підходи одночасно, наприклад, використовуючи нейронні мережі для розбиття даних на кластери, а ARIMA — для побудови безпосередньо прогнозу часового ряду.

Слід також зазначити, що якщо модель ARIMA (X) давно і широко відома, нейронні мережі досі має безліч нерозглянутих частин, оскільки інтерес до них з'являється з певною періодичністю.

Однак, незважаючи на сказане вище, вже зараз обидві моделі показали себе, як ефективні, тому варто їх порівняти, для більш повного і всебічного розуміння предметної області.

3. Математична модель

Штучна нейронна мережа — це математична модель, побудована на принципі організації та функціонування біологічних нейронних мереж — нервових клітинних мереж живого організму. З погляду математики це система пов'язаних і взаємодіючих обчислювальних одиниць, званих нейронами. Простий приклад нейронної мережі показаний на рисунку 1.

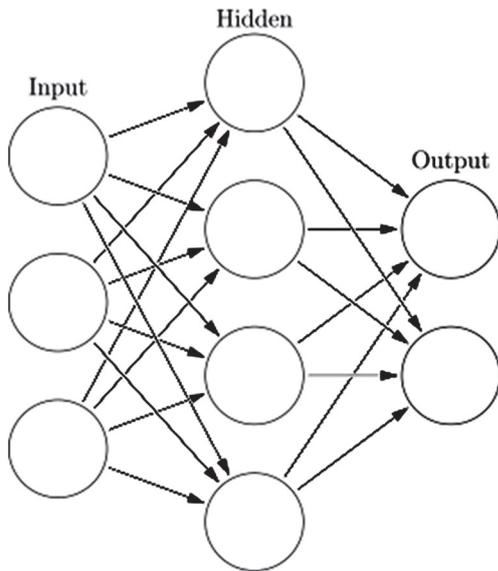


Рис. 1. Приклад штучної нейронної мережі (ANN)

Цей приклад ANN складається з трьох шарів: вхід, прихований і вихідний. Ці шари є основними етапами процесу роботи ІНН. Вхідний шар отримує інформацію, прихований шар (або шари) обробляє його, а вихідний шар повертає результати.

У свою чергу, нейрон — це обчислювальна одиниця, яка отримує інформацію, обчислює її та передає її далі. Вони поділяються на три основних типи, як і відповідні шари. Приклад нейрона показаний на рисунку 2.

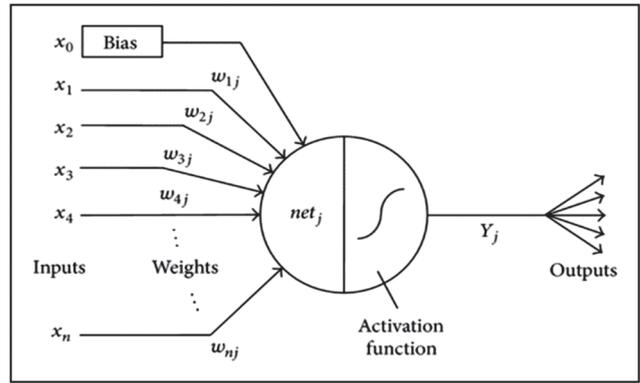


Рис. 2. Приклад нейрона

Як викладено вище, нейрон отримує набір сигналів. Вони можуть бути або взяті ззовні, або передані іншими нейронами. Отримані значення помножуються на ваговий коефіцієнт. Таким чином, нейрон отримує всі дані в функції-суматорі, представленій нижче:

$$u = \sum_{i=1}^n \omega_i x_i, \quad (1)$$

де ω — вектор синаптичних ваг і x — вектор вхідних значень.

Тепер нам потрібно передати отримане значення в функцію активації. Зазвичай логістична функція (сигмоїд) та її похідні приймаються як активатор. Формула (2) показує приклад такої функції:

$$S(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{e^x + 1}, \quad (2)$$

де x — результат функції-суматора. Значення цієї функції знаходиться в діапазоні від 0 до 1.

Тоді результат роботи переноситься на наступні пов'язані нейрони або на вихід. Подальші дії в мережі можуть відрізнитися в залежності від її типу.

Приблизно нейронна мережа навчається за наявними даними. Проте, якщо нейронна мережа передає дані лише з одного зразка, вона може налаштувати результати спеціально для нього. Щоб уникнути цієї ситуації, дані для тренувань поділяються на три категорії: навчання, тестування та перевірки.

Навчальна група даних впливає на навчальний процес і служить джерелом даних для нейронної мережі. Група тестування необхідна для перевірки результатів навчання за подібними даними. У той же час він забезпечує належну підготовку нейронної мережі. Група підтвердження/перевірки потрібна як остаточний тест мережі. Вона вступає у гру, коли розробник неодноразово задоволений результатами проходження навчань та тестування зразків.

Для створення зразків вихідні дані можна розділити на декілька частин у різних пропорціях. Це дає змогу мати дані одного типу, які не спричинять впливу на мережу через часте виникнення в процесі навчання.

Звичайно, кожне завдання вимагає свого підходу та точності, однак, правило ділення «70/20/10» часто використовується на практиці. Сімдесят відсотків даних потрапляє на навчання, двадцять на тест, а десять відсотків залишаються на останньому контрольній точці — перевірка.

Тепер, зрозумівши основи побудови та навчання ANN, ми проаналізуємо метод ARIMA.

Модель ARIMA, вона ж інтегрована модель авторегресії — ковзносереднього або модель Бокса-Дженкінса, є узагальненням ARMA-моделі і додає до даної моделі складову інтеграції. Найбільш простим способом для розбору даної моделі буде розбиття її за складовими акроніма.

AR — авторегресія. Дана складова являє собою модель часових рядів, в якій поточне значення часового ряду лінійно залежить від попередніх значень цього ж ряду. Формула авторегресії порядку p виглядає, як:

$$X_t = c + \sum_{i=1}^p a_i X_{t-i} + \epsilon_t, \quad (3)$$

де a_i — параметр моделі, він же, коефіцієнт авторегресії (визначається через систему рівнянь), ϵ_t — білий шум, c — постійна.

При використанні моделі авторегресії необхідно визначити порядок p і кількість рівнянь, необхідне для максимально точного прогнозу за допомогою коефіцієнтів авторегресії.

I — інтеграція. При прогнозуванні часового ряду $Z(t)$ може виникнути ситуація, що прогноз буде точнішим і простіше при роботі не з самим процесом $Z(t)$, а з його зміною, тобто, в такому випадку, ми отримуємо часовий ряд

$$\bar{Z}(t) = Z(t) - Z(t-1), \quad (4)$$

Після цього, ми так само використовуємо модель авторегресії.

Також варто відзначити дві деталі. Застосовувати прийом з інтеграцією можна двічі, отримуючи таким чином інтеграцію другого порядку. Однак, в той же час, варто відзначити, що брати порядок більше другого не має сенсу.

Крім цього, в разі наявності у ряду періодичності, наприклад, добової або годинної, то різниця рядів можна отримувати з урахуванням періоду. В такому випадку, для прикладу тимчасового ряду з похвилинними вимірами, формула може виглядати як:

$$\bar{Z}(t) = Z(t) - Z(t-60), \quad (5)$$

MA — ковзне середнє. Дана модель використовує залежність між наглядом і залишковою помилкою від моделі ковзносереднього, застосованої до спостережень, що містить викиди. Під викидами в даному випадку маються на увазі «піки» і «западини», що вибиваються із загального фону тимчасового ряду.

Використання моделі ковзносереднього обумовлено як раз згладжуванням подібних подібних ділянок. Формула простого ковзного середнього порядку n має вигляд:

$$SMA_t = \frac{1}{n} \sum_{i=0}^{n-1} p_{t-i}, \quad (6)$$

де p_{t-i} — значення вихідної функції в точці $t-i$, n — кількість значень вихідної функції для розрахунку ковзносереднього, її також називають згладжуючим інтервалом.

Таким чином, дізнавшись структуру і основні закономірності, що лежать в основі методів ARIMA і ANN, ми можемо провести їх практичне порівняння на тимчасових рядах.

4. Експериментальне порівняння результатів прогнозування з використанням методів ARIMA та ANN

Як приклад візьмемо щомісячний тимчасовий ряд вартості тонни житнього хліба з України в проміжку від січня 2012 року до жовтня 2018. Його зображено на рисунку 3.

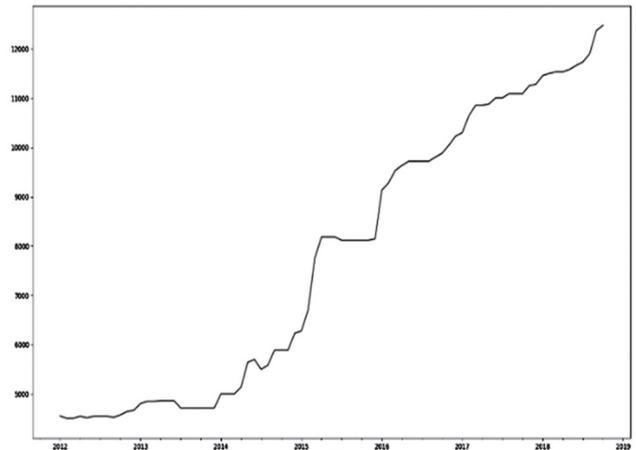


Рис. 3. Часовий ряд вартості хліба за тону

Для побудови прогнозу часового ряду воспользуємося моделлю ARIMA з наступними параметрами: ARIMA (p, d, q) = ARIMA (5, 1, 0), де p — кількість спостережень за затримкою, включених в модель, також називається порядком запізнювання, d — кількість раз, коли необроблені спостереження будуть диференційовані, також називається ступенем диференціації, q — розмір вікна ковзної середньої, також званий порядком ковзної середньої. На малюнку 4 представлений прогноз останніх 12 місяців часового ряду.

Таблиця 1 наводить порівняння очікуваних і реальних значень часового ряду. Помилка прогнозування обчислюється як:

$$E = \frac{(v_{\text{реал}} - v_{\text{ожд}})}{v_{\text{реал}}}, \quad (7)$$

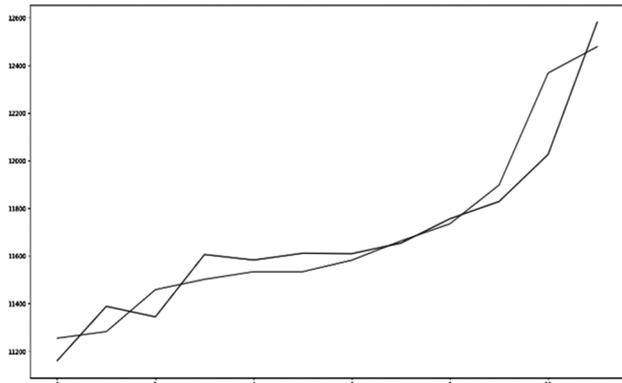


Рис. 4. ARIMA-прогноз останніх 12 місяців часового ряду

Таблиця 1

Порівняння та похибка ARIMA-методу

Значення місяця	Реальне значення ($v_{\text{реал}}$)	Прогнозоване значення ($v_{\text{ожд}}$)	Похибка прогнозу (E)
11/2017	11255,30	11160,634571	0,008411
12/2017	11282,61	11388,754791	-0,009408
01/2018	11458,82	11344,667999	0,009962
02/2018	11502,31	11606,377510	-0,009048
03/2018	11533,98	11583,260397	-0,004273
04/2018	11533,98	11611,704373	-0,006739
05/2018	11582,67	11609,905483	-0,002351
06/2018	11663,32	11654,096395	0,000791
07/2018	11735,57	11756,559486	-0,001789
08/2018	11897,85	11828,813015	0,005802
09/2018	12368,51	12026,622196	0,027642
10/2018	12478,97	12582,273452	-0,008278

Тепер проведемо аналогічні обчислення для часового ряду за допомогою штучних нейронних мереж. Для аналізу даних за допомогою нейронної мережі скористаємося типовою тришаровою мережею з одним прихованим шаром. Результати аналізу можна побачити на рисунку 5.

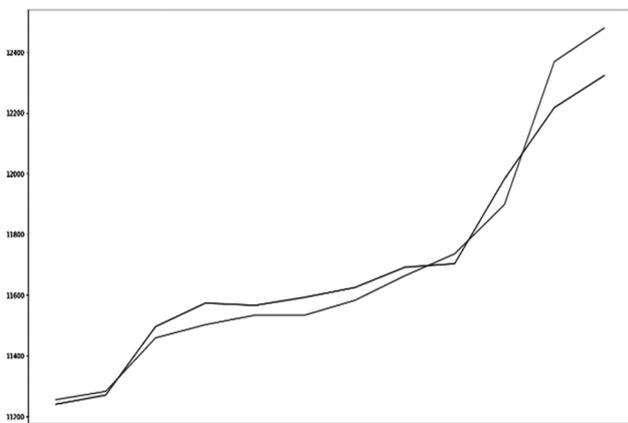


Рис. 5 ANN-прогнозування останніх 12 місяців часового ряду

Числові результати роботи, реальні значення та похибку штучної нейронної мережі сформульовано у таблиці 2.

Таблиця 2

Порівняння та похибка ANN-методу.

Значення місяця	Реальне значення ($v_{\text{реал}}$)	Прогнозоване значення ($v_{\text{ожд}}$)	Похибка прогнозу (E)
11/2017	11255,30	11240.233434	0,001339
12/2017	11282,61	11270.412341	0,001081
01/2018	11458,82	11495.942533	-0,003239
02/2018	11502,31	11573.437618	-0,006184
03/2018	11533,98	11566.127683	-0,002787
04/2018	11533,98	11592.643498	-0,005086
05/2018	11582,67	11624.876249	-0,003644
06/2018	11663,32	11691.359675	-0,002404
07/2018	11735,57	11703.135786	0,002763
08/2018	11897,85	11982.138124	-0,007084
09/2018	12368,51	12217.165807	0,012236
10/2018	12478,97	12322.415602	0,012545

Бачимо, що абсолютна похибка прогнозу для ARIMA-прогнозу дорівнює 0,007875, а для ANN-прогнозування — 0,005033, що є припустимим відхиленням при прогнозуванні.

5. Опис прототипу веб-системи прогнозування

Для розробки веб-системи використано кілька технологій, частина з яких взяла на себе відображення контенту, частина займається зв'язком з іншими складовими програмної системи. Для розробки серверної частини використано Django Framework. Взаємодія між клієнтом та сервером відбувається через HTTPS-з'єднання. Для опису моделей прогнозування використовуються бібліотеки:

- numpy — розширення мови Python, що додає підтримку великих багатовимірних масивів і матриць, разом з великою бібліотекою високорівневих математичних функцій для операцій з цими масивами;
- scipy — відкрита бібліотека високоякісних наукових інструментів для мови програмування Python;
- sklearn — безкоштовна бібліотека машинного навчання для Python,
- pandas — популярна бібліотека для обробки і аналізу даних;
- matplotlib — бібліотека на мові програмування Python для візуалізації даних двовимірної (2D) графікою (3D графіка також підтримується);
- pillow — бібліотека, призначена для роботи з растровою графікою.

Основним компонентом даної системи є серверна частина. В якості СУБД було обрано PostgreSQL. Дана система є популярним рішенням, з великою підтримкою та документацією. Модульний рівень містить основні класи, що описують бізнес-модель.

Для виклику методів використовується REST API, створене за допомогою django-rest-framework,

яке допомагає обробляти потік даних по протоколу HTTP.

Сервер розподілений на кілька рівнів, що пов'язані між собою. До рівнів додатку входять: власне рівень API, бізнес-рівень, рівень доступу до бази даних.

Для розробки сайту було використано стандартний набір технологій: HTML, CSS, JS, JQuery, AJAX.

На рис.6 наведено сторінку з критичними відхиленнями прогнозу, його середньоквадратична похибка та додаткові дані, залежно від методу (значення коефіцієнту згладжування, тощо).

На рисунку 7 продемонстровано відносні похибки кожного з методів на кожному кроці створення прогнозу, зведені у єдину таблицю. Під даною таблицею приведено значення середньо-

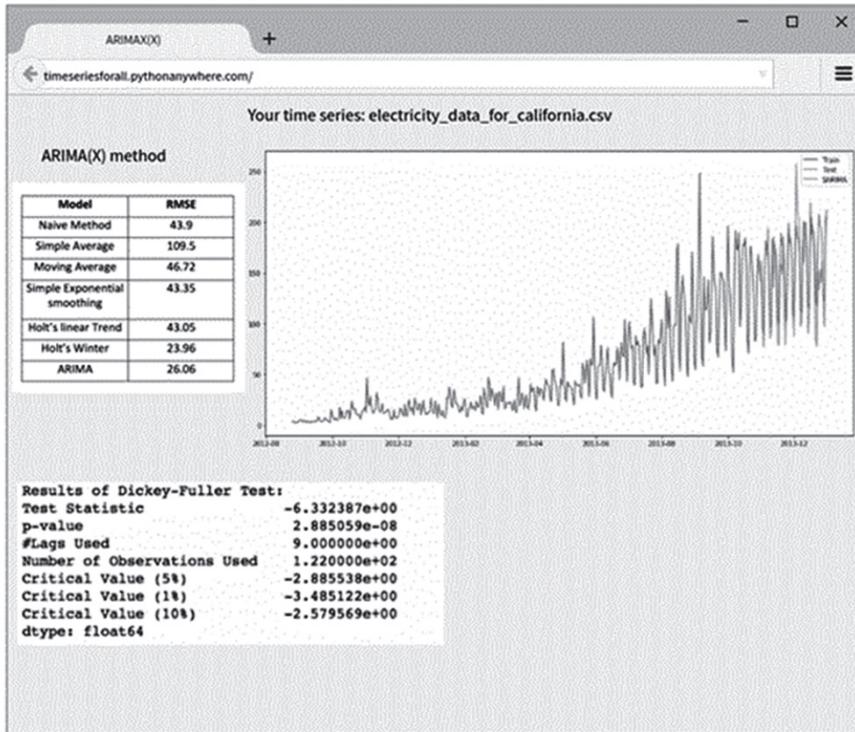


Рис. 6. Приклад вікна детальних результатів прогнозу завантаженого ряд

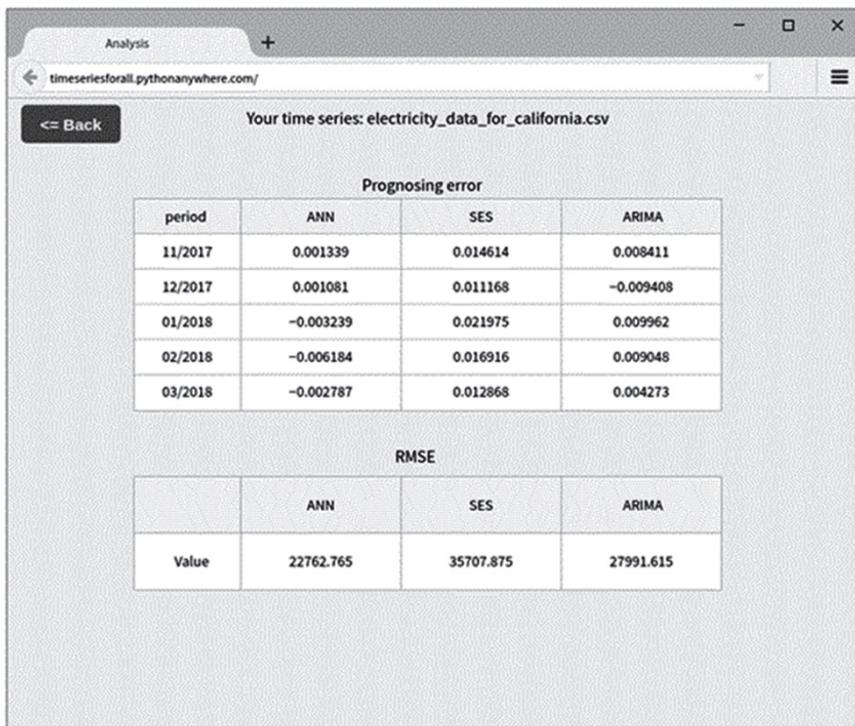


Рис. 7. Приклад вікна похибок прогнозів

квадратичного відхилення (RMSE) для кожного з методів.

Дана версія проекту надає змогу завантажити числовий ряд у форматі .csv, на основі яких, після аналізу будуть створені прогнози за допомогою трьох методів. Окрім цього, отримані графіки можна завантажити для подальшого аналізу.

Висновки

В межах даної роботи були розглянуті та проаналізовані такі методи прогнозування часових рядів, як ARIMA і ANN. Продуктивність обох методів була перевірена на однакових часових рядах. Результати показали, що і модель ARIMA, і модель ANN можуть досягти гарного прогнозу при застосуванні до реальних проблем і, таким чином, можуть ефективно використовуватися для задач прогнозування. Також, варто відзначити, що нейронмерева модель має більш високу точність прогнозу на проміжках часу, найбільш близьких до вихідних даних, в той же час, модель ARIMA і, відповідно, споріднені з нею, краще визначають напрямки і значення прогнозу на «далеких ділянках» часових рядів, і таким чином, добре себе показують для довгострокового прогнозування.

Отримані результати дозволяють висунути ідеї про використання обох моделей одночасно, що може компенсувати недоліки кожної з них. В першу чергу, моделі можуть бути використані окремо, для більш точного прогнозування значень на необхідний часовий проміжок. Також, через різницю у результатах можна висунути ідеї про поєднання вищезазначених методів. В першу чергу, можна казати про використання обох методів з врахуванням розгляду лише певних періодів в кожному з них. Однак, існує більш цікава альтернатива. Один з методів (зазвичай, штучні нейронні мережі) можна

застосувати для вирішення задачі кластеризації, що дозволить краще виокремити діапазони для прогнозування. А після цього, до отриманих наборів даних використати ARIMA-прогнозування.

Список літератури:

- [1] Чучуева И.А. Модель прогнозирования временных рядов по выборке максимального подобия, диссертация ... канд. тех. наук / Московский державный технический университет им. Н.Е. Баумана. Москва, 2012.
- [2] Егошин А.В. Анализ и прогнозування складних стохастичних сигналів на основі методів ведення кордонів реалізацій динамічних систем: Автореферат дисертації ... канд. техн. наук. Санкт-Петербург, 2009. 19 с.
- [3] Gheyas I.A., Smith L.S. A Neural Network Approach to Time Series Forecasting // Proceedings of the World Congress on Engineering, London, 2009 Vol 2 [електронний ресурс]. P. 1292 — 1296. URL: www.iaeng.org/publication/WCE2009/WCE2009_pp1292-1296.pdf (дата звернення 28.10.2018).
- [4] Alfares H.K., Nazeeruddin M. Electric load forecasting: literature survey and classification of methods // International Journal of Systems Science. 2002 Vol 33. P. 23 — 34.
- [5] Hinman J., Hickey E. Modeling and forecasting short term electricity load using regression analysis // Journal of Institute for Regulatory Policy Studies [електронний ресурс]. 2009. 51 p. URL: <http://www.irps.ilstu.edu/research/documents/LoadForecastingHinman-HickeyFall2009.pdf> (дата звернення 28.10.2018).
- [6] A. Jain, A. M. Kumar, "Hybrid neural network models for hydrologic time series forecasting," Applied Soft Computing Journal, vol. 7, no. 2, pp. 585-592, 2007.
- [7] J. Sterba and K. Hilovska, "The implementation of hybrid ARIMA neural network prediction model for aggregate water consumption prediction," Aplimat-Journal of Applied Mathematics, vol. 3, no. 3, pp. 123-131, 2010.

Поступила до редколегії 21.11.2018

УДК 681.3.07

Євгеній Путятін¹, Павло Оробінський², Вячеслав Ляшенко³¹Харківський національний університет радіоелектроніки,
Харків, Україна, yevhenii.putiatin@nure.ua²Харківський національний університет радіоелектроніки,
Харків, Україна, orbpster@gmail.com³Харківський національний університет радіоелектроніки,
Харків, Україна, lyashenko.vyacheslav@gmail.com

ВИЯВЛЕННЯ ВУЗЛИКІВ ЛЕГКОГО НА ЦИФРОВИХ МЕДИЧНИХ ЗОБРАЖЕННЯХ

Дана робота представляє дослідження, в якому розглядаються питання цифрової обробки та аналізу медичних зображень. В якості медичних зображень розглянуті зображення легкого людини, які отримані за допомогою комп'ютерної томографії. Запропоновано процедуру виявлення вузликів легкого. Це допомагає проведенню діагностики захворювання раку легенів. Показана працездатність й ефективність запропонованої процедури.

ВУЗЛИКИ ЛЕГКОГО, ЦИФРОВЕ ЗОБРАЖЕННЯ, ЕНЕРГЕТИЧНИЙ РІВЕНЬ, НЕЙРОННА МЕРЕЖА, КОМП'ЮТЕРНА ТОМОГРАФІЯ, ОЧАГ УРАЖЕННЯ.

Putyatin Ye., Orobinskiy P., Lyashenko V. Detection of lung nodules on digital medical images. This paper presents a study that addresses the issues of digital processing and analysis of medical images. As the medical images considered human lung images that are obtained by computer tomography. Procedure for detecting lung nodules considered. This helps to conduct the diagnosis of lung cancer disease. The efficiency and effectiveness of the proposed procedure is shown.

NODULES OF THE LUNG, DIGITAL IMAGE, ENERGY LEVEL, NEURAL NETWORK, COMPUTED TOMOGRAPHY, LESION FOCUS.

Путятин Е., Оробинский П., Ляшенко В. Обнаружение узелков легкого на цифровых медицинских изображениях. Данная работа представляет исследование, в котором рассматриваются вопросы цифровой обработки и анализа медицинских изображений. В качестве медицинских изображений рассмотрены изображения легкого человека, которые получены при помощи компьютерной томографии. Предложена процедура обнаружения узелков легкого. Это помогает проведению диагностики заболевания рака легких. Показана работоспособность и эффективность предложенной процедуры.

УЗЕЛКИ ЛЕГКОГО, ЦИФРОВЕ ИЗОБРАЖЕНИЕ, ЭНЕРГЕТИЧЕСКИЙ УРОВЕНЬ, НЕЙРОННАЯ СЕТЬ, КОМПЬЮТЕРНАЯ ТОМОГРАФИЯ, ОЧАГ ПОРАЖЕНИЯ.

Вступ

Обробка зображень є одним з поширених напрямів пізнання оточуючого нас середовища. Цей напрям пізнання застосовується у різноманітних інформаційних системах не тільки для аналізу наявної інформації, а й для отримання додаткових даних [1, 2]. Такі додаткові дані здатні розширити розуміння досліджуваного, вірно оцінити наявну інформацію. Отже, обробка зображень є потужним інструментом з аналізу даних. При цьому такий інструмент сприяє дослідженню тих процесів та явищ, які досить складно виявити за допомогою звичайного людського зору.

Основу застосування різноманітних методів та підходів щодо обробки зображень визначає, насамперед, фіксація досліджуваного (процесу, явища, об'єкту) за допомогою різноманітних систем та перетворення такої інформації у цифровий образ, який й визначає певне цифрове зображення. Серед окремих систем з отримання цифрових зображень досліджуваного слід вказати [3, 4]:

- фотографування або відео фіксацію;
- застосування даних ультразвукового аналізу;

– впровадження комп'ютерної томографії або позитронно-емісійної томографії;

– світової або електронної мікроскопії тощо.

У той же час серед окремих напрямів пізнання оточуючого нас середовища, на підставі обробки та аналізу цифрових зображень, варто виділити цифрові медичні зображення. Тобто, це цифрові зображення що пов'язані з відображенням та дослідженням різноманітних медичних процесів, які спрямовані на вивчення можливих хвороб людини або їх попередження. Цифрові медичні зображення дозволяють проводити поглиблене дослідження організму людини, вивчати динаміку його функціонування та проводити діагностику можливих захворювань на ранніх стадіях їх розвитку. Таким чином, цифрові медичні зображення – це спеціальні зображення, які вимагають застосування й окремих методів щодо їх обробки відповідно до предметної області, яка не лише визначає джерело формування відповідного цифрового медичного зображення, але й обумовлює присутність специфічних ознак розуміння відповідної інформації.

Останнім часом серед цифрових медичних зображень особливої уваги заслуговують зображення

легень людини. Це пов'язано із необхідністю попередження такої хвороби як рак, яка набуває значного поширення.

1. Виявлення вузликів легень як об'єкт дослідження

Цифрове зображення легень людини, зазвичай, отримується за допомогою комп'ютерної томографії (КТ). Потім до таких цифрових зображень застосовуються різноманітні методи з метою їх обробки та виявлення необхідної для дослідження інформації. У нашому випадку об'єктом такого дослідження є злоякісні вузлики що пов'язані з первісними пухлинами або метастазами. Тож важливість діагностики ранніх первинних пухлин легень, виявлення легневих вузликів має вирішальне значення.

У працях [5-7] надається докладний аналіз методів, які застосовуються для виявлення вузликів легень. Зокрема в дослідженні [5] робиться наголос на порівняльній оцінці щодо упровадження різноманітних методів виявлення вузликів легень. Так для цього можуть бути застосовані такі підходи та методи, що засновані на:

- методології опорних векторів;
- застосуванні спеціалізованих функцій для виявлення вузликів;
- ідеології багатошарового перцептронну.

Для визначення достовірності отриманих результатів в роботі [7] застосовується метод експертних оцінок, який дозволяє порівняти упроваджені підходи щодо виявлення вузликів легень. Також методологія експертних оцінок застосовується для покращення вище згаданих підходів щодо ідентифікації таких вузликів.

В дослідженні [8] виявлення потенційно важливих зон для діагностики захворювання легень здійснюється на підставі нормалізації вхідних даних, класифікації потенційних зон ураження легень за допомогою методів статистичного аналізу.

Разом з цим в роботі [9] для аналізу цифрових зображень легкого застосовується методологія вейвлет аналізу. Після чого розраховуються такі статичні атрибути як: автокореляція, ентропія, контраст та енергія. Надалі ці атрибути складають основу для побудови класифікатора імовірних уражень легкого та визначення найбільш небезпечних вузликів легкого.

Значна кількість досліджень робить наголос на упровадженні автоматичних методів діагностики захворювань легень людини та упровадженні певної класифікації таких захворювань, де також використовується методологія експертних оцінок. Прикладом такого дослідження слід вказати праці [10, 11].

Не менш важливим питанням при обробці цифрових зображень легень є застосування методології нейронних мереж. При цьому це може бути застосування:

- штучної нейронної мережі [12], яка заснована на поглибленому навчанні стосовно медичної візуалізації;
- загортальної нейронної мережі [13], яка дозволяє враховувати наявні особливості виявлення вузликів легких (клас вузликів, форму вузликів тощо).

Однак незважаючи на таку наявність різноманітних досліджень з приводу встановленої проблематики питання існують все ж таки деякі аспекти, які потребують особливого ставлення до їх розв'язання. Насамперед це виявлення вузликів легень у складних місцях їх потенційного розташування. Іншим питанням є зменшення кількості помилкових спрацювань задіяного підходу щодо розв'язку відміченого питання дослідження. Тож, саме у підсумку й визначає основні проблемні питання даного дослідження.

2. Набір зображень КТ легень як база для проведення досліджень

Отже у даній статті ми будемо розглядати цифрові зображення легень людини, які отримані в процесі процедури КТ з метою ідентифікації потенційних зон можливих захворювань. Для ідентифікації таких зон ми створюємо бібліотеку різних зображень легень людини. Далі на підставі оцінок експертів ми вибираємо ті зображення, на яких представлені характерні патологічні області ураження легких. Серед таких зображень ми також обираємо (за допомогою експертів) найбільш складні патологічні області. До таких, найбільш складних патологічних зон ураження легень ми відносимо осередки складної локалізації таких зон на зображенні [14]. При цьому ми розглядаємо ті зони, які зосереджені поблизу плеври або судин.

Приклад зображення легкого зі складною патологією в локалізації можливих осередків ураження поблизу плеври або судин подані на рис. 1 (зокрема, це утворення біля судин, які помилково можна прийняти за судини). Це півтонові зображення в діапазоні зміни точок яскравості від 0 до 255.

Отже, ми можемо говорити про необхідність вирішення досить таки складного завдання, так як потенційні зони ураження легень необхідно ідентифікувати на тлі природних перешкод – плеври або розгалуженої мережі судин. Тобто, основним завданням даного дослідження є ідентифікації складних зон ураження легень поблизу плеври і судин.

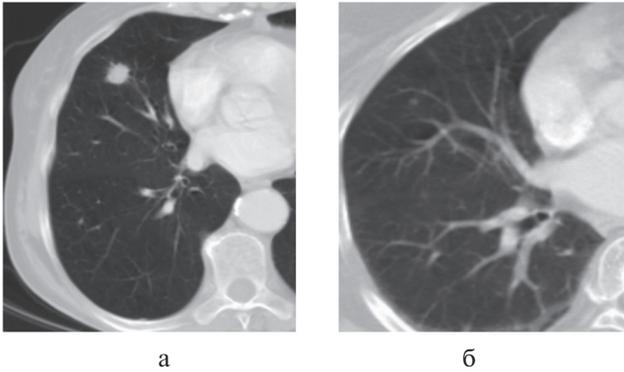


Рис. 1. Приклади складної патології щодо локалізації імовірних зон ураження легень

У той же час для ідентифікації таких зон потенційного ураження легень, як показав попередній аналіз літературних джерел, необхідно використовувати декілька різних процедур обробки відповідного цифрового зображення. Тож, окремим питанням даного дослідження постає визначення такої послідовності обробки цифрового зображення легень.

Рішення відмічених вище завдань даного дослідження передбачається на основі використання ряду процедур обробки зображень легень, отриманих в результаті КТ. При цьому однією зі складових загальної послідовності обробки цифрового зображення легень є використання ідеології нейронних мереж. Дане зауваження засновано на загальних підходах виявлення вузликів легень, що відмічають у свої працях різні науковці [5, 12, 13].

3. Енергетичні рівні зображення як вхідні дані для ідентифікації вузликів легких

З метою реалізації певного алгоритму для виявлення вузликів легкого на його цифровому зображенні необхідно, насамперед, мати деяку вхідну інформацію. Внаслідок того, що потенційні вузлики можуть бути помилково ідентифіковані як частина плеври або судин ми пропонуємо перейти до сукупності енергетичних рівнів вхідного зображення B .

Під енергетичним рівнем будемо розуміти незмінність структури досліджуваного вхідного зображення незалежно від значень заздалегідь визначеного порогу яскравості для його окремих точок.

Тобто, енергетичним рівнем будемо називати відрізок імовірних значень яскравості, де для двох будь-яких відмінних значень яскравості що належать цьому відрізку, їх функції бінаризації є однаковими.

Такий перехід дозволяє запровадити процедуру розкладання вхідного зображення на окремі енергетичні рівні, з визначенням областей інтересу для кожного такого рівня. У підсумку такий підхід

сприяє підкресленню особливостей при виявленні вузликів легень порівняно з зображенням плеври або судин на відповідних медичних зображеннях.

Під функцією бінаризації ми будемо розуміти певну функцію, яка на вхід приймає інформацію про значення яскравості окремої точки зображення та її сусідніх точок, а повертає на виході або 0, або 1. Тоді, вхідне півтонове зображення може бути розбите на h енергетичних рівнів E_i ($i = \overline{1, h}$), де деякий енергетичний рівень являє собою:

$$E(a, b) = \{Y(z) = Y(z') \forall z, z' | z \in [a, b], z' \in [a', b'], [a', b'] \subset [a, b], [a, b] \subset [0, 255]\}, \quad (1)$$

функція бінаризації Y може мати наступний вигляд [15]:

$$Y = \begin{cases} 1, \text{ якщо } \frac{1}{1 - \sqrt{\sum_{i,j=0}^{mz} (z_{ij} - \sigma(z))^2}} > p, \\ 0, \text{ у протилежному випадку,} \end{cases} \quad (2)$$

де mz – околиця точки z з різними рівнями яскравості, які знаходяться у діапазоні $z \in [a_j, b_j]$, $j = \overline{1, k}$, k – кількість енергетичних рівнів для вхідного зображення B ($k \in h$), p – припустиме значення порогу для проведення процедури бінаризації, $\sigma(z)$ – середньоквадратичне відхилення значень рівнів яскравості зображення в околиці mz точки z .

Тоді слід вказати, що функція бінаризації визначає певну множину енергетичних рівнів для вхідного зображення B :

$$B \xrightarrow{Y} \{E_i\}. \quad (3)$$

На рис. 2 (в якості прикладу) наведені окремі енергетичні рівні для зображення рис. 1а з різними параметрами бінаризації.

З метою знаходження енергетичних рівнів будеться гістограма значень яскравості точок досліджуваного зображення, яка оцінюється на підставі одношарової нейронної мережі. Тоді результатом такої процедури є вектор значень порогів для об'єднання енергетичних рівнів досліджуваного зображення у наступному вигляді:

$$E = ([e_{10}, e_{11}], [e_{20}, e_{21}], \dots, [e_{k0}, e_{k1}]), \quad (4)$$

де k – кількість енергетичних рівнів, e_{k0} – нижня межа значення порогу k -енергетичного рівня, e_{k1} – верхня межа значення порогу k -енергетичного рівня.

На кожному енергетичному рівні за допомогою ймовірнісної сегментації визначається потенційна область інтересів зон ураження легень.

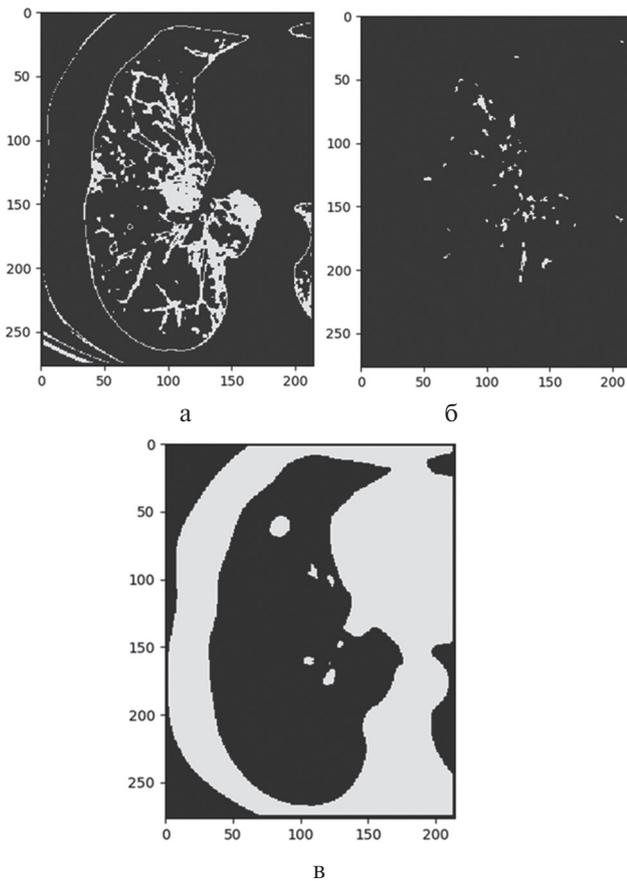


Рис. 2. Приклади енергетичних рівнів для зображення наведеного на рис. 1а –

а) $z \in [50, 75]$; б) $z \in [100, 150]$; в) $z \in [180, 220]$

На рис. 3 наведено приклади побудови гістограм для окремих енергетичних рівнів (рис. 3а та рис. 3б) для зображення на рис. 1б, зображення областей потенційного інтересу після застосування ймовірнісної сегментації на основі побудованих гістограм для окремих енергетичних рівнів (рис. 3в).

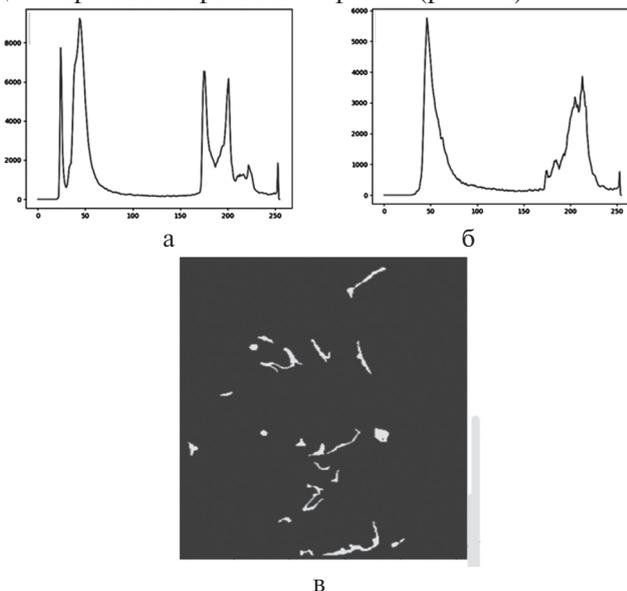


Рис. 3. Приклад побудови гістограм для енергетичних рівнів (рис. 3а та рис. 3б) та зображення областей потенційного інтересу після застосування ймовірнісної сегментації на основі побудованих гістограм (рис. 3в)

Після отримання областей потенційного інтересу для окремих зон ураження легень ми застосовуємо до таких даних загортальну нейронну мережу з метою встановлення форми областей інтересу. Якщо відповідна форма визначена як конгломерат, то ми робимо розбиття такого конгломерату шляхом визначення ознак його форми на основі багатовимірного нормального розподілу. Так відбувається до тих пір, поки на виході класифікатора бракуватиме жодного конгломерату.

4. Ідентифікація потенційних зон ураження легкого

Для ідентифікації потенційних зон ураження легкого ми використовуємо двошарову нейронну мережу (с одним прихованим шаром) [16]. На вхід такої мережі подаються дані щодо форми, класу форми, наявності меж та положення потенційної зони на зображенні легкого. На виході ми отримуємо ймовірність приналежності досліджуваної зони до потенційно небезпечної.

$$f(x, w) = w_0^{(2)} + \sum_{i=1}^m w_i^{(2)} \varphi \left(\sum_{j=1}^n x_j w_{ij}^{(1)} + w_{i0}^{(1)} \right), \quad (5)$$

де x – компоненти вхідного вектора, $x \in R^n$,

$$w = \left(\left[w_0^{(2)}, w_i^{(2)} \right], i=1, \dots, m; \left[w_0^{(1)}, w_j^{(1)} \right], j=1, \dots, n \right),$$

$$w_0^{(2)}, w_i^{(2)}, w_0^{(1)}, w_j^{(1)} -$$

набір невідомих параметрів, які оцінюються, φ – функція активації нейрону, m – кількість нейронів, n – розмірність вектору x .

Для проведення даного дослідження ми використовували декілька різних функцій активації [16]:

сигмоїдальна функція:

$$\varphi(s) = \frac{s}{1 + |s|}, \quad (6)$$

експоненціальна лінійна функція:

$$\varphi(\alpha, s) = \begin{cases} \alpha(e^s - 1), & \text{якщо } s < 0, \\ s, & \text{якщо } s \geq 0, \end{cases} \quad (7)$$

тангенс функція:

$$\varphi(s) = th(s) = \frac{(e^s - e^{-s})}{(e^s + e^{-s})}, \quad (8)$$

вигнута тотожна функція:

$$\varphi(s) = \frac{\sqrt{s^2 + 1} - 1}{2} + s. \quad (9)$$

Для уникнення надлишкових змінних, попереднє навчання було зроблено за допомогою мінімізації по w квадратичної помилки і негладкого функціоналу для згладжування такої помилки [16, 17].

З метою прискорення роботи описаної вище процедури для ідентифікації потенційних осередків ураження легкого ми також пропонуємо об'єднати різні нейронні мережі в єдину

згортально-рекурсивну нейронну мережу з 7 прихованими шарами – СРН-7СТ. За допомогою цієї мережі, в кінцевому підсумку і була встановлена багатофакторна модель для визначення зон потенційного ураження легенів.

5. Отримані результати та їх обговорення

Окремі результати роботи пропонованого підходу наведено на рис 4. На рис. 4а та рис. 4б потенційно небезпечні вузлики легкого позначено жовтим підписом. На рис. 4в потенційно небезпечний вузлик легкого виділено зеленим кольором.

Достовірність отриманих результатів було узгоджено з галузевими експертами, які мають значний досвід у виявленні потенційно небезпечних вузликів легкого.

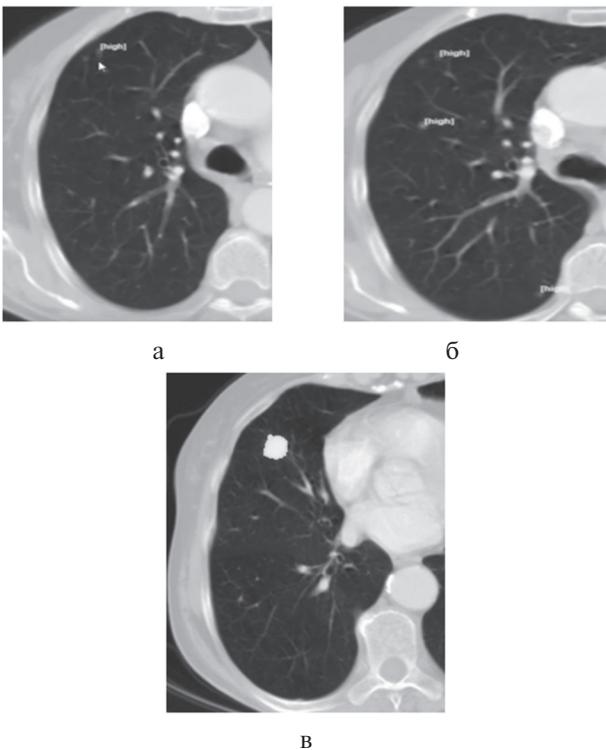


Рис. 4. Результати застосування процедури СРН-7СТ щодо визначення потенційно небезпечних вузликів легкого

Реалізація пропонованої процедури виявлення потенційних зон ураження легень була проведена для бази даних КТ легенів по 2400 зображенням. В результаті у 320 пацієнтів було виявлено 5340 легеневих вузликів різного розташування:

- 3390 легеневих вузликів мали внутріпаренхімальное розташування (63,5%),
- 1080 мали субплевральних розташування (20,2%),
- 870 легеневих вузликів були периваскулярні (16,3%).

Ми також дослідили ефективність застосування різних функцій активації нейронної мережі з точки зору отримання більш достовірних результатів

щодо виявлення вузликів легкого. Такі дані наведено у таблиці 1.

Таблиця 1

Порівняльні результати застосування процедури СРН-7СТ щодо визначення потенційно небезпечних вузликів легкого відповідно до застосування різних функцій активації нейронної мережі

Функція активації	% виявлення внутріпаренхіматозного розташування	% виявлення субплевральних розташувань	% виявлення периваскулярного розташування	% хибних спрацювань
сигмоїдальна	98.5	75.4	72.2	8
експоненціальна	96	75	61.8	12
тангенс	97.9	75.4	70	11.5
вигнута тотожна	97.5	72.5	69	12

Аналіз даних табл. 1 показує, що найбільш ефективною, з точки зору імовірності виявлення вузликів легкого та зменшення хибних спрацювань, виявляється сигмоїдальна функція активація нейронної мережі. На другому місці за ефективність розв’язку поставленого завдання виявляється тангенс функція активація нейронної мережі.

Поряд з цим в дослідженні [18] пропонується процедура що базується на застосуванні оптимальної нейронної мережі та яка визначає точність виявлення вузликів на рівні 94,56%.

Також варто зазначити, що певним обмеженням для виявлення зон ураження легень є розмір таких зон. Такі зони не повинні бути меншими за 2 мм. Порівнюючи це значення з іншими дослідженнями варто відмітити перевагу нашої процедури СРН-7СТ. Наприклад, у дослідженні [10] таке обмеження визначається розміром потенційної зони ураження на рівні 3-4 мм.

Висновки

Результати досліджень показали, що завдання ідентифікації точок потенційних зон ураження легень є досить таким складним процесом. Відмічено, що такий процес вимагає застосування різних процедур обробки, перетворення та аналізу вхідних даних. Тож з метою розв'язку поставленого завдання дослідження ми використовували процедури розкладання вихідного зображення на енергетичні рівні та бінаризації вихідного зображення, методологію нейронних мереж. Кожен етап запропонованої процедури було використано для розв'язку конкретного питання загальної задачі ідентифікації точок потенційних зон ураження легень. Як результат запропонована узагальнена процедура ідентифікації точок потенційних зон ураження легень, яка заснована на застосуванні згортально-рекурсивної нейронної мережі з 7 прихованими шарами. Проведено апробацію запропонованої процедури з різними функціями активації нейронної мережі. Отримані результати свідчать про практичність, достовірності та доцільності застосування запропонованого для ідентифікації зон ураження легень.

Список літератури

- [1] Lyashenko V., Matarneh R., Kobylin O., Putyatin Y. Contour detection and allocation for cytological images using wavelet analysis methodology // *International Journal of Advance Research in Computer Science and Management Studies*. – 2016. – № 4(1). – P. 85–94.
- [2] Lyashenko V., Babker A., Lyubchenko V. Wavelet Analysis of Cytological Preparations Image in Different Color Systems // *Open Access Library Journal*. – 2017. – № 4(7). – P. 1-9.
- [3] Schlüter S. et al. Image processing of multiphase images obtained via X-ray microtomography: a review // *Water Resources Research*. – 2014. – № 50(4). – P. 3615-3639.
- [4] Eklund A. et al. Medical image processing on the GPU—Past, present and future // *Medical image analysis*. – 2013. – № 17(8). – P. 1073-1094.
- [5] Wang Z. et al. Improved lung nodule diagnosis accuracy using lung CT images with uncertain class // *Computer methods and programs in biomedicine*. – 2018. – № 162. – P. 197-209.
- [6] Valente I. R. S. et al. Automatic 3D pulmonary nodule detection in CT images: a survey // *Computer methods and programs in biomedicine*. – 2016. – № 124. – P. 91-107.
- [7] Auffermann W. F., Little B. P., Tridandapani S. Teaching search patterns to medical trainees in an educational laboratory to improve perception of pulmonary nodules // *Journal of Medical Imaging*. – 2015. – № 3(1). – P. 011006.
- [8] Parmar C. et al. Data analysis strategies in medical imaging // *Clinical Cancer Research*. – 2018. – № 24(15). – C. 3492-3499.
- [9] Arulmurugan R., Anandakumar H. Early Detection of Lung Cancer Using Wavelet Feature Descriptor and Feed Forward Back Propagation Neural Networks Classifier // *Computational Vision and Bio Inspired Computing*. – Springer, Cham, 2018. – P. 103-110.
- [10] Dobbins III J. T. et al. Multi-institutional evaluation of digital tomosynthesis, dual-energy radiography, and conventional chest radiography for the detection and management of pulmonary nodules // *Radiology*. – 2016. – № 282(1). – P. 236-250.
- [11] Al Mohammad B., Brennan P. C., Mello-Thoms C. A review of lung cancer screening and the role of computer-aided detection // *Clinical radiology*. – 2017. – № 72(6). – P. 433-442.
- [12] Lee J. G. et al. Deep learning in medical imaging: general overview // *Korean journal of radiology*. – 2017. – № 18(4). – P. 570-584.
- [13] Suzuki K. Overview of deep learning in medical imaging // *Radiological physics and technology*. – 2017. – № 10(3). – P. 257-273.
- [14] You X. et al. CT diagnosis and differentiation of benign and malignant varieties of solitary fibrous tumor of the pleura // *Medicine*. – 2017. – № 96(49). – e9058.
- [15] Dhara A. K. et al. A segmentation framework of pulmonary nodules in lung CT images // *Journal of Digital Imaging*. – 2015. – № 29(1). – P. 148-148.
- [16] Krizhevsky A., Sutskever I., Hinton G. E. Imagenet classification with deep convolutional neural networks // *Advances in neural information processing systems*. – 2012. – P. 1097-1105.
- [17] Jung K. H., Park H., Hwang W. Deep Learning for Medical Image Analysis: Applications to Computed Tomography and Magnetic Resonance Imaging // *Hanyang Medical Reviews*. – 2017. – № 37(2). – P. 61-70.
- [18] Lakshmanaprabu S. K. et al. Optimal deep learning model for classification of lung cancer on CT images // *Future Generation Computer Systems*. – 2019. – № 92. – P. 374-382.

Надійшла до редколегії 28.11.2018



В.М. Леви́кін, О.В. Чала

ХНУРЕ, м. Харків, Україна, levykinvictor@gmail.com

ХНУРЕ, м. Харків, Україна, oksana.chala@nure.ua

МЕТОД ПІДТРИМКИ УПРАВЛІНСЬКИХ РІШЕНЬ В УМОВАХ НЕВИЗНАЧЕНОСТІ НА ОСНОВІ ТЕМПОРАЛЬНИХ ЗНАНЬ

Розглянуто проблему підтримки управлінських рішень в умовах невизначеності щодо об'єкту та процесу управління. Невизначеність щодо об'єкту управління характеризується неповнотою інформації про його стан. Невизначеність щодо процесу управління характеризується неповнотою формальних знань про цей процес внаслідок використання неформалізованих знань «за замовчуванням» при прийнятті управлінських рішень. Виділено ключові характеристики комплексного управлінського рішення як такого, що містить у собі всі альтернативи вибору послідовності дій із усунення поточної проблемної ситуації. Запропоновано темпоральну модель комплексного управлінського рішення, яка містить темпоральні правила, що дозволяють оцінити можливі переходи між станами об'єкту управління, і, на цій основі, вибрати одну із альтернатив управлінського рішення. Запропоновано метод підтримки управлінських рішень в умовах невизначеності на основі темпоральних знань. Метод забезпечує підвищення ефективності підтримки прийняття управлінських рішень за рахунок ітеративного доповнення бази темпоральних знань по мірі зміни стану об'єкту управління.

УПРАВЛІНСЬКЕ РІШЕННЯ, ПІДТРИМКА ПРИЙНЯТТЯ РІШЕНЬ, ТИМПОРАЛЬНА ЗАЛЕЖНІСТЬ, ТИМПОРАЛЬНА БАЗА ЗНАНЬ

Левыкин В.М., Чала О.В. Метод поддержки управленческих решений в условиях неопределенности на основе темпоральных знаний. Рассмотрена проблема поддержки управленческих решений в условиях неопределенности относительно объекта и процесса управления. Неопределенность относительно объекта управления характеризуются неполнотой информации о его состоянии. Неопределенность относительно процесса управления характеризуется неполнотой формальных знаний об этом процессе в результате использования неформализованных знаний «по умолчанию» при принятии управленческих решений. Выделены ключевые характеристики комплексного управленческого решения как такого, которое включает в себя все альтернативы выбора последовательности действий по устранению текущей проблемной ситуации. Предложена темпоральная модель комплексного управленческого решения, которая содержит темпоральные правила, позволяющие оценить возможные переходы между состояниями объекта управления, и, на этой основе, выбрать одну из альтернатив управленческого решения. Предложен метод поддержки управленческих решений в условиях неопределенности на основе темпоральных знаний. Метод обеспечивает повышение эффективности поддержки принятия управленческих решений за счет итеративного дополнения базы темпоральных знаний по мере изменения состояния объекта управления.

УПРАВЛЕНЧЕСКОЕ РЕШЕНИЕ, ПОДДЕРЖКА ПРИНЯТИЯ РЕШЕНИЙ, ТЕМПОРАЛЬНАЯ ЗАВИСИМОСТЬ, ТЕМПОРАЛЬНАЯ БАЗА ЗНАНИЙ

Levykin V.M. Chala O.V. Method of support of managerial decisions in conditions of uncertainty on the basis of temporal knowledge. The problem of support of managerial decisions in the conditions of uncertainty about object and process of management is considered. Uncertainty about the object of management is characterized by incomplete information about its condition. Uncertainty over the management process is characterized by incomplete formal knowledge of this process as a result of the use of informal knowledge "by default" when making managerial decisions. The key features of a comprehensive management solution are highlighted as including all alternatives for choosing a sequence of actions to eliminate the current problem situation. The temporal model of the complex management solution containing temporal rules, which allows to estimate possible transitions between the states of the object of management, and on this basis, to choose one of the alternatives of the managerial decision is proposed. The method of support of managerial decisions in conditions of uncertainty on the basis of temporal knowledge is offered. The method provides an increase in the effectiveness of supporting the adoption of managerial decisions at the expense of the iterative addition of the base of temporal knowledge as the state of the object of management changes.

MANAGEMENT SOLUTION, DECISION SUPPORT, TEMPORAL DEPENDENCE, TEMPORAL KNOWLEDGE BASE

Вступ

Ефективне управління організаційною системою у сучасному конкурентному середовищі пов'язано із виконанням частково структурованих знання-ємних задач у умовах невизначеності як щодо стану об'єкту управління, так і щодо процесу управління. Такі задачі мають детерміновані

моделі для підмножини типових варіантів реалізації. Невизначеність при використанні цих моделей виникає внаслідок неповноти інформації про параметри організаційної системи як об'єкту управління та про зовнішні впливи на цей об'єкт.

У випадку нетипових станів та нетипових режимів роботи ОУ крім невизначеності щодо об'єкту

управління виникає також функціональна й структурна невизначеність щодо процесу управління. Остання пов'язана із використаннями керівництвом та персоналом неявних персональних знань, або знань «за замовчуванням». Такі знання відображають персональний досвід, мають процедурний характер, можуть мати протиріччя та формуються із явних знань внаслідок процесу інтерналізації.

Тому реалізація управління організаційною системою в умовах невизначеності потребує послідовного вирішення задач підготовки та реалізації управлінських рішень. Задача підготовки рішень містить послідовність фаз з розпізнавання проблеми та формування набору можливих управлінських рішень для подальшого раціонального вибору. Ця задача є достатньо трудомісткою і потребує автоматизованої підтримки, що і визначає актуальність розглянутої проблеми.

Управлінське рішення (УР) є способом розв'язання проблеми управління, який представлений множиною взаємопов'язаних дій [1]. Прийняття управлінських рішень полягає у виборі особою, що приймає рішення (ОПР) однієї із альтернативних послідовностей дій для розв'язання певної проблеми управління [2].

В роботах [3, 4] розрізняють аналітичні (раціональні), а також адаптаційні та рішення на основі досвіду. Раціональні рішення використовують детерміновані моделі та придатні для підтримки структурованих задач. Адаптаційні рішення додатково використовують евристичні залежності та придатні для вирішення частково структурованих задач. Рішення на осевої досвіду базуються на неявних, неформалізованих залежностях [5] і використовуються як в частково структурованих, так і неструктурованих задачах. Підтримка рішень останніх двох видів зазвичай є знання-орієнтованою, що потребує реалізації бази знань в системах підтримки прийняття рішень (СППР).

Таку базу знань потрібно постійно підтримувати в актуальному стані. Для її поповнення використовуються комунікативні та пасивні методи інженерії знань. Перша група методів передбачає вилучення знань безпосередньо у експертів, що потребує суттєвих витрат часу і не дає можливості підтримувати базу даних в актуальному стані [6].

Пасивні методи включають інтелектуальних аналіз текстів, даних, процесів. Вони забезпечують можливість автоматизованого поповнення бази знань на основі аналізу електронних документів та інформації про поведінку об'єкту управління у базі даних організації, що формується в результаті роботи систем обробки транзакцій на оперативному рівні управління.

Аналіз документів та транзакцій в більшості випадків не дозволяє виділити каузальні залежності,

що відображають причини виконання тих чи інших управлінських дій у складі управлінського рішення. Однак такий аналіз дає можливість виділити темпоральні залежності [7, 8], які задають послідовності у часі управляючих дій та відповідні послідовності станів об'єкту управління.

Підходи до автоматизованої побудови темпоральної бази знань для підтримки управління розглянуто в роботах [9, 10]. Однак питання побудови темпоральної моделі управлінського рішення як знання-ємного процесу та підтримки формування управлінських рішень з використанням темпоральних залежностей потребують подальшого розгляду.

1. Постановка задачі

Метою статті є розробка знання-орієнтованого підходу до підтримки управлінських рішень в умовах невизначеності щодо об'єкту та процесу управління на основі аналізу інформації про зміну станів об'єкту управління у часі.

Для досягнення цієї мети розробляється темпоральна модель комплексного управлінського рішення, а також метод підтримки управлінських рішень в умовах невизначеності на основі темпоральних знань

2. Темпоральна модель комплексного управлінського рішення

Управлінське рішення призначене для усунення поточної проблемної ситуації в умовах невизначеності та представлено послідовністю управляючих дій, яка реалізує послідовність зміни станів об'єкту управління. В рамках єдиного терміну «управлінське рішення» поєднуються як формалізовані, так і евристичні, а також неформалізовані послідовності дій.

Перші ґрунтуються на явних причинно-наслідкових зв'язках, мають формальний критерій оцінки і тому забезпечують вибір оптимального за даним критерієм управлінського рішення.

Другі використовують відомі з практики залежності та враховують відомі обмеження у діяльності організації і, як наслідок, призначені для вибору раціонального рішення згідно відомих обмежень.

Третій вид послідовностей управляючих дій будується на основі неформалізованих знань про об'єкт та процес управління. Ці персональні знання ОПР або кваліфікованих виконавців можуть враховувати як відомі причинно-наслідкові залежності, поточні обмеження діяльності, а також додаткові контекстні залежності, отримані в результаті досвіду. Тому в умовах неповноти інформації щодо об'єкту управління персональні знання забезпечують можливість знайти ефективне рішення. Тобто отримані рішення в принципі можуть бути оптимальними або раціональними і тому

вони широко застосовуються на практиці. Однак внаслідок неявного характеру цих знань їх опис у традиційному формальному вигляді, наприклад через множину причинно-наслідкових зв'язків пов'язаний із значними труднощами. Це утруднює тиражування знань про управлінські рішення для забезпечення автоматизованої їх підтримки.

Запропонований темпоральний підхід до побудови загального опису управлінських рішень всіх розглянутих видів полягає у використанні зважених темпоральних залежностей. Такі залежності визначають послідовність у часі виконання дій у складі управлінського рішення. Оскільки управлінські дії послідовно змінюють стан об'єкту управління, то вказані залежності також задають послідовність зміни станів ОУ у часі. Додання ваг до темпоральних залежностей дає можливість порівнювати альтернативні варіанти управлінських рішень за сумою ваг.

Управлінські рішення згідно запропонованого темпорального підходу формуються при вирішенні задачі підготовки рішень шляхом виводу на темпоральній базі знань з урахуванням організаційної структури підприємства.

Для забезпечення раціонального вибору ОПР в рамках даної задачі будується множина альтернативних рішень. Кожна з альтернатив містить у собі послідовність управляючих дій, яка повинна забезпечити перехід організаційної системи як об'єкту управління від поточного проблемного стану до цільового стану. Тобто початковий та кінцевий стани співпадають у всіх запропонованих альтернатив. Також для підмножини альтернатив можуть співпадати проміжні стани організаційної системи, які виникають внаслідок реалізації окремих дій у складі управлінського рішення.

Зазначені особливості множини альтернативних рішень свідчать про те, що їх слід розглядати як одне комплексне управлінське рішення (КУР), яке має декілька різних реалізацій. Для такого комплексного рішення доцільно виділити ряд загальних властивостей, які не залежать від предметної області.

Ці загальні властивості базуються на виділенні темпорального аспекту об'єкту та процесу управління.

Темпоральний аспект даних та знань фіксує пов'язані з ними моменти або інтервали часу, наприклад моменти запису цих даних, їх редагування, моменти фіксації стану об'єкту управління, тощо.

Комплексне управлінське рішення, представлене з урахуванням особливостей темпорального аспекту, має такі властивості.

По-перше, структурно комплексне рішення містить множину альтернативних послідовностей

управляючих дій у часі, контекст виконання цих дій, а також залежності, що визначають реалізацію однієї з можливих послідовностей у заданому контексті.

По-друге, контекст виконання управляючих дій у КУР задається через множину станів об'єкту управління, включаючи початковий та цільовий стани. Кожен із станів характеризується множиною властивостей елементарних об'єктів (артефактів), які входять до складу організаційної системи. Підмножина значень цих властивостей визначає контекстно-залежні умови виконання відповідної управляючої дії, що входить до складу управлінського рішення.

По-третє, множина послідовностей переходів станів об'єкту управління задається у вигляді множини темпоральних залежностей між цими станами, або між управлінськими діями, кожна з яких привела до відповідного стану. Також темпоральні залежності можуть задавати контекстні зв'язки між станами та управлінськими діями.

Темпоральні залежності визначають упорядкованість станів об'єкту управління у часі. Відповідно, кожен із альтернативних варіантів управлінського рішення визначається множиною темпоральних залежностей.

В-четвертих, кожен перехід між станами ОУ, заданий в КУР у вигляді темпоральної залежності, відображає виконання однієї управлінської дії в рамках одного або декількох альтернативних управлінських рішень.

В-п'ятих, темпоральні залежності між станами об'єкту управління повинні мати числову оцінку, яка характеризує ефективність відповідної управляючої дії у складі управлінського рішення. Така оцінка дає можливість порівняти альтернативні варіанти управлінського рішення, до складу якого входять вказані залежності. Тобто інтегральна числова оцінка множини темпоральних залежностей, що визначають один із альтернативних варіантів управлінського рішення, дає можливість оцінити, наприклад ризик переходу від поточного стану до цільового для вибраного рішення.

В-шостих, темпоральне представлення комплексного управлінського рішення не містить семантики для альтернативних послідовностей дій, оскільки не враховує причинно-наслідкові зв'язки між станами об'єкту управління.

Формально темпоральна модель комплексного управлінського рішення \mathcal{M}_U має вигляд:

$$\mathcal{M}_U = (s_0, s_{aim}, S, P, R, L), \quad (1)$$

де s_0 – початковий (проблемний) стан об'єкту управління; s_{aim} – цільовий стан ОУ; S – кінцева множина станів об'єкту управління, $s_0, s_{aim} \in S$;

$P \subseteq S \times S$ – множина можливих (апріорно визначених або відомих із реалізованих управлінських рішень) переходів між станами; R – множина темпоральних правил, що визначають можливі переходи між станами для альтернативних варіантів управлінського рішення; L – оцінка темпоральних правил, що визначає їх ваги для заданих переходів між станами із P .

Кожен стан $s \in S$ характеризується множиною значень параметрів артефактів (елементарних об'єктів, що входять до складу комплексного організаційного об'єкту управління). Прикладами артефактів є документи, що обробляються в організації, обладнання, виробу, тощо. Їх параметрами є назва, стан, та інші властивості. Тоді s доцільно розглядати як множину пар (параметр a , значення v), причому в якості одного із параметрів обов'язково виступає мітка часу τ .

Темпоральні залежності R згідно представлених властивостей комплексного управлінського рішення можуть задавати зв'язок як між станами об'єкту управління в цілому, що виникають в результат виконання управляючих дій, так і між станом контексту та управляючими діями.

Темпоральні відношення між станами можуть бути визначені як для послідовних станів, так і для довільної пари станів. Відмінність між ними полягає в тому, що для послідовної пари станів задається відношення строгого порядку, а для довільної пари станів об'єкту управління – відношення часткового порядку. Тому множина темпоральних правил має такий склад:

$$R = R_X \cup R_F \cup R_U, \quad (2)$$

де R_X – множина темпоральних відношень для пар послідовних станів об'єкту управління; R_F – множина темпоральних відношень для довільних пар станів об'єкту управління; R_U – множина контекстних темпоральних відношень.

Оцінка L темпоральних правил кожному переходу між парою станів ставить у відповідність сумарну вагу W темпоральних правил, що пов'язують ці стани:

$$L: P \times R \rightarrow W. \quad (3)$$

Інтерпретація моделі темпоральної моделі комплексного управлінського рішення полягає у визначенні сумарної ваги правил, що описують один із альтернативних варіантів переходу від проблемного стану s_0 до цільового стану s_{aim} . Традиційно така інтерпретація полягає у визначеності істинності кон'юнкції логічних формул, що формалізують послідовність переходу від початкового до цільового стану.

Однак з урахуванням невизначеності при прийнятті управлінських рішень оцінка останніх має

ймовірнісний характер. Тому при виборі однієї з альтернативних реалізацій комплексного управлінського рішення доцільно порівнювати сумарну оцінку правил за умови, що така оцінка пов'язана із ймовірністю використання кожного правила в управлінському рішенні. Наприклад, для правила $r_k \in R$, що задає темпоральне відношення між станами s_i та s_j , відношення виконуваності в моделі \mathcal{M}_U має вигляд:

$$\mathcal{M}_U, (s_i, s_j) | r_k \Leftrightarrow L(p_{ij}, r_k) = w_k, \quad (4)$$

де p_{ij} – перехід між станами s_i та s_j ; w_k – оцінка залежності r_k .

Тоді інтерпретація одного із варіантів комплексного управлінського рішення еквівалентна формуванню оцінки цього варіанту і має вигляд:

$$\mathcal{M}_U, (s_i, s_j) | r_k \Leftrightarrow L(p_{ij}, r_k) = w_k, \quad (5)$$

де Q – кон'юнкція темпоральних залежностей r_k .

Згідно розглянутого вище темпорального опису абстракції процесу, комплексне управлінське рішення із наведеними характеристиками є процесом, який містить у своїй структурі знання для вибору одного із альтернативних варіантів управляючих дій. Процес із такими характеристиками є знання-ємним (knowledge intensive). Вибраний варіант управлінського рішення розглядатимемо як екземпляр знання-ємного процесу.

Вибір одного із альтернативних варіантів реалізації цього процесу виконується на основі інформації про стан контексту та персональних знань особи, що приймає рішення.

Внаслідок неповноти знань про причинно-наслідкові зв'язки в темпоральній моделі для порівняння варіантів управлінського рішення доцільно використовувати ймовірнісну оцінку. Така оцінка дає можливість визначити ризики імплементації рішення, та відповідає процесу прийняття управлінських рішень.

Комплексне рішення у формі знання-ємного процесу може бути представлено у вигляді орієнтованого графу, вершини якого відповідають станам об'єкту управління, а дуги – управлінським діям, що забезпечують перехід між цими станами.

Граф має початкову вершину, від якої відходять дуги та яка відповідає початковому проблемному стану. Також комплексне рішення має одну кінцеву вершину, до якої сходяться дуги, та яка відповідає цільовому стану.

Кожна реалізація управлінського рішення становить собою шлях по графу від початкового до кінцевого стану. Темпоральні залежності задають як окремі дуги, так і послідовність цих дуг.

Приклад такого графу наведено на рис. 1.

Кожна альтернативна послідовність, що забезпечує перехід від початкового до цільового стану, є одним із альтернативних управлінських рішень.

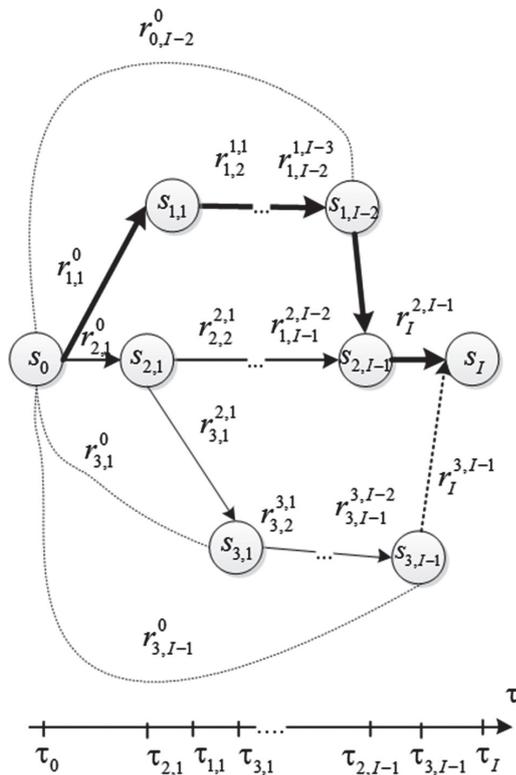


Рис. 1. Представлення комплексного управлінського рішення у вигляді направлено ациклічного графу

Подвійна індексація станів застосована для визначення поточного номеру екземпляру рішення. Наприклад, $s_{2,1}$ – перший стан другого варіанту рішення. Початковий та кінцевий стани об’єкту управління належать до всіх варіантів, тому подвійна індексація на малюнку не застосовується. На шкалі часу показані моменти реалізації цих рішень. Ця шкала відображає упорядкованість у часі, яку задають темпоральні відношення.

Запропонована модель комплексного управлінського рішення дає можливість реалізувати підтримку рішень в умовах невизначеності на основі темпоральних знань.

3. Метод підтримки управлінських рішень в умовах невизначеності на основі темпоральних знань

Підтримка управлінських рішень на основі темпоральних знань в умовах невизначеності щодо об’єкту та процесу управління повинна забезпечувати інтеграцію задач прийняття рішення та автоматизованої побудови бази знань на основі інкрементної стратегії.

Задача прийняття управлінського рішення полягає у знаходженні оптимального або раціонального способу досягнення цільового стану із поточного стану організаційної системи з урахуванням наявних ресурсів та впливу зовнішнього середовища.

Для досягнення цільового стану послідовно виконується підготовка множини альтернативних рішень та реалізація вибраного рішення.

В процесі підготовки рішення виконується розпізнавання проблемної ситуації та формування множини альтернативних управлінських рішень. Фаза розпізнавання проблемної ситуації виконується шляхом збору даних про стан організаційної системи з тим, щоб отримати контекстно-залежні патерни даних, які дозволяють виявити проблемну ситуацію. Після виявлення такої ситуації виконується класифікація існуючої проблеми. Фаза формування управлінських рішень передбачає формування критеріїв оцінки цих рішень, побудову множини альтернативних рішень у вигляді упорядкованих наборів дій, а також оцінку кожного із цих рішень за визначеним критерієм.

Реалізації управлінського рішення містить послідовність фаз із вибору та імплементацію рішення. Вибір раціонального або обмежено-раціонального рішення виконується ОПР згідно визначеного критерію оцінки та існуючих обмежень.

Імплементація рішення полягає у його впровадженні (узгодженні та виконанні), проведенні оцінки результатів, а також виконанні аналізу причини виникнення проблемної ситуації.

Для вирішення розглянутих задач використовуються: дані, які характеризують стан організаційної системи; формалізовані явні знання про об’єкти та процеси управління; неформальні знання про контекстно-залежні закономірності функціонування об’єкту управління.

Запропонований метод підтримки рішень з використанням темпоральних знань базується на результатах робіт [11,12].

В якості вхідних даних використовується інформація, про реалізовані послідовності станів об’єкту управління, отримана в результаті роботи систем обробки транзакцій.

Кожен стан організаційної системи як об’єкту управління поєднує стани підмножини елементарних об’єктів – артефактів. В якості вхідних знань використовуються залежності видів R_X , R_F , та R_U із темпоральної бази знань.

Метод містить собі такі етапи.

Етап 1. Аналіз поточної ситуації на ОУ та розпізнавання пробної ситуації. Даний етап відповідає першій фазі задачі підготовки рішень. На даному етапі виконуються такі кроки.

Крок 1.1. Збір інформації про поточну ситуацію у формі множини темпоральних залежностей, що привели від початкового стану s_0 до поточного стану s_j . Темпоральні залежності відбираються для кожної відомої послідовності $\langle s_0, \dots, s_{j-1}, s_j \rangle$.

Крок 1.2. Збір інформації у формі множини темпоральних залежностей про альтернативні послідовності станів ОУ після стану s_{j-1} .

Крок 1.3. Виявлення та класифікація проблемної ситуації на основі порівняльної оцінки темпоральних залежностей за результатами кроків 1.1 та 1.2.

Етап 2. Формування комплексного управлінського рішення у вигляді (1) з використанням темпоральних правил у випадку виявлення проблемної ситуації. На даному етапі виконується формування набору послідовностей темпоральних правил переходу від поточного стану s_j до цільового стану s_{aim} , а також оцінка цих послідовностей.

Етап містить у собі такі кроки.

Крок 2.1. Формування альтернативних послідовностей $\langle s_j, \dots, s_{aim} \rangle$ переходів між станами об'єкту управління.

Крок 2.2. Відбір з бази знань множини темпоральних правил для кожної послідовності, отриманої в результаті кроку 2.1.

Крок 2.3. Оцінка кожної послідовності $\langle s_j, \dots, s_{aim} \rangle$ на основі оцінок темпоральних правил. Отримана оцінка дає можливість порівняти ризики реалізації альтернативних варіантів управлінського рішення.

Етап 3. Побудова темпоральних залежностей для поточного стану s_j .

На даному етапі виконується актуалізація бази темпоральної знань шляхом виконання таких кроків.

Крок 3.1. Виявлення темпоральних залежностей видів R_X , R_F , та R_U , що пов'язують поточний стан s_j із попередніми станами ОУ.

Крок 3.2. Оцінка отриманих темпоральних залежностей.

Етап 4. Доповнення бази темпоральної бази знань.

Крок 4.1. Семантична перевірка експертом отриманих на кроці 3.2 темпоральних залежностей.

Крок 4.2. Уточнення множини темпоральних залежностей за результатами кроку 4.1.

Крок 4.3. Доповнення бази знань уточненою на кроці 4.2 підмножиною темпоральних правил.

Крок 4.4. Доповнення таксономії, що відображає ієрархічну організаційну структуру.

Розглянутий метод виконується при кожному переході організаційної системи як об'єкту управління до нового стану.

Висновки

Запропоновано темпоральну модель комплексного управлінського рішення, яка містить множини станів об'єкту управління, в тому числі початковий та цільовий стани, множини переходів між цими станами, множини темпоральних правил для цих переходів, а також оцінку темпоральних правил. Модель забезпечує можливість раціонального вибору управлінського рішення на основі інтегральної оцінки темпоральних правил для всіх

альтернатив, що входять до складу комплексного управлінського рішення.

Запропоновано метод підтримки управлінських рішень в умовах невизначеності на основі темпоральних знань. Метод містить у собі етапи аналізу поточної ситуації, формування комплексного управлінського рішення, побудови темпоральних залежностей та доповнення бази знань отриманими залежностями. Метод дає можливість підвищити ефективність підтримки прийняття управлінських рішень за рахунок постійного доповнення бази знань темпоральними правилами, що використовуються для оцінки цих рішень.

Список літератури:

- [1] E. Frank Harrison. The essence of management decision / E. Frank Harrison, Monique A. Pelletier // Management Decision. – 2000. – Vol. 38, Iss. 7. – P. 462 – 470.
- [2] Смирнов, Э.А. Управленческие решения / Э.А. Смирнов. – М.: Инфра-М, 2001. – 264 с.
- [3] Huber G.P. The nature of organizational decision making and the design of decision support systems. – Management Information Systems Quarterly. – 1981. – Vol. 5, Iss. 2. – P. 1-10.
- [4] Das, T.K. Cognitive biases and strategic decision processes: An integrative perspective / T.K. Das, Teng B.S. // Journal of Management Studies. – 1999. – Vol.36, Iss. 6. – P. 757–778.
- [5] Kalynychenko, O. Implementation of search mechanism for implicit dependences in process mining / O. Kalynychenko, S.Chalyi, Y. Bodyanskiy, V. Golian, N. Golian / 2013 IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS). Institute of Electrical and Electronics Engineers (IEEE), 2013.
- [6] Kendal S.L. An introduction to knowledge engineering / S.L. Kendal, M. Creen. - London: Springer, 2007. – 290p.
- [7] Chala O. Logical-probabilistic representation of casual dependencies between events in business-process management // Науково-технічний журнал, Сучасні інформаційні системи, 2018, Том 2, № 2. - С. 40-44.
- [8] Чала О. В. Побудова темпоральних правил для представлення знань в інформаційно-управляючих системах // Науково-технічний журнал, Сучасні інформаційні системи, 2018, Том 2, № 3. - С. 54-59.
- [9] Levykin V. Method of automated construction and expansion of the knowledge base of the business process management system / V. Levykin, O. Chala // EUREKA: Physics and Engineering, 2018, Vol. 4. - P. 29-35.
- [10] Левикін В. М. Концепція автоматизованої побудови бази знань у системі процесного управління / В. М. Левикін, О.В. Чала // Науково-технічний журнал «Біоніка інтелекту», 2017, № 2(89). С. 77-83.
- [11] Чала О.В. Принцип та метод еволюційної побудови бази знань на основі аналізу логів ІС процесного управління // Науково-технічний журнал «Біоніка інтелекту», 2017, № 1(88). - С. 80-84.
- [12] Левикін В. М, Чала О. Підтримка управління складним об'єктом в ІУС з використанням темпоральних залежностей. Науково-технічний журнал «Біоніка інтелекту», 2018, № 1 (90). С. 110-115.

Надійшла до редколегії 20.09.2018

УДК 004.6; 550.8.05

Р.М.Камінський¹, Н.Б.Шаховська², Б.Т.Ладанівський³, Л.Г.Савків⁴¹Національний університет «Львівська політехніка», kaminsky.roman@gmail.com²Національний університет «Львівська політехніка», nataliya.b.shakhovska@lpnu.ua,³Карпатське відділення Інституту геофізики Національної академії наук України, bogys@cb-igph.lviv.ua⁴Карпатське відділення Інституту геофізики Національної академії наук України

ПОБУДОВА МОДЕЛІ ГЕОЕЛЕКТРИЧНОГО РОЗРІЗУ ЗА РЕЗУЛЬТАТАМИ ФОРМАЛЬНОЇ ІНТЕРПРЕТАЦІЇ ДАНИХ МЕТОДУ ЗОНДУВАННЯ СТАНОВЛЕННЯМ ПОЛЯ В БЛИЖНІЙ ЗОНІ

Стаття присвячена розробленню методу визначення характеристичних показників геоелектричних шарів та побудови за цими даними моделі георозрізу. Це дозволить швидко побудувати наближені моделі геоелектричного розрізу з використанням простої комп'ютерної техніки, що є особливо актуальне в умовах експериментальних польових робіт. Розроблений метод розрахунку параметрів приблизної моделі геоелектричного розрізу за табличними даними отриманих на етапі формальної інтерпретації кривої електричного опору. На основі запропонованого методу викладений алгоритм та створена програма розрахунку, описані вхідні та вихідні дані, форма представлення результату.

ПОЛЬОВІ ДОСЛІДЖЕННЯ, ГЕОФІЗИЧНІ ДАНІ, ФОРМАЛЬНА ІНТЕРПРЕТАЦІЯ, ЕКСПРЕС-АНАЛІЗ, МЕТОД ЗОНДУВАННЯ СТАНОВЛЕННЯМ ПОЛЯ В БЛИЖНІЙ ЗОНІ, МОДЕЛЬ ГЕОЕЛЕКТРИЧНОГО РОЗРІЗУ

Kaminsky R., Shakhovska N., Ladanivsky B., Savkiv L. Construction of the geoelectric cutting model by final interpretation of the data processing method for the position of the field in a neighborhood zone. The article is devoted to the development of a method for the determination of characteristic indices of geoelectric layers and the construction of these geocoding models based on these data. This will allow to quickly build approximate models of the geoelectric section using simple computer technology, which is especially relevant in the field of experimental field work. A method for calculating the parameters of an approximate model of a geoelectric section is developed based on tabular data obtained at the stage of the formal interpretation of the electric resistance curve. On the basis of the proposed method, the algorithm is presented and a calculation program is created, the input and output data are described, the form of presentation of the result.

FIELD STUDIES, GEOPHYSICAL DATA, FORMAL INTERPRETATION, EXPRESS ANALYSIS, METHOD OF PROCESSING FOR FIELD DEVELOPMENT IN A NEIGHBORHOOD ZONE, GEOELECTRIC CUTTING MODEL

Каминский Р., Шаховська Н., Ладанивский Б., Савкив Л. Построение модели геоэлектрических разрезов по результатам формальной интерпретации данных метода зондирования становления поля в ближней зоне. Статья посвящена разработке метода определения характеристических показателей геоэлектрических слоев и построения по этим данным модели георозреза. Это позволит быстро построить приближенные модели геоэлектрических разрезов с использованием простой компьютерной техники, особенно актуален в условиях экспериментальных полевых работ. Разработанный метод расчета параметров приближенной модели геоэлектрических разрезов по табличными данными полученных на этапе формальной интерпретации кривой электрического сопротивления. На основе предложенного метода выложенный алгоритм и создана программа расчетной, описаны входные и выходные данные, форма представления результата.

ПОЛЕВЫЕ ИССЛЕДОВАНИЯ, ГЕОФИЗИЧЕСКИЕ ДАННЫЕ, В ФОРМАЛЬНОМ ИНТЕРПРЕТАЦИЯ, ЭКСПРЕС-САНАЛИЗ, МЕТОД ЗОНДИРОВАНИЯ СТАНОВЛЕНИЯ ПОЛЯ В БЛИЖНЕЙ ЗОНЕ, МОДЕЛЬ ГЕОЭЛЕКТРИЧЕСКИХ РАЗРЕЗАХ

Вступ

Для дослідження та вивчення фізичних процесів і явищ, що відбуваються у поверхневих шарах та надрах Землі, а також на її поверхні і в навколосферному просторі використовують різноманітні геофізичні методи та методики. Такі дослідження організовуються у різний спосіб: це можуть бути систематичні наукові дослідження у формі безперервного моніторингу чи режимних геофізичних спостережень, або ж періодичні вивчення окремих регіонів чи площ. Кожен з таких напрямків дає

важливу первинну інформацію для подальшого вивчення цих процесів.

Безперервні геофізичні спостереження ведуться постійно в стаціонарних та тимчасових пунктах спостережень. На сьогоднішній час, збір та реєстрація фізичних полів відбувається переважно автоматично, шляхом створення і поповнення різноманітних баз даних. За даними регулярних та багаторічних спостережень можна оцінювати та аналізувати параметри полів та їх характеристики, вивчати динаміку змін з протягом тривалого

періоду часу, контролювати критично-важливі показники. Зазвичай подібні дослідження забезпечують міжнародні мережі, світові та національні центри, консорціуми, геологічні, екологічні та інші служби, об'єднання, агентства. Згадані організації, окрім інших типів даних, працюють з геомагнітними, сейсмічними, магнітотелуричними спостереженнями, формують відповідні архіви, подають візуальне представлення таких даних в режимі online і, крім того, надають спеціальний або безпосередній доступ до них

Польові геофізичні дослідження електромагнітними методами дозволяють вивчати локальні області або регіони. Такі роботи проводяться як вздовж профілю, так і на окремій площі для виявлення, дослідження чи прогнозування небезпечних геодинамічних явищ: зсуви, карстові процеси, руйнування гребель водо- та сховищ, тощо. Незалежно від характеру проведення спостережень, профільні це чи площинні, предметом досліджень є природні або штучні фізичні поля, а об'єктами досліджень виступають: елементи соціальної інфраструктури (школи, навчально-виховні комплекси, автодороги, майданчики будівництва), відпрацьовані шахти, дамби хвостосховищ, екологічно-проблемні території гірничо-промислових агломерацій.

Для приповерхневих вивчень верхньої частини геологічного розрізу використовують і метод зондування становленням електромагнітного поля у ближній зоні (ЗСБ, TDEM, TEM). Остаточні висновки щодо досліджуваної території формуються на основі розширеного аналізу, детальної обробки та інтерпретації польового матеріалу. Однак, для швидкої оцінки геодинамічної ситуації, особливо на поточній точці спостережень, важливим є експрес-аналіз первинних геофізичних даних.

1. Постановка проблеми

Результатом профільних чи площинних електророзвідувальних спостережень методом ЗСБ є реєстрація перехідних процесів [1, 2, 3, 4].

Отримані інформаційні сигнали трансформуються у параметри середовища – електропровідність, позірний електричний опір, глибина, значення яких у подальшому використовуються для побудови геоелектричних розрізів, що характеризують геологічну будову та структурні особливості досліджуваного регіону [5, 6, 7]. Очевидно, що такий аналіз потребує затрат часу і зусиль кваліфікованих спеціалістів, відповідної обчислювальної техніки. Проте доволі часто буває так, що необхідно швидко оцінити геодинамічну ситуацію за даними спостережень, тобто виникає потреба отримати спрощені геоелектричні розрізи. Насамперед це є актуальним в полі, коли необхідно оперативно проаналізувати зібрані дані на поточній точці профілю і на основі такого аналізу прийняти рішення про подальші вимірювання, а саме – для уточнення відстані до наступної точки спостережень чи зміну параметрів вимірювальної установки.

Залежно від геометрії досліджуваної території, вимірювання можуть проводитись або вздовж профілю, або охоплювати певну площу. Точки спостережень (підмети ПК) прокладають з однаковим розрахунковим кроком a , c (рис. 1), виміри відбуваються послідовно точка за точкою.

Якщо геодинамічна ситуація на сусідніх чи попередніх пікетах дослідження була незмінна, доцільним є відстань до наступної точки збільшити. Якщо ж параметри геологічного середовища (опір, глибина) вже помітно змінилися, або принаймні появилася така тенденція, крок подальших вимірів варто зменшити, причому, можливо, і доречним буде повернутися до попереднього пікету з меншим кроком для уточнення і деталізації геологічної ситуації на цій точці. Таким чином, змінюючи ще в процесі вимірів схему розташування пікетів від теоретичного регулярного до реального ситуативного, отримуємо можливість завжди мати максимально точні та правдиві дані вздовж профілю, а за окремих умов, коли петлі мають довжину 100 і більше метрів, такий підхід ще й суттєво зменшить

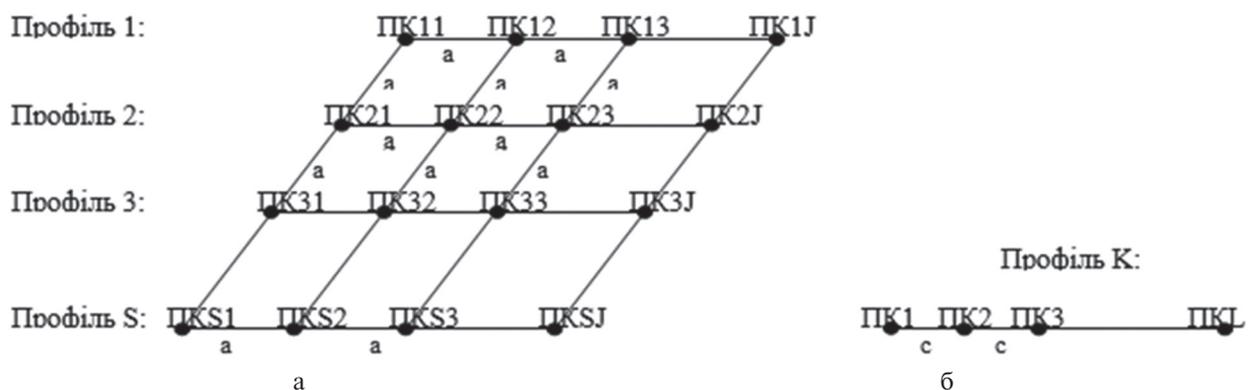


Рис. 1. Схема розташування пікетів при:

а – площинних та б – профільних дослідженнях; а, с – крок між точками спостережень

час проведення польових досліджень. Окрім цього, чим точнішими є результати польових вимірювань загалом, тим точнішою є обробка та інтерпретація даних, а разом з тим – і самі геоелектричні розрізи досліджуваної території. З огляду на це побудова приблизної моделі геоелектричного розрізу за даними ЗСБ має важливе науково-прикладне значення та є актуальною задачею.

Метою цієї статті є розробка методу визначення характеристичних показників геоелектричних шарів та побудови за цими даними моделі георозрізу.

2. Огляд літератури

Класична інтерпретація польових даних методу ЗСБ є складною і тривалою процедурою, яка включає поетапні, багатокрокові розрахунки і розв'язки прямих та обернених задач. Викладені у літературних джерелах методики описують процес обробки починаючи від ручної побудови на подвійних логарифмічних бланках кривих зондувань (із зареєстрованих імпульсів становлення поля), розрахунки узагальнених параметрів розрізу і підбір такої моделі геоелектричного розрізу, щоб теоретична та експериментальна криві збігалися якнайкраще, розробляються і свої методики трансформації інформаційних сигналів ЗСБ в параметри середовища [7, 4, 6, 5, 8, 9].

Характерним для згаданих робіт є те, що здебільшого в них описані основні формули для розрахунків, викладені загальні підходи щодо обробки та інтерпретації, проте у жодній не висвітлені питання швидкого експрес-аналізу польових даних з детальним покроковим алгоритмом практичної реалізації.

3. Матеріали та методи

Суть методу ЗСБ, застосування. Метод зондування шляхом становлення електромагнітного поля у ближній зоні джерела поля часто застосовується при розв'язанні задач структурно-пошукового, інженерно-геологічного та гідрогеологічного спрямування. Він дозволяє з достатньо високою роздільною здатністю визначати електричні параметри середовища та літологічний склад порід, виявляти структурні та тектонічні особливості геологічного середовища на відносно невеликих глибинах. Метод ЗСБ заснований на явищі електромагнітної індукції, належить до методів зі штучним (контрольованим) джерелом і полягає у вивченні поля перехідних процесів, що виникає в землі при зміні струму в джерелі [6, 4, 10]. За вимірними експериментальними значеннями з простих побудов кривих становлення можна аналізувати первинні геофізичні дані, проводити формальну інтерпретацію, суть котрої полягає у розрахунку залежності

поздовжньої сумарної провідності $S(h)$ та питомого електричного опору $\rho(h)$ від глибини [7, 9].

Побудова моделі геоелектричного розрізу. Точніші параметри геологічного середовища (опір, глибина, товщина, кількість шарів) можна отримати з моделі геоелектричного розрізу. Для побудови такої моделі беруться до уваги результати формальної інтерпретації, а саме – крива опору і як допоміжна – крива провідності.

Традиційною моделлю для методів зондування є модель горизонтально-шаруватого середовища, оскільки суттєво властивості геологічного середовища змінюються по вертикалі як в планетарному масштабі, так і при детальних дослідженнях. З урахуванням цього факту і будуються моделі. На рис. 2 і рис. 4 подані 2 варіанти побудови моделі: схематична і точна.

Схематична побудова моделі. Схематична побудова моделі геоелектричного розрізу є доволі простою та інтуїтивно зрозумілою і проводиться на основі візуальної оцінки кривих поздовжньої сумарної провідності $S(h)$ та питомого електричного опору $\rho(h)$, значення яких отримуємо на етапі формальної інтерпретації даних. Спрощене подання такої процедури представлено на рис. 2.

Точна побудова моделі. Схематично модель геоелектричного розрізу будується за даними формальної інтерпретації. З кривих поздовжньої сумарної провідності $S(h)$ та питомого електричного опору $\rho(h)$, що представлені на рис. 2, наочно можна бачити розподіл середовища на шари, а також – приблизні їхні товщини. Питомі опори шарів – це найбільші та найменші значення кривої $\rho(h)$, орієнтовні границі шарів – точки перегину цієї ж кривої. Хоча описана процедура і є достатньо очевидною, проте скористатися нею для автоматизованого розрахунку та експрес-аналізу даних практично неможливо. Лише знаючи аналітичний вираз функції, можна знайти згадані величини. Однак, на практиці, зазвичай, залежність одного параметра від іншого часто представляється набором експериментальних даних, при цьому аналітичний вигляд цієї залежності, як і результатів формальної інтерпретації, невідомий. У таких випадках мають справу з інформацією, що подається у вигляді таблиці, тобто маємо табличне представлення даних. З огляду на це, пропонується метод розрахунку параметрів приблизної моделі геоелектричного розрізу за табличними даними інтерпретації.

Метод розрахунку параметрів приблизної моделі геоелектричного розрізу за табличними даними. Суть методу полягає у тому, що характеристичні параметри геологічного середовища, тобто параметри

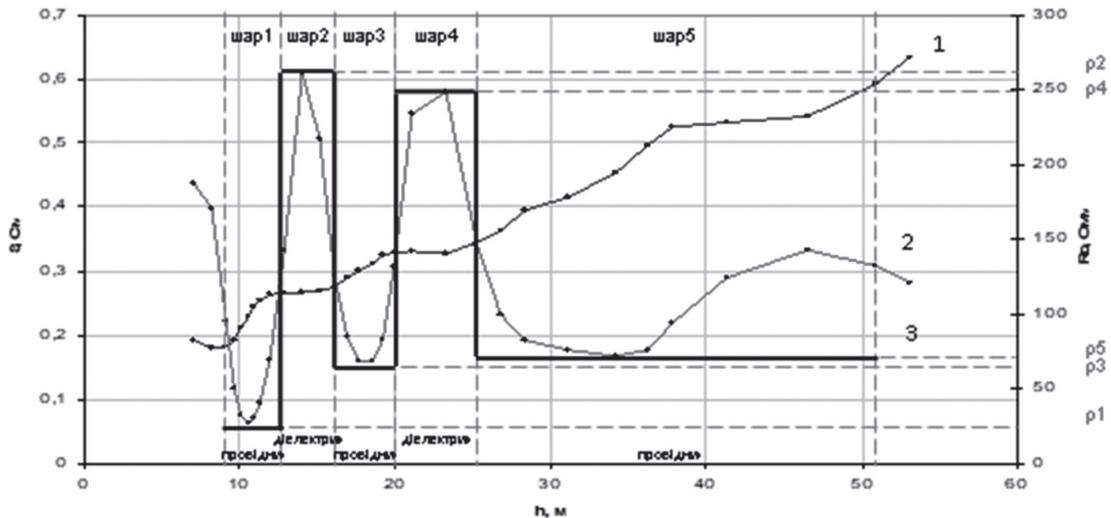


Рис. 2. Схематична побудова моделі

(1 – крива поздовжньої сумарної провідності $S(h)$, 2 – крива питомого електричного опору $\rho(h)$, 3 – модель геоелектричного розрізу)

моделі геоелектричного розрізу, розраховуються з отриманої на етапі формальної інтерпретації кривої електричного опору із застосуванням для знаходження потрібних масивів питомих опорів і відповідних їм глибин першої та другої похідної кривої $\rho(h)$. Результат формальної інтерпретації – це масиви значень опорів та глибин. Знайти похідні за даними у табличному представленні можна двома шляхами: перший – через пошук деякої приблизної функції, яка б замінила табличні значення. І другий – пошук похідних за сусідніми точками, що базується на фізичному змісті самої похідної, тобто те, що перша похідна – це швидкість, котра розраховується як відношення різниць ординат сусідніх точок до різниці їх абсцис: $\approx \Delta y / \Delta x$. Перший спосіб знаходження похідної для швидкої обробки чи експрес-аналізу не є цілком прийнятний. У другому варіанті для приблизного обчислення похідних передбачається використання чисельного диференціювання з допомогою методу кінцевих різниць. У цьому випадку похідні виступають як швидкості зміни функції, тобто визначаються відношенням приросту функції до приросту її аргументу і розраховуються через елементарні операції віднімання та ділення. Якщо функція задана масивами значень, а результатом формальної інтерпретації є саме такі набори даних (рис. 2), похідну у точці x_i можна знайти за формулою: $f'(x_i) = (y_{i+1} - y_i) / (x_{i+1} - x_i)$. Такий підхід для розрахунку похідних з використанням найпростіших математичних операцій для експрес-аналізу є повністю задовільний.

Для прикладу, за даними формальної інтерпретації розрахуємо першу та другу похідну кривої $\rho(h)$ (рис. 3).

На основі отриманих значень точну побудову моделі представимо на рис. 4.

D12		= (C12-C11)/(A12-A11)				
	A	B	C	D	E	F
1	h (м)	S (См)	Ro (Омм)	1похідна	2похідна	
2	7,08	0,19	187,14			
3	8,19	0,18	170,99	-14,43		
4	9,14	0,18	95,11	-80,40	-69,89	
5	9,64	0,19	49,81	-89,97	-19,02	
6	10,07	0,21	32,03	-41,96	113,29	
7	10,54	0,23	27,22	-10,12	67,07	
8	10,94	0,24	29,47	5,65	39,61	
9	11,34	0,25	40,62	27,51	53,94	
10	11,97	0,26	69,12	45,66	29,07	
11	12,92	0,27	142,53	77,48	33,60	
12	14,02	0,27	261,19	107,62	27,33	
13	15,13	0,27	217,14	-39,53	-132,03	
14	16,13	0,28	125,79	-91,29	-51,74	

Рис. 3. Розрахунок похідних кривої $\rho(h)$

Опис запропонованого методу:

1) розраховується крива питомого електричного опору $\rho(h)$, її перша та друга похідна (методом кінцевих різниць);

2) масиви значень Y та X для побудови моделі геоелектричного розрізу беруться з кривої $\rho(h)$ з урахуванням першої та другої похідних, а саме:

– значення Y – значення питомих опорів шарів ρ – береться з кривої $\rho(h)$ в точках її екстремумів (значення y);

– значення X – значення границь шарів h – береться з кривої $\rho(h)$ в точках її перегину (значення x), тобто в точках, де x міняє значення;

3) для пошуку похідних, рівних 0 (точки переходу через 0), достатньо скористатися лінійною інтерполяцією на основі двох сусідніх точок;

4) на етапі пошуку значень Y питомих опорів шарів пропонується для отримання максимально точного результату спершу криву $\rho(h)$ в інтервалах між дискретними розрахованими значеннями інтерполювати сплайнами.

Запропонований алгоритм цілком придатний для швидкої обробки чи експрес-аналізу експериментальних даних і складений таким чином, щоб його легко можна було запрограмувати,

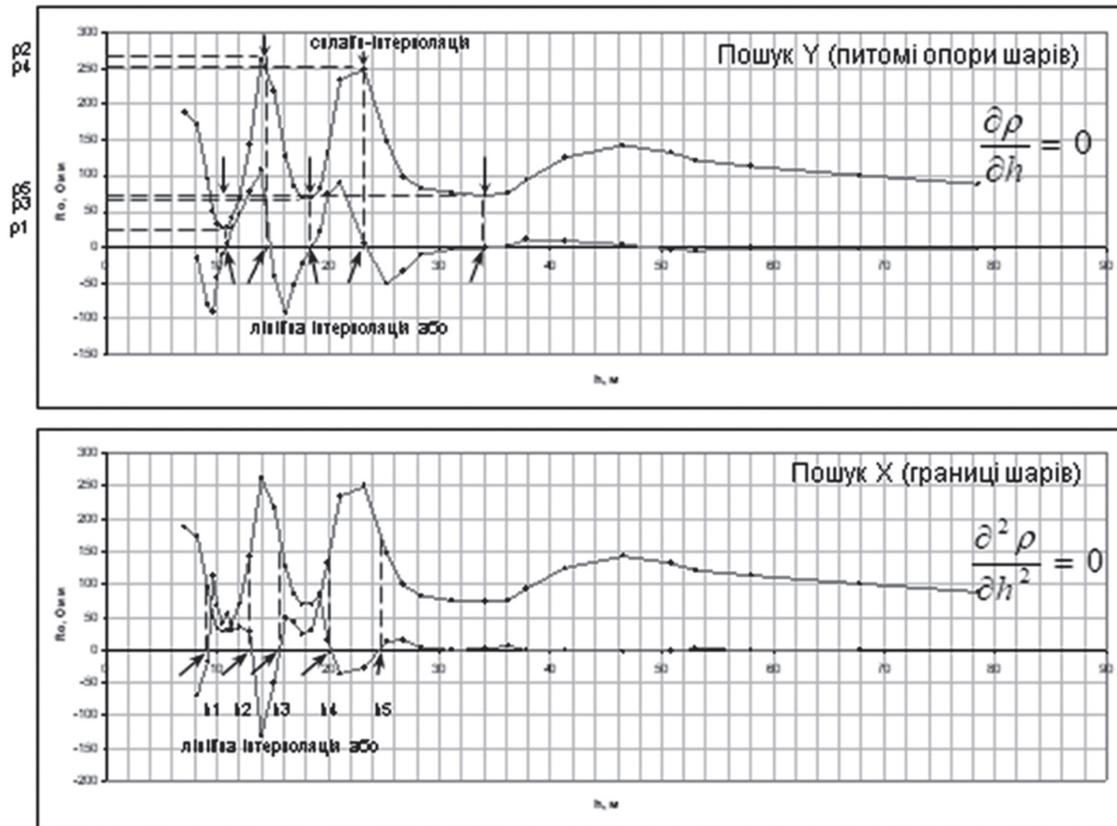


Рис. 4. Точна побудова моделі

використавши нескладні математичні операції та функції. До прикладу, у пакеті Octave, що є вільним програмним забезпеченням і поширюється під ліцензією GNU GPL, серед вбудованих функцій є чимало для одномірної інтерполяції з можливістю вибору потрібного методу [11]. Так, для лінійної інтерполяції можна скористатися функцією `interp1()` або розрахувати потрібне значення з рівняння прямої за двома точками самостійно, для сплайн-інтерполяції з урахуванням неперервності першої та другої похідної для найкращого результату варто обрати функцію `spline()`.

На основі викладеного методу у середовищі GNU Octave розроблена програма розрахунку параметрів геологічного середовища для побудови моделі геоелектричного розрізу (рис. 5).

Вхідними даними для цієї програми є результати формальної інтерпретації: масив глибин (h) та відповідний їм масив питомого електричного опору (R_0), що представляють собою криву залежності (h) (рис. 2). Саме ці величини використовуються для подальших розрахунків, знаходження першої і другої похідних, пошуку опорів шарів та глибин залягання. Вихідними даними є знайдені масиви електричного опору, приблизних меж шарів, а також – сформовані остаточні масиви значень для побудови приблизної моделі геосередовища. Результат роботи програми представлений на рис. 5б – це графік-модель геоелектричного розрізу,

збережений у форматі png. Засоби пакету GNU Octave підтримують роботу з багатьма іншими форматами зображень.

4. Обговорення

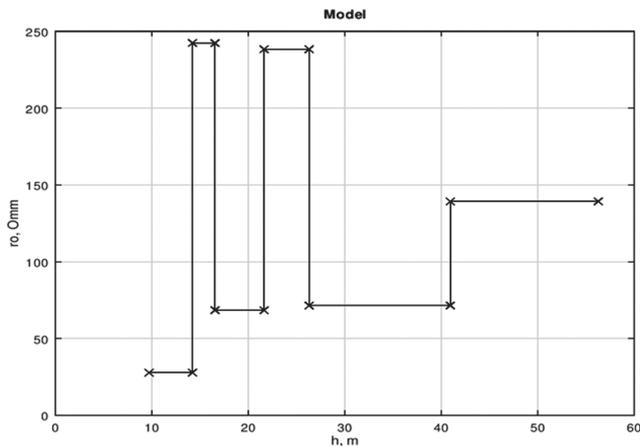
Зазначені особливості є важливими при вирішенні практичних задач чи роботі з даними експериментальних польових робіт, а визначення характеристичних показників геоелектричних шарів описаним методом та їх уточнення за рахунок зміни кроку спостережень чи параметрів вимірювальної установки надасть можливість для швидкої побудови моделі георозрізу.

Оскільки тривалість виконання польових робіт залежить і від оперативної оцінки геодинамічної ситуації безпосередньо на поточній точці спостережень, запропонований метод з використанням простих розрахунків, що не потребує потужного програмного забезпечення чи комп'ютерної техніки, легко і доволі ефективно можна буде використовувати для швидкої експрес-інтерпретації як у стаціонарних умовах, так і особливо в умовах польових робіт. Це дозволить оптимально спланувати подальші профільні чи площинні вимірювання, оперативно оцінити геодинамічну ситуацію, скоротити загальний час досліджень на тій чи іншій території, отримати точніші географічно первинні дані спостережень, а загалом – і достовірніші результати обробки та інтерпретації.

```

#!/usr/bin/octave
#
#####
fName = argv{1};
A = load (fName);
N = size(A,1);

# dRho/dh
for i = 2:N
    p1(i)=(A(i,5)-A(i-1,5))/(A(i,3)-A(i-1,3));
endfor #
N = size(p1,2);
j=0;
for i = 2:N-1
    if (sign(p1(i)) != sign(p1(i+1)))
        j=j+1;
        x(j)=(A(i+1,3)-A(i,3))*(-p1(i))/(p1(i+1)-
            p1(i))+A(i,3);
    endif
endfor
ro=spline(A(:,3),A(:,5),x);
# d2Rho/dh2
for i = 3:N
    p2(i)=(p1(i)-p1(i-1))/(A(i,3)-A(i-1,3));
endfor
    
```



а б
Рис. 5. Уривок програми розрахунку (а) та результат роботи (б) – модель геоелектричного розрізу за даними формальної інтерпретації

Висновки

За результатами проведеної роботи можна зроби такі висновки:

- розроблений метод розрахунку параметрів приблизної моделі геоелектричного розрізу за табличними даними отриманих на етапі формальної інтерпретації кривої електричного опору;
- на основі запропонованого методу викладений алгоритм та створена програма розрахунку, описані вхідні та вихідні дані, форма представлення результату;
- середовищем розробки програмних реалізацій обраний безкоштовний спеціалізований математичний пакет з ліцензією GNU GPL – Octave.

Запропонований в роботі метод призначений для роботи з геофізичними даними методу зондування становленням електромагнітного поля в ближній зоні з вимірювальною установкою „петля в петлі”. Розрахунок параметрів моделі геоелектричного розрізу ведеться на основі кривої питомого електричного опору. Характерними особливостями методу є:

- можливість знаходження параметрів моделі за даними, представленими таблично;
- для забезпечення найкращого результату розрахунку пропонується використання відповідних методів інтерполяції в залежності від ситуації;
- застосування в ході обчислень простих формул, нескладних розрахунків та елементарних математичних операцій.

Список літератури:

[1] M.N. Nabighian and J.C. Macnae Time Domain Electromagnetic Prospecting Methods, Chapter 6 in Electromagnetic Methods in Applied Geophysics, Volume 2, Application, Parts A and B. Edited by Misac N. Nabighian, Published January 1, 1991, <https://doi.org/10.1190/1.9781560802686.ch6> , pages 427-520.

[2] George V. Keller Principles of time-domain electromagnetic (TDEM) sounding. The Leading Edge. Vol.16.; Issue. 4.; Pages. 355-357 (Issue publication date: April 1997), <https://doi.org/10.1190/1.14337632>.

[3] L.L. Vanyan Electromagnetic Depth Soundings. Translated from Russian by George V. Keller. Consultants Bureau, New York, 1967, pages 312.

[4] Б.К. Матвеев Электроразведка: Учеб. для вузов. – 2-е изд., перераб. и доп. – М.: Недра, 1990 – 368 с.

[5] Толстой М.І., Гожик А.П., Рева М.В., Степанюк В.П., Сухорада А.В. Основи геофізики (методи розвідувальної геофізики): Підручник. – К.: Видавничо-поліграфічний центр „Київський університет”, 2006. – 446 с.

[6] Электроразведка: пособие по электроразведочной практике для студентов геофизических специальностей. Под редакцией проф. В.К. Хмелевского, доц. И.Н. Модина, доц. А.Г. Яковлева – М.: 2005. – 311 с.

[7] Сидоров В.А., Тикшаев В.В. Электроразведка зондированиями становлением поля в ближней зоне – Саратов, 1969. – 68 с.

[8] Рева М.В. Трансформація інформативних сигналів становлення поля в ефективні геоелектричні параметри з використанням інтегральних часових характеристик перехідних процесів / М.В. Рева, Т.В. Руденко, Н.С. Єгорова, Л.П. Андрієвська // Збірник наукових праць „Теоретичні та прикладні аспекти геоінформатики”. – 2010. – Вип.. 7. – С. 110-129.

[9] Сидоров В.А. Импульсная индуктивная электроразведка – М.: Недра, 1985. – 192 с.

[10] Савків Л.Г. Розробка структур файлів для оптимального відображення первинних електромагнітних даних геофізичних досліджень // Вісник Національного університету „Львівська політехніка”. Інформаційні системи та мережі. – 2018. – № 887. – С. 105–115.

[11] GNU Octave [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу: <https://octave.org/doc/v4.0.0/index.html> (дата звернення 17.11.2018) – Назва з екрана.

Надійшла до редколегії 03.10.2019

УДК 004.932.2



Е.Д. Михнова

ХНТУСХ, 61002, ул. Алчевских 44, Харьков, Украина, mikhnova@yahoo.co.uk

ИЗВЛЕЧЕНИЕ МУЛЬТИМЕДИЙНЫХ ДАННЫХ С НЕЧЕТКИМ СООТВЕТСТВИЕМ

С резким увеличением объемов общедоступной мультимедийной информации, возрастает потребность ее обработки в реальном времени. В статье проанализированы проблемы обработки всех типов мультимедийного контента с использованием современных методов искусственного интеллекта, особое внимание уделено извлечению данных из памяти переводов с нечетким соответствием поисковому запросу. Основываясь на теории полугрупп, предложены математические формулировки для представления и извлечения данных с нечетким соответствием. Предложенные выкладки могут с успехом применяться не только для поиска видео и текстовых данных, но и для анализа биомедицинской информации.

ПОИСК МУЛЬТИМЕДИЙНЫХ ДАННЫХ, ПАМ'ЯТЬ ПЕРЕКЛАДОВ, РОЗПІЗНАВАННЯ УСНОЇ ТА ПИСЬМОВОЇ МОВИ, ОБРОБКА ЗОБРАЖЕНЬ, ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ВІДЕОРЯДУ, ТЕОРІЯ ПОЛУГРУПП

О.Д. Михнова. Вилучення мультимедійних даних з нечіткою відповідністю. Із значним збільшенням обсягів загальнодоступної мультимедійної інформації, зростає потреба її обробки у реальному часі. Стаття аналізує проблеми обробки усіх типів мультимедійного контенту з застосуванням сучасних методів штучного інтелекту, особливу увагу приділено вилученню даних із пам'яті перекладів з нечіткою відповідністю до пошукового запиту. Базуючись на теорії напівгруп, запропоновано математичні формулювання для подання та вилучення даних з нечіткою відповідністю. Запропоновані викладки можуть успішно використовуватись не тільки для пошуку відео і текстових даних, але й для аналізу біомедицинської інформації.

ПОШУК МУЛЬТИМЕДИЙНИХ ДАНИХ, ПАМ'ЯТЬ ПЕРЕКЛАДІВ, РОЗПІЗНАВАННЯ УСНОЇ ТА ПИСЬМОВОЇ МОВИ, ОБРОБКА ЗОБРАЖЕНЬ, ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ВІДЕОРЯДУ, ТЕОРІЯ НАПІВГРУП

O.D. Mikhnova. Multimedia data extraction with fuzzy match. With the rapid growth of generally available multimedia data, its real-time processing becomes an urgent problem. This article analyses all types of multimedia content processing with the help of artificial intelligent methods, special attention is given to translation memory data extraction with fuzzy match to a search query. Mathematical groundings based on semi-group theory are provided for content matching. The proposed formulations can be successfully used in video and text search applications as well as biomedical information processing.

MULTIMEDIA DATA SEARCH, TRANSLATION MEMORY, SPEECH AND TEXT RECOGNITION, IMAGE PROCESSING, INTELLIGENT ANALYSIS OF VIDEO SERIES, SEMI-GROUP SET THEORY

Введение

Под мультимедийными данными принято понимать комбинацию аудио и визуальной информации, которая вместе может формировать видеоряд в случае постепенного изменения содержания, объединенного одной сюжетной линией. Текстовая составляющая также может присутствовать в видеоданных в виде графической информации, текст также является неотъемлемой частью речевых данных. Настоящая статья оперирует со всеми разновидностями мультимедийного контента с точки зрения сегментации и сопоставления последовательностей. Предложены математические формулировки четкого и нечеткого соответствия для поисковых приложений. В этом разделе статьи кратко описаны существующие методы обработки и распознавания мультимедиа, что является основой для следующего раздела с математическими выкладками и новыми предложениями.

Чтобы приступить к изучению существующих методов интеллектуальной обработки текстовых данных стоит упомянуть, что на момент написания этой статьи население земного шара разговаривает на 7100 языках приблизительно, не говоря

о диалектах. Большая часть алгоритмов обработки естественного языка предназначена для широко используемых языковых семей, для таких языков как китайский, английский, испанский, арабский, русский, немецкий и др. [1] Стоит отметить, что настоящее исследование нацелено на романо-германскую и восточно-славянскую языковую пару. В отличие от китайского и других языков на основе иероглифов, здесь нет необходимости значительной трансформации структуры предложений и перестановки порядка слов во время перевода. В работе изучены последние тенденции сопоставления текстовых последовательностей в лидирующих инструментах на основе памяти переводов и терминологических баз (SDL Trados Studio, Star Transit, Atril Deja Vu, MemoQ, MemSource).

Полностью автоматизированные системы перевода, такие как Google Translate и Promt, предполагают морфологический анализ с последующей лексической трансформацией с использованием двуязычных словарей, затем производится локальное изменение порядка слов, и наконец, морфологическая генерация на целевом языке. Такие системы иногда обогащены сематическими

анализаторами и генераторами, включающими так называемое интерлингвальное представление. Стоит отметить, что целью настоящего исследования не является ни выделение частей речи (это может быть реализовано с помощью широко известной скрытой модели Маркова, марковской модели максимальной энтропии, рекуррентной нейронной сети), ни анализ формальных грамматик. Во внимание принимаются только термины и сегментированные видеокadres такими, как они есть.

Векторная или дистрибутивная модели смысла, как правило, основаны на семантических сетях, где реализованы иерархические связи между понятиями реального мира, однако все эти моменты требуют дополнительной реализации. Матрицы совмещения также применяются для представления связей между терминами. Существует 2 основных подхода к извлечению взаимосвязей: ручную созданные шаблоны и машинное обучение (с «учителем» и без). Первая группа алгоритмов обладает наибольшей точностью, но шаблоны, созданные вручную, в большей степени ориентированы на определенную область знаний. Аналогично, машинное обучение с учителем предполагает ручное аннотирование обучающей выборки, которая впоследствии используется классификаторами для аннотирования остальной части данных. Машинное обучение без учителя, как правило, предполагает использование набора данных для обучения из общего доступа или аналогичного источника, для отбора подобных последовательностей с учетом некоторого вероятностного шаблона [1].

Еще одна группа методов интеллектуального анализа текстовых и графических видеоданных — это нейронные сети, которые предполагают последовательное применение некоторой функции к исходным параметрам для поиска значимого результата. Эта группа методов успешно используется в машинном переводе, диалоговых системах и для генерации речи. Однако, за счет улучшения качества страдает производительность по сравнению с аналогичными системами интеллектуальной обработки [1].

Краткий обзор методов поиска и индексации данных предложен в [2-3]. Здесь упомянуты алгоритмы минимального расстояния редактирования и динамического программирования. Первая группа алгоритмов нацелена на сведение первой анализируемой последовательности до второй с минимальными усилиями, то есть по кратчайшему пути. Различные методы динамического программирования предполагают решение задач путем их разбиения на подзадачи с предложением решений для каждой из них. Эти два подхода в начале 2000-х годов были основными для выравнивания двух

последовательностей, однако они неприменимы для больших объемов мультимедийных данных, которые доступны сегодня, к тому же они неприменимы в разных областях как уникальный подход, поскольку потребуются трансформации для достижения требуемого качества результатов.

Современные подходы к интеллектуальной обработке текста предполагают использование регулярных выражений. Но такая алгебраическая модель может быть реализована исключительно для текстовых данных, тогда как целью настоящей статьи является унификация представления любого типа мультимедийных данных. Еще один подход — это n-граммовая модель прогноза, которая определяет вероятность каждого последующего словосочетания, основываясь на предыдущих. Общая оценка вероятности последовательности выполняется путем перемножения целого ряда вероятностей: вероятности встречаемости префиксов, суффиксов и т.п. Такой подход оказывает неприменимым для больших объемов выборки. Хотя, некоторые такие оценки реализованы в Google для подсчета частоты встречаемости слов [1]. Более общим подходом является расчет энтропии, которая показывает частотную зависимость анализируемой последовательности.

$$E = -\sum_{c=1}^h u_c \log_2 u_c,$$

где h — количество возможных результатов, а u_c — частота встречаемости в выбранной области.

Другой частотный подход — простой классификатор Байеса, который представляет последовательность терминов в виде неупорядоченного множества, так называемого «мешка слов», принимая во внимание частоту встречаемости каждого слова для выполнения классификации. К недостатку такого подхода можно отнести потребность в обучающей выборке, хотя этот подход успешно используется для обнаружения спама, автоматического определения языка и других аналогичных задач.

Интеллектуальный анализ видеоданных считается наиболее сложным типом обработки мультимедийных данных, хотя зачастую он ограничивается распознаванием последовательности изображений с последующей сегментацией и сопоставлением. Аудио составляющая, как правило, опускается или рассматривается отдельно, несмотря на не меньшую ее значимость. Современные методы обработки видеоданных коротко описаны в [4]. На рис. 1-2 показаны круговые диаграммы, где представлены методы интеллектуального анализа текста и видео, которые используются чаще других.

В этой статье внимание сконцентрировано на «чистых» данных (pure data), чтобы предпринять попытку унификации процесса обработки для

разных типов мультимедийного контента. В вышеприведенных примерах нечеткое сопоставление реализовано с помощью линейного поиска и простого перебора. Такой поиск ограничен парами «исходник-перевод», ранее сохраненными и доступными внутри данного переводимого документа (хотя ранее сохраненные сегменты могут быть выполнены другими переводчиками и содержать текст других источников, которые объединены тематикой с текущим переводимым документом). Результаты поиска могут включать «незапрашиваемый» контекст, это как правило и помогает экспертам в области перевода определиться с наилучшим решением. Основная проблема, которая встает здесь, – это релевантность таких результатов запросу конечного пользователя.

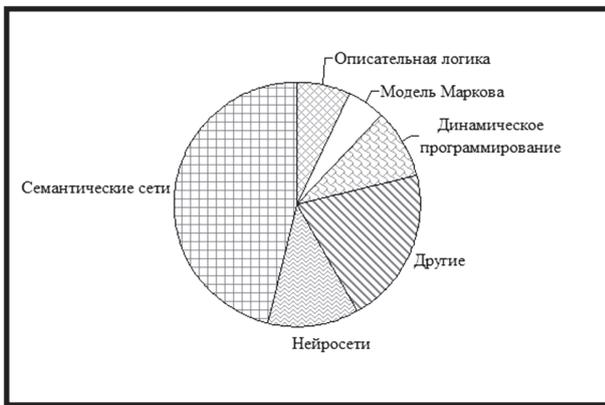


Рис. 1. Популярность методов интеллектуальной обработки текста



Рис. 2. Популярность методов интеллектуальной обработки видео

Очевидно, что семантическая составляющая в обработке текста превалирует над видео на сегодняшний день. В любом случае, цель настоящего исследования состоит в извлечении данных, которые несут значимую информацию. Это означает необходимость преобразования неструктурированной информации из видео или текста в структурированные данные, обрабатываемые впоследствии

или необходимые для принятия корректных решений экспертами. Иными словами, возникает потребность отбора наблюдений из большого объема выборки, возможно с разной интерпретацией одного контента, которые наиболее близки запросу. Также необходимо сравнение векторов признаков для определения их сходства.

1. Основная часть

Вероятностная модель перевода ставит своей целью нахождение наилучшего соответствия целевых сегментов перевода для исходной иностранной последовательности:

$$\text{Перевод} = \text{arg max } P(\text{Перевод} | \text{Исходник}).$$

Рассмотрим для этого следующие положения. Пусть Ω – память переводов (упорядоченная последовательность сегментированных исходных сегментов с их переводом для всего переводимого текста) или видеоряд (упорядоченный набор кадров или сцен). Положим $\rho(x, y)$ – метрика или расстояние между двумя элементами этого множества Ω . Здесь, x и y – минимально возможные элементы памяти переводов или видеоряда. Иными словами, это термины в случае системы на основе памяти переводов или кадры в случае решения задачи распознавания видеоданных. Положим, $\bar{x} = \langle x_1, x_2, \dots, x_n \rangle \in F$ и $\bar{y} = \langle y_1, y_2, \dots, y_m \rangle \in F$, где F – множество всех исходных сегментов перевода или всех сцен видео. Тогда, \bar{x} и \bar{y} – это два сегмента или предложения для сравнения (или две сцены), а $\langle x_1, x_2, \dots, x_n \rangle$ и $\langle y_1, y_2, \dots, y_m \rangle$ – исходные термины/кадры этих предложений/сцен, количество которых равно n и m соответственно. Для любого романо-германского или восточно-славянского предложения и любого кадра сцены важен порядок следования элементов. Изменение порядка следования ведет к некорректной трактовке смысла.

Для сопоставления содержимого, будь то текст или последовательность изображений, можно использовать матрицу расстояний $A(\bar{x}, \bar{y})$ для пары элементов \bar{x} и \bar{y} с разным количеством терминов/кадров в предложении/сцене. Соответственно, и матрица будет содержать s строк и q столбцов. Положим, общее количество терминов/кадров q в первом предложении/сцене \bar{x} меньше или равно общему числу терминов/кадров s во втором предложении/сцене \bar{y} . Таким образом, $s = m - q + 1$, поскольку $m = s + q - 1$, что позволяет сравнить термины/кадры меньшего предложения/сцены с большим:

$$A(\bar{x}, \bar{y}) = \begin{pmatrix} \rho(x_1, y_1) & \dots & \rho(x_q, y_q) \\ \vdots & \ddots & \vdots \\ \rho(x_1, y_s) & \dots & \rho(x_q, y_{s+q-1}) \end{pmatrix}.$$

Такая матрица покажет соответствие друг другу двух предложений или сцен с разным количеством элементов. Если в матрица будет содержать нулевую строку, это означает, что меньшее предложение/сцена входит в состав большего набора элементов. Когда два предложения/сцены полностью идентичны, то есть наблюдается полное 100% соответствие, то все элементы матрицы будут равны нулю. Если общее количество элементов (терминов или кадров) первого и второго предложения/сцены равны, в матрице будет только одна строка [5].

Кроме описанного выше, задание действий в мультипликативном множестве можно выполнить разными способами. Интуитивно понятным является представление с помощью простого перечисления результатов действий для всех пар элементов множества. Те же данные «исходник-перевод» или двух сравниваемых сцен видео можно представить в виде таблицы Кели. В этой таблице на пересечении строки, соответствующей вектору \bar{x} , и столбца, соответствующего вектору \bar{y} , стоит такой элемент $\rho(\bar{x}, \bar{y})$, который показывает сходство между каждой парой сравниваемых компонентов, что по сути и реализовано в современном инструментари на основе памяти переводов в виде реляционной таблицы базы данных.

Вышеприведенная схема сопоставления сегментов не дает представления об их близости в числовой форме. Для этой цели может применяться модель в векторном пространстве, которая предполагает представление сегмента текста в виде частоты встречаемости термина во всем документе. Такая модель известна под названием «модель мешка слов» (bag of words) [6]. Рассмотрим те же два вектора \bar{x} и \bar{y} , тогда их близость можно определить с помощью следующей формулы:

$$\rho(\bar{x}, \bar{y}) = \frac{\bar{x} \times \bar{y}}{|\bar{x}| \times |\bar{y}|}$$

где $|\bar{x}|$ и $|\bar{y}|$ — нормы каждого вектора.

В 1989 Ж. Салтон предложил частотный подход TFIDF (Term Frequency, Inverse Document Frequency), который определяет вес для более значимых терминов документа. Термин считается более значимым, если он чаще встречается в документе. Однако, если тот же термин часто наблюдается в других документах, он считается менее значимым, поскольку не характеризует документ в полной мере. Вес w_{ij} термина t_j в документе d_i можно определить в виде следующего равенства:

$$w_{ij} = tf_{ij} \times \log_2 \frac{N}{n}$$

где tf_{ij} частота термина t_j в документе d_i , а N — количество всех исследуемых документов, n — количество документов, где термин t_j присутствует хотя бы один раз [7].

Добавление нового термина в набор «исходный текст — перевод» можно представить в виде нижеприведенного свойства изоморфных множеств. Понятие изоморфных мультипликативных множеств в теории полугрупп введено для двух мультипликативных множеств (исходный текст и его перевод) с одинаковым количеством элементов, в которых действия определены одинаковым образом и они отличаются незначительно, между элементами этих множеств можно поставить взаимно однозначное соответствие. С введением нового элемента, для изоморфных множеств характерно следующее свойство: $XY = Z \Rightarrow X'Y' = Z'$.

Чтобы произвольное мультипликативное множество было изоморфно какому-либо другому мультипликативному множеству, нужно задать необходимое и достаточное условие такого соответствия. Такой процесс называется нахождением абстрактной характеристики множества [8], то есть поиск корректного перевода — это, по существу, нахождение абстрактной характеристики множества, правила сопоставления элементов.

Теория мультипликативных множеств также включает в себя и другие важные свойства, которые характерны для обработки мультимедийной информации: ассоциативность $(XY)Z = X(YZ)$, коммутативность $XY = YX$, обратимость справа $XZ = Y$ и слева $ZX = Y$, сокращение справа $XZ = YZ \Rightarrow X = Y$ и слева $ZX = ZY \Rightarrow X = Y$ или свойство однозначности деления, как его еще называют. Следует отметить, что в том случае, когда действия мультипликативного множества не являются коммутативными, для отдельных пар элементов множества перестановка все-таки может быть справедливой. Введем определение полугруппы с точки зрения обработки текста. Полугруппой называется непустое множество, в котором для любой пары взятых в определенном порядке элементов XY определен новый элемент, называемый их произведением, причем для любых трех элементов множества X, Y, Z всегда справедливо свойство ассоциативности $(XY)Z = X(YZ)$ [8-9]. Это свойство в особенности характерно для сегментации данных и поиска совпадений.

Для дальнейших исследований свойств полугрупп и операций над мультимедийными данными, как следствие, потребуется использовать понятие и некоторые свойства отношений. Отношением в множестве называется любое бинарное правило, которое отбирает пары элементов из этого множества с указанием порядка их следования, то есть по сути определяется набор «исходный текст — перевод», кот который выбирается из памяти переводов. Обозначение $X \approx Y(\eta)$ определяет отношение η соответствия переводимой фразы для X и

Y , тогда как обозначение $X \leq Y(\eta)$ предполагает включение данных X в последовательность Y .

Важными с точки зрения поиска соответствия перевода являются определения верхней и нижней границ, полуструктуры (решетки) и полной структуры множества [8-9]. Положим, для заданного отношения частичной упорядоченности в множестве Ω с учетом $\Omega' \in \Omega$ и $X \in \Omega$ имеет место $Z \leq X$ при $Z \in \Omega'$. Тогда X является верхней границей Ω' . По аналогии можно определить и нижнюю границу. Таким образом, верхнюю и нижнюю границы можно использовать в качестве терминальных допустимых переводов при необходимости их отбора из памяти переводов.

Однако, стоит отметить, что таких верхних и нижних границ может быть несколько, в отличие от единственной точной верхней и точной нижней границы, которые определяются следующим образом. Верхняя граница X множества Ω' называется точной верхней границей, если X при этом является нижней границей множества Ψ всех верхних границ Ω' . Множество Ω называется полуструктурой (semilattice) или полурешеткой (demi-treillis), если для любой пары элементов существует точная нижняя граница. Множество Ω называется структурой, если для любой пары элементов существует точная верхняя и точная нижняя границы. Ω называется полной структурой, если для любого его подмножества существует точная верхняя и точная нижняя границы [8]. Благодаря точным и неточным границам можно обеспечить извлечение данных с нечетким соответствием поисковому запросу.

2. Экспериментальная часть

Разработка любого нового метода предполагает оценку качества его работы по сравнению с существующими. Для расчета качества полученных результатов разработан целый ряд эвристических методов, таких как BLEU, NIST, TER, METEOR, AMBER, MP4IBM1, LEPOR, полнота и точность. Кроме полноты и точности, с помощью которых можно оценивать результаты обработки видеоданных, остальные метрики предназначены для анализа качества перевода двуязычных документов, хотя во всех этих алгоритмах за основу, так или иначе, взяты меры полноты и точности [10].

Преимущество использования двух показателей совместно заключается в том, что во многих случаях один из них оказывается важнее другого. Одни пользователи не любят просматривать ложно позитивные данные и хотят увидеть пусть меньше результатов, но чтобы все они были релевантными (высокая точность). В противоположность этому, другие пользователи, заинтересованы в том, чтобы полнота поиска была максимально возможной, и снисходительно относятся к низкой точности. Точность и полнота поиска противоречат друг

другу: полноту всегда можно повысить до единицы (при очень низкой точности), возвращая все варианты. Полнота не убывает при увеличении количества результатов поиска. С другой стороны, качественный метод поиска предполагает, что при увеличении числа отобранных данных, как правило, снижается точность.

В целом, задача стоит в достижении определенной полноты поиска при удовлетворительном уровне ложно позитивных результатов. Показатель, позволяющий найти баланс между точностью и полнотой поиска, называется F-мерой [11]. Если точности и полноте назначить одинаковые веса, получится сбалансированная F-мера, которая может быть использована для оценки качества предложенного метода.

Метрикой близости транскрипции в области автоматического распознавания речи является отношение ошибочно распознанных слов (Word Error Rate), которое рассчитывается как минимальное расстояние редактирования. Метрики WER и PER (Player Efficiency Rating) могут использоваться как для оценки качества обработки устной речи, так и для напечатанного двуязычного текста, однако для последнего потребуются некоторые модификации этих метрик, поскольку для перевода существует множество корректных вариантов передачи иностранного текста. По этой причине большая часть метрик для оценки двуязычных текстов сравнивает варианты перевода нескольких человек [1].

На рис. 3 показан пример многовариантного перевода и его представления в системе на основе памяти переводов Trados, а на рис. 4 увеличенные области поиска соответствий и распознавания терминов, что доказывает вышеприведенные положения. Из рисунков видно, что существующая система не предлагает подстановку из терминологического словаря (это остается на усмотрение эксперта-переводчика), несмотря на распознавание терминов. Подстановка в данном случае выполнена из памяти переводов, содержащей 2 варианта. Предложенный метод извлечения данных с нечетким соответствием призван устранить этот недостаток благодаря процедурам сегментации и оценки терминологической близости.

Выводы

Несмотря на то, что речевые и текстовые языковые процессоры сталкиваются с проблемами структурной и координационной двойственности, современные интеллектуальные системы обладают неоспоримыми преимуществами перед методами предыдущего поколения. Достаточно взглянуть на простые приложения реального времени под управлением операционных систем iOS и Android, разработанные и выпущенные в конце 2018 года, такие как приложения по распознаванию

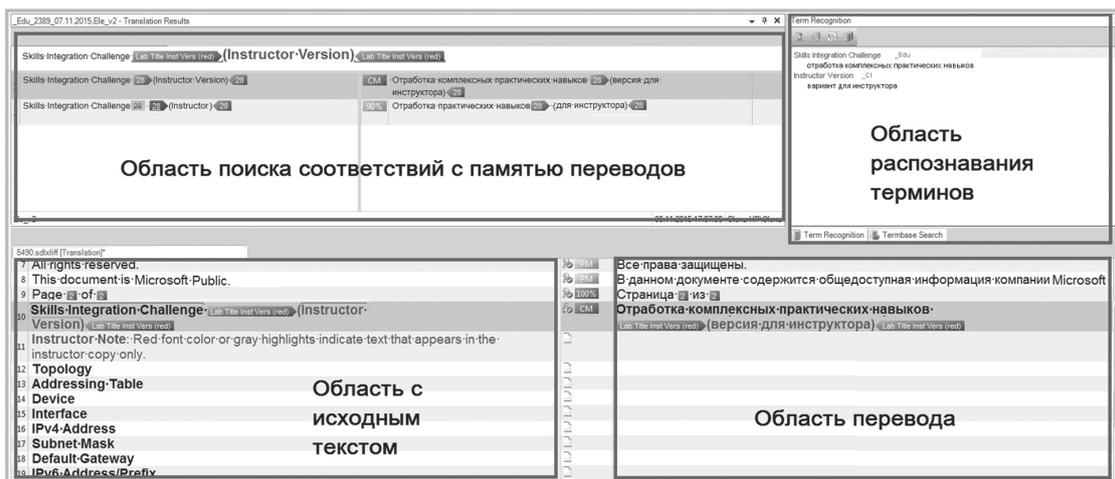


Рис. 3. Пример многовариантного перевода и его представления в системе на основе памяти переводов Trados

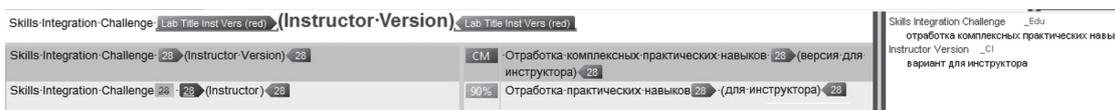


Рис. 4. Увеличенные области поиска соответствий и распознавания терминов в системе Trados

биометрических данных, программа перевода устной речи Speak and Translate Павла Донова, и качественно новый уровень автоматизации станет очевидным. Приложения на базе описательной логики, реализованные с помощью методов Semantic Web, таких как Web Ontology Language (OWL) и RDF-триплеты (Resource Description Framework), также обрели огромную популярность в последнее время. Хотя формальное описание и развертывание семантических онтологий для различных областей знаний является достаточно громоздким.

В этой статье предпринята попытка разработки унифицированной модели распознавания текста и изображений, использование которой обобщает процедуру обработки мультимедийной информации большого объема. Следует также отметить, что в статье не затрагиваются вопросы кореферентности и связи сущностей, не предусмотрено извлечение событий и связей между ними, не рассматривается распознавание поименованных сущностей, нет семантики, только четкие однозначные данные. Предложенные математические формулировки на основе теории полугрупп позволяют одинаково корректно представить все типы доступного мультимедийного контента. Ценность исследований заключается по-большей части в теоретических основах, однако практическая реализация также очень актуальна для сопоставления контента в поисковых приложениях.

Согласно [1], концепции интеллектуального анализа текста открывают еще большие возможности для секвенирования нового поколения в биомедицине и биоинформатике. Примеры задач распознавания в биомедицине включают, но не ограничиваются анализом генома, нуклеотидной последовательности ДНК и РНК, мутаций и

взаимодействия протеинов. Важно отметить, что единое представление биомедицинских, текстовых, звуковых и графических данных позволяет создать высокоточные системы с большим числом параметров.

Список литературы:

- [1] Jurafsky D., Martin J.H. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. – 2-nd edition. – New Jersey: Prentice Hall, 2008. – 1032 p.
- [2] Navarro G. A guided tour to approximate string matching. In: ACM Computing Surveys. – 2001 – Vol. 33, No. 1. – pp. 31-88.
- [3] Navarro G., Baeza-Yates R., Sutinen E., Tarhio J. Indexing Methods for Approximate String Matching. In: IEEE Data Engineering Bulletin. – 2001 – Vol. 24, No. 4. – pp. 19–27.
- [4] Mashtalir S., Mikhnova O. Detecting Significant Changes in Image Sequences. In: Multimedia Forensics and Security. – 2017 – Vol. 115. – pp. 161-191.
- [5] Mashtalir S., Mikhnova O., Stolbovy M. Sequence matching for content-based video retrieval. In: Proceedings of the 2018 IEEE Second International Conference on Data Stream Mining & Processing. – 2018 – pp. 549-553.
- [6] Nahm U.Y., Mooney R.J. Mining Soft-Matching Rules from Textual Data. In: Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence. – 2001 – pp. 979-984.
- [7] Salton G. Automatic Text Processing: The Transformation, Analysis and Retrieval of Information by Computer. – Boston: Addison-Wesley, 1989. – 530 p.
- [8] Ляпин Е.С. Полугруппы. – М.: Государственное издательство физико-математической литературы, 1960. – 592 с.
- [9] Акимов О.Е. Дискретная математика: логика, группы, графы. – М.: Лаборатория Базовых Знаний, 2001. – 352 с.
- [10] Miller G.A., Beebe-Center J.G., Some psychological methods for evaluating the quality of translations. In: Mechanical Translation. – 1956 – Vol.3, No. 3. – pp. 73-80.
- [11] Маннинг К.Д., Рагхаван П., Шютце Х. Введение в информационный поиск. – М.: Вильямс, 2011. – 528 с.

Поступила в редколлегию 15.10.2018

УДК 004.003



В.М. Леви́кін, І.О.Юр'єв

ХНУРЕ, м. Харків, Україна, viktor.levykin@nure.ua

ХНУРЕ, м. Харків, Україна, ivan.iuriev@nure.ua

РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ УПРАВЛІННЯ СИСТЕМОЮ НАДАННЯ ІТ-СЕРВІСІВ

Розглянуто проблему управління системою надання ІТ-сервісів. Проведено аналіз існуючих програмних засобів та інформаційних технологій що дозволяють управляти ІТ-сервісами в умовах зміни функціональних вимог. Запропоновано інформаційну технологію, яка представляє собою послідовність робіт об'єднаних в 4 основні стадії: формування запиту на зміну функціональних вимог, пошук ІТ-сервісів в каталозі, вибір найкращого ІТ-сервісу під функціональні вимоги, оцінка впливу обраного ІТ-сервісу на інфраструктуру підприємства. Розроблена програмна реалізація запропонованої інформаційної технології. Представлені результати застосування програмної реалізації інформаційної технології при реалізації запиту на зміну функціональних вимог. Результати її застосування дозволили скоротити витрати робочого часу персоналу підприємства при проведенні робіт з реалізації запитів на зміну функціональних вимог.

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ, СИСТЕМА НАДАННЯ ІТ-СЕРВІСІВ, ЗАПИТ НА ЗМІНУ, ФУНКЦІОНАЛЬНІ ВИМОГИ

Левыкин В.М., Юрьев И.А. Разработка информационной технологии управления системой предоставления ИТ-сервисов. Рассмотрено проблему управления системой предоставления ИТ-сервисов. Проведен анализ существующих программных средств и информационных технологий позволяющих управлять ИТ-сервисами в условиях изменяющихся функциональных требований. Предложена информационная технология которая представляет собой последовательность работ объединяемых в 4 основные стадии: формирование запроса на изменение функциональных требований, поиск ИТ-сервисов в каталоге, выбора наилучшего ИТ-сервиса под функциональные требования, оценка влияния выбранного ИТ-сервиса на инфраструктуру предприятия. Разработана программная реализация данной информационной технологии. Представлены результаты применения программной реализации информационной технологии при реализации запроса на изменение функциональных требований. Результаты ее применения позволили сократить затраты рабочего времени персонала предприятия при проведении работ по реализации запросов на изменение функциональных требований.

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ, СИСТЕМА ПРЕДОСТАВЛЕННЯ ІТ-СЕРВІСІВ, ЗАПРОС НА ІЗМЕНЕННЯ, ФУНКЦІОНАЛЬНІ ТРЕБОВАВАННЯ

Levykin V.M., Iuriev I.O. Development of information technology management system for the provision of IT services. Considered the problem of managing the system of providing IT services. The analysis of existing software and information technologies allowing to manage IT services in the face of changing functional requirements. An information technology has been proposed which is a sequence of works combined into 4 main stages: forming a request for changing functional requirements, searching for IT services in a catalog, choosing the best IT service for functional requirements, evaluating the impact of a selected IT service on an enterprise infrastructure. A software implementation of this information technology has been developed. Presents the results of the application of software implementation of information technology in the implementation of the request to change the functional requirements. The results of its application have allowed to reduce the time spent by the personnel of the enterprise during the implementation of requests for changes in functional requirements.

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ, ІТ СЕРВІС ПРОВІДЕННЯ СИСТЕМИ, ЗМІНА ЗАПИТУ, ФУНКЦІОНАЛЬНІ ТРЕБОВАВАННЯ

Вступ

Одним з передових підходів до створення і експлуатації інформаційних систем є використання сервіс-орієнтованої архітектури (SOA). SOA — це концепція побудови архітектури інформаційної системи з слабо пов'язаних між собою частин на основі сервісів — окремих компонентів з фіксованими інтерфейсами, що виконують певні функції. Відповідно до сучасними дослідженнями в області експлуатації подібних систем [1-3], первинним структурним елементом SOA інформаційних систем є сервіс, а не підсистема, функція або компонент.

SOA пропонує єдину схему взаємодії сервісів незалежно від того, чи є сервіс частиною однієї

програми, перебуває в іншому адресному просторі багатопроекторної системи, на іншій апаратній платформі і т.д. Все це забезпечує гнучкість SOA, здатність систем, реалізованих в такій архітектурі, реагувати на зміни в бізнес-процесах динамічно і без складних трансформацій на інтеграційному рівні.

Розробка і впровадження на підприємствах нових сервісів обумовлено появою нових бізнес-процесів і розширенням кола вирішуваних завдань. При цьому не останню роль відіграє зворотний зв'язок, яка склалася між бізнесом, пред'являє свої вимоги до складу сервісів, і сервісами, які покликані ці вимоги реалізувати.

Головний вигравш від застосування SOA досягається за рахунок багаторазового використання сервісів. Навіть якщо на автоматизацію одного процесу доведеться затратити більше часу і коштів, мінімізації витрат можна домогтися на більш тривалій період, коли при автоматизації наступних бізнес процесів будуть повторно використовуватися вже розроблені сервіси. Крім того, SOA спрощує інтеграцію нових додатків в існуючу інформаційну систему. В рамках SOA нові програмні продукти повинні легко інтегруватися в існуючу інформаційну систему через механізм сервісів.

В результаті перехід від монолітних архітектур до архітектур, заснованих на сервісах, виник ряд проблем з управління такими системами. Наприклад, слабка зв'язаність призводить до руйнування уявлення про роботу системи, яке існувало раніше, при використанні монолітних систем. Звичайні засоби адміністрування SOA інформаційних систем в більшу міру зосереджені на досягненні найвищої продуктивності апаратного і програмного забезпечення.

Основною перевагою SOA є те, що галузева специфіка застосування сервісів практично відсутня. Залежно від індустрії використовуються різні способи організації роботи і компонування сервісів [4-6]. Наприклад, промислове підприємство може мати не більше десяти виробничих майданчиків, на яких необхідно здійснювати їх підтримку, а у банків і торгових мереж, кількість точок, майданчиків, які потребують обслуговування, обчислюється сотнями, в результаті і способи експлуатації ІС в обох випадках різні.

Таким чином, можна зробити висновок про те, що впровадження на малих підприємствах повномасштабних сервіс-орієнтованих інформаційних систем нераціонально. Реальний ефект можна отримати для середніх і великих підприємств при великій кількості сервісів і при можливості використання різних комбінацій сервісів.

1. Постановка задачі

Метою статті є розробка інформаційної технології управління системою надання ІТ-сервісів, а також її програмна реалізація.

2. Аналіз існуючих підходів до управління ІТ-сервісами

Існуючі підходи до управління сервісами в основному представлені у вигляді «кращих практик». «Кращі практики» і методології управління сервісами, як правило, розроблені великими компаніями і засновані на їх практичному досвіді. Найбільш відомими представниками «кращих практик» є методології управління сервісами: ITIL, MOF, HP, ITGI [7-9].

Дані методології та стандарти були використані при розробці різних ІТ і програмних продуктів з управління сервісами реалізують бізнес потреби підприємства. Незважаючи на те, що більшість з них адаптовані під особливості конкретних галузей і функціональних завдань, всі вони спочатку базуються на існуючих базових підходах до управління.

ITILv3 надає собою опис найбільш важливих видів діяльності в роботі ІТ-служби, а також перелік сфер відповідальності і завдань, які можуть бути адаптовані і застосовані в більшості компаній. На базі ITIL провідні компанії розробили свої структуровані підходи до управління сервісами [10,11]. Серед них HP ITSM Reference Model компанії Hewlett-Packard, IT Process Model компанії IBM, MOF компанії Microsoft і багато інших. Це стало однією з причин, за якими ITIL фактично став стандартом опису фундаментальних процесів управління сервісами.

Варто відзначити, що, як і більшість «кращих практик», в ITIL тільки задекларовано безліч процесів без докладного опису їх практичної реалізації. Так в рамках процесу експлуатації послуг передбачається реалізація наступних завдань:

- ініціалізація і передача на розгляд запитів на зміни (Request for Change, RFC), які дозволять вирішити проблеми, що виникають в процесі експлуатації;
- участь ІТ-служби у зустрічах з керівництвом для обговорення позиції, ризиків і проблем виникають при експлуатації сервісів;
- участь в реалізації змін відповідно до приписів розділу управління змінами;
- здійснення «відкатів» в рамках процесів управління змінами (в разі невдалих змін);
- забезпечення підтримки у визначенні та управлінні моделями змін, які мають відношення до компонентів і послуг етапу експлуатації;
- формування розкладів змін і підготовка до них персоналу.

На етапі експлуатації для підтримки цілісності і повноти інформації про сервіси і збоїв в їхній роботі ІТ-службами підприємств активно використовуються системи управління наданням ІТ-сервісів (ITSM). ITSM-система - це набір інструментів та баз даних, які використовуються постачальником послуг для управління конфігураціями SOA інформаційних систем. У базах даних ITSM міститься інформація про інциденти, проблеми, відомих помилках, зміни і релізах і можуть міститися дані про співробітників, постачальників, місцях розташування, бізнес-одинацях, замовників і користувачів. ITSM включає в себе інструменти для збору, зберігання, управління, оновлення та подання інформації про всі конфігураційних одинацях і

їх взаєминах. ITSM знаходиться під управлінням процесу управління конфігураціями і використовується всіма процесами управління послугами.

Проведений аналіз структури і застосування ІТІЛ показав, що вона є найбільш перспективною базою для побудови ефективної системи надання послуг. ІТІЛ взята за основу багатьма найбільшими компаніями при розробці моделей, технологій і ПО для технічної підтримки процесів управління сервісами.

Основним недоліком ІТІЛ є описовий характер процесів управління сервісами. Більшість необхідних на практиці завдань управління лише задекларовані в ІТІЛ, але реалізація цих завдань покладається на користувача.

3. Розробка інформаційної технології управління системою надання ІТ-сервісів

Основним завданням системи надання ІТ-сервісів є забезпечення доступності ІТ-сервісів для кінцевих користувачів і підтримки їх функцій. Співробітникам підприємства необхідно ефективно взаємодіяти між собою, а також з існуючими та потенційними замовниками, своєчасно обробляти безліч даних. Для цього необхідна наявність відповідної інфраструктури підприємства.

На основі розроблених моделей і методів [12-14] пропонується схема інформаційної технології (ІТ) управління наданням ІТ-сервісів підприємства. Інформаційна технологія включає чотири стадії:

Стадія 1. Формування запиту на зміну функціональних вимог;

Стадія 2. Пошук ІТ-сервісів в каталозі;

Стадія 3. Оцінка і вибір ІТ-сервісів відповідно до функціональних вимог;

Стадія 4. Оцінка впливу нового ІТ-сервісу на інфраструктуру підприємства.

Розроблена ІТ призначена для автоматизації процесу встановлення відповідності між запитами на зміну функціональних вимог і ІТ-сервісами інформаційної системи. Це необхідно керівництву підприємства для того, щоб оцінити наслідки від використання того чи іншого ІТ-сервісу і прийняти відповідне управлінське рішення.

Основними користувачами ІТ управління системою надання ІТ-сервісів є співробітники ІТ-служби підприємства, в обов'язки яких входить безпосередньо обробка заявок користувачів, формування запитів на зміну, управління ІТ-сервісами, а також оцінки впливу нового ІТ-сервісу на інфраструктуру підприємства. Співробітники ІТ-служби передають результати роботи інформаційної технології керівництву, яке безпосередньо приймає рішення про використання ІТ-сервісу. Представлена ІТ може бути використана на

середніх і великих підприємствах у зв'язку з наявністю у таких підприємств великої кількості різних ІТ-сервісів і достатнього потоку запитів на зміну функціональних вимог.

Для реалізації даної інформаційної технології був використаний web-орієнтований підхід. Підставою для використання даного підходу є, в першу чергу, велика кількість задіяних співробітників. Користувачі залишають заявки з новими вимогами, експерти оцінюють інфраструктуру підприємства і різні ІТ-сервіси, співробітники ІТ-служби формують запити на зміну, отримують інформацію з різних відділів підприємства і т.д. В результаті такої розподіленої структури і великої кількості співробітників, задіяних в роботі інформаційної технології, був обраний web-орієнтований підхід до розробки web-додатки для реалізації даної технології. Використання такого підходу дозволить застосовувати існуючі бази даних експлуатованих на підприємстві інформаційних систем при оцінці інфраструктури підприємства, отримуючи до них віддалений доступ через Інтернет. Доступ до web-інтерфейсу надається адміністратором інформаційної системи підприємства співробітнику шляхом видачі відповідного логіна і пароля. Оскільки дана ІТ реалізована у вигляді web-додатки, вона безпосередньо може бути зареєстрована в каталозі ІТ-сервісів, і основні параметри і показники її функціонування можуть бути узгоджені між керівництвом підприємства та ІТ-службою у вигляді SLA.

Виходячи з того, що процес встановлення відповідності між запитом на зміну і ІТ-сервісами виникає, як правило, при розширенні підприємства або його реорганізації, прийняття зваженого управлінського рішення щодо використання або придбання ІТ-сервісу вимагає досить великого часу. У зв'язку з тим, що сам процес пошуку ІТ-сервісу в каталозі займає від кількох хвилин до кількох секунд (в залежності від кількості ІТ-сервісів), загальний час, який співробітник витрачає на обробку RFC і пошук ІТ-сервісів в системі, обмежується декількома годинами робочого часу.

Початковий процес оцінки інфраструктури підприємства є досить трудомістким і вимагає збору великої кількості інформації по різних відділах підприємства, в зв'язку з цим тимчасові витрати на виконання даного етапу ІТ залежать від розмірів підприємства і часу, необхідного для підготовки відповідної інформації.

Виконання першої стадії інформаційної технології полягає в формуванні запиту на зміну функціональних вимог. На основі функціональних вимог викладених у запиті отриманих з різних джерел співробітник ІТ-служби формує новий запит на

зміну. Для цього на підставі запропонованої категорно-функторні моделі визначення відповідності ІТ-сервісів запитам на зміну він виділяє з набору різних текстових описів вимог безліч функціональних вимог. Дані функціональні вимоги групуються в рамках RFC і реєструються в системі шляхом заповнення відповідної форми.

Друга стадія ІТ полягає в підборі ІТ-сервісів під функціональні завдання, визначені відповідно до запиту на зміну. Дане завдання вирішується шляхом пошуку ІТ-сервісів, що реалізують необхідні функціональні завдання. Підготовчим етапом для цього є формування каталогу ІТ-сервісів.

Для формування каталогу ІТ-сервісів необхідно одноразово зібрати на експлуатується об'єкті всю інформацію про використовувані ІТ-сервісах і, використовуючи категорно-функторні модель визначення відповідності ІТ-сервісів запитам на зміну, на підставі описів даних ІТ-сервісів сформувати безліч функціональних завдань, для яких дані сервіси призначені.

Після того, як каталог ІТ-сервісів сформований і функціонує, з'являється можливість виконати підбір ІТ-сервісів під конкретний запит на зміну.

Для цього необхідно програмно виконати пошук ІТ-сервісів під конкретну функціональну задачу. Використовуючи модель управління системою надання ІТ-сервісів, в залежності від результатів пошуку, приймається рішення про можливість використання існуючих ІТ-сервісів для вирішення конкретних завдань:

– в разі, якщо в системі було знайдено кілька ІТ-сервісів під функціональне завдання, необхідно перейти до третьої стадії ІТ для вибору одного з ІТ-сервісів;

– в разі, якщо в результаті пошуку не було знайдено жодного ІТ-сервісу, співробітнику ІТ-служби необхідно проаналізувати існуючі на ринку ІТ-сервіси та сформувати з них безліч альтернативних ІТ-сервісів і перейти на стадію чотири;

– в разі, якщо в системі знайдений єдиний ІТ-сервіс під дані функціональні завдання, він повинен бути використаний для закриття даного запиту на зміну функціональних вимог.

Для вирішення завдання встановлення відповідності отриманих запитів на зміну і ІТ-сервісів була розроблена схема алгоритму пошуку ІТ-сервісів, представлена на рис. 1.

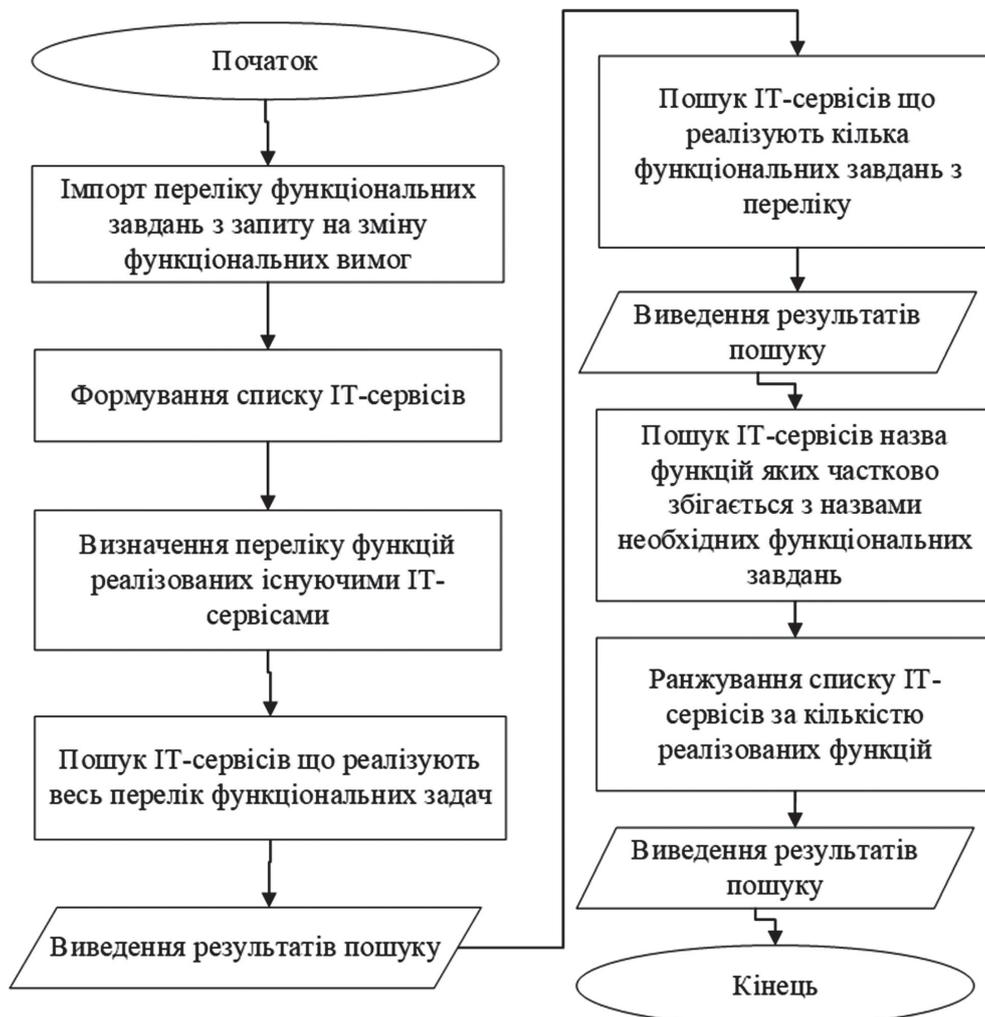


Рис. 1. Алгоритм пошуку ІТ-сервісів в системі

Третя стадія ІТ використовується в ситуації, коли необхідно зробити вибір одного з декількох ІТ-сервісів. Для цього, використовуючи метод вибору ІТ-сервісів. З безлічі ІТ-сервісів, сформованих на другій стадії, і отриманих експертних оцінок програмно вибирається ІТ-сервіс, найбільш задовольняє висунутим критеріям оцінки альтернатив. Після того, як в рамках системи надання ІТ-сервісів був визначений ІТ-сервіс, що задовольняє всім вимогам запиту на зміну, необхідно оцінити його вплив на існуючу інфраструктуру підприємства.

В рамках четвертої стадії ІТ виконується оцінки стану експлуатованої інфраструктури підприємства на основі витрат і експертних оцінок. Системою формується кількісна оцінка інфраструктури підприємства. Для цього на підставі моделі оцінки інфраструктури підприємств необхідно зібрати інформацію про витрати підприємства на підтримку існуючої інфраструктури підприємства до впровадження нового ІТ-сервісу.

Далі, відповідно до методу оцінювання інфраструктури підприємства, необхідно повторно визначити витрати на зміну інфраструктури підприємства з урахуванням витрат на впровадження нового ІТ-сервісу. Дані про витрати до і після впровадження нового ІТ-сервісу, з урахуванням передбачуваного прибутку від впровадження даного ІТ-сервісу, дозволяють розрахувати рентабельність витрат на використання ІТ-сервісу. Дані про рентабельність витрат зберігається в системі і буду використані в подальшому при формуванні рекомендацій щодо доцільності коригуючого впливу (використання даного ІТ-сервісу).

Формування якісної складової оцінки інфраструктури підприємства здійснюється із залученням експертів, які формують оцінки елементів інфраструктури підприємства з урахуванням обраного ІТ-сервісу. Експертні оцінки обробляються з використанням методу оцінки інфраструктури підприємства, таким чином формується загальна оцінка стану інфраструктури даного підприємства. Значення отриманої оцінки зберігається в системі і використовується при формування рекомендацій щодо доцільності коригуючого впливу.

Формування рекомендацій про доцільність коригуючого впливу здійснюється на підставі отриманого раніше значення рентабельності витрат і якісної оцінки інфраструктури підприємства.

При розробці інформаційного забезпечення ІТ були враховані вимоги до складу та повноти інформації про ІТ-сервісах в каталозі, тому що опис ІТ-сервісу є основним елементом, з яким взаємодіють всі стадії ІТ. Формування опису ІТ-сервісу повинно проводитися на основі затвердженого «оголошення про рівень ІТ-сервісу», що дозволяє

заносити в каталог тільки достовірну інформацію про функції та умови надання ІТ-сервісів.

В рамках інформаційного забезпечення окреме місце приділяється питанням зберігання і обробки отриманих експертних оцінок на різних стадіях ІТ, внаслідок чого повинні бути передбачені сховища для експертних оцінок якості інфраструктури підприємства і експертних оцінок ІТ-сервісів.

4. Програмна реалізація інформаційної технології

За реалізацію кожної бізнес-потреби користувача в системі відповідає певний ІТ-сервіс або набір ІТ-сервісів. Коли виникає новий запит на зміну, доцільно спочатку перевірити, чи існує в системі ІТ-сервіс, повністю або частково реалізує необхідні функції. Для цього необхідно виконати пошук ІТ-сервісів в системі відповідно до переліку функцій, зазначених у запиті на зміну. Необхідно відзначити, що програма виконує трьохетапний пошук (рис. 2). На першому етапі виконується пошук на повний збіг переліку функціональних вимог з введенням переліком функціональних можливостей, в результаті заповнюється розділ «Повний збіг з переліком функціональних можливостей». Якщо в результаті пошуку було знайдено хоча б один ІТ-сервіс, подальші етапи не виконуються. На другому етапі програма відбирає ІТ-сервіси, які реалізують хоча б одне функціональне вимога з введеного переліку, в результаті заповнюється розділ «ІТ-сервіси, що реалізують окремі функціональні вимоги». На третьому етапі програма відбирає ІТ-сервіси, найменування функцій яких містить окремі слова або кілька слів з переліку функціональних вимог.

Як видно з «Пошук дав» (рис.2), в каталозі ІТ-сервісів відсутній ІТ-сервіс, повністю реалізує всі необхідні функціональні вимоги. Наприклад, в модулі «Інтернет-магазин» повністю реалізована тільки одна вимога «Авторизації по СМС», а також базові функції стандартного Інтернет-магазину, при цьому функції, пов'язаних з електронним полісом, що не знайдено. Були знайдені також різні модулі, які частково реалізують необхідні функції, але не для електронного поліса, а для різних договорів страхування.

5. Перспективи подальшого дослідження

Перспективним напрямком подальшого дослідження в області управління надання ІТ-сервісів є розробка моделей і методів дозволяють автоматизовано виділяти функціональні завдання з описів надходять в систему ІТ-сервісів. Проведення подібних досліджень обумовлено необхідністю повністю автоматизувати процес встановлення відповідності між існуючим на підприємства набором ІТ-сервісів і надходять в систему запитами на зміну функціональних вимог.

Поиск ИТ-сервиса в соответствии с запросом на изменение

В выпадающем списке выберите интересующий Вас "Запрос" и нажмите кнопку "Загрузить информацию по запросу"

Выберите запрос: № 132 ▾ Электронный полис ▾ Павлов Виктор Сергеевич ▾

Загрузить информацию по запросу

Для выполнения поиска нажмите кнопку "Поиск"

Список функциональных требований к ИТ-сервисам:	Результаты поиска ИТ-сервисов в каталоге
заполнение электронного полиса проверка электронного полиса оформление электронного полиса продажа электронного полиса печать договора по электронному полису проверка срока действия электронного полиса оплата электронного полиса расчет комиссионного вознаграждения по электронному полису авторизация через СМС	Полное совпадение с перечнем функциональных требований
	Не найдено ИТ-сервисов полностью реализующих все функциональные требования
	ИТ-сервисы полностью реализующие отдельные функциональные требования:
	Название ИТ-сервиса: Реализованные в данном ИТ-сервисе функциональные требования:
	Интернет-магазин авторизация через СМС
	ИТ-сервисы, в которых реализованы отдельные части функциональных требований:
	Название ИТ-сервиса: Функции ИТ-сервиса:
	Интернет-магазин оформление полиса, оформление, продажа, оплата, печать, смс, авторизация через смс, заполнение
	Учет договоров проверка, проверка договора, печать, срок, печать договора
	Комиссионные комиссионные, расчет, печать, проверка
Андеррайтинг проверка, проверка полиса, полис	

Рис. 2. Фрагмент екранної форми програмної реалізації інформаційної технології «Результати пошуку ІТ-сервісів»

Висновки

Проведено аналіз існуючих підходів і методологій управління наданням ІТ-сервісів. Визначено їх основні переваги та недоліки. За результатами аналізу використання бібліотеки ІТІЛ є найбільш обґрунтованим для побудови різних систем управління ІТ-сервісами.

Проведено аналіз сервіс-орієнтованих інформаційних систем побудованих на принципах ІТІЛ. Визначено моделі, використання яких є основою для побудови ІТSM-систем.

Запропоновано інформаційну технологію яка представляє собою послідовність робіт об'єднуються в 4 основні стадії: формування запиту на зміну функціональних вимог, пошук ІТ-сервісів в каталозі, вибору найкращого ІТ-сервісу під функціональні вимоги, оцінка впливу обраного ІТ-сервісу на інфраструктуру підприємства.

Розроблена програмна реалізація даної інформаційної технології. Представлені результати застосування програмної реалізації інформаційної

технології при реалізації запиту на зміну функціональних вимог. Результати її застосування дозволили скоротити витрати робочого часу персоналу підприємства при проведенні робіт по реалізації запитів на зміну функціональних вимог.

Список літератури:

- [1] Teubner A., Remfert C. Giving IT Services a Theoretical Backing. Human Interface and the Management of Information: Information, Knowledge and Interaction Design. 2017. Vol. 10273. P. 448-468. doi:10.1007/978-3-319-58521-5_35.
- [2] Micić Z., Blagojević M., Micić M. Innovation and knowledge trends through standardisation of IT applications. Computer Standards & Interfaces. 2014. Vol. 36. No. 2. P. 423-434. doi:10.1016/j.csi.2013.08.013.
- [3] Mayo D., Escalona F., Ross M., Staples G., Staples M. A quality management based on the Quality Model life cycle. Computer Standards & Interfaces. 2013. Vol. 34. P. 396-412. doi:10.1016/j.csi.2012.01.004.
- [4] Maevsky D. A. A New Approach to Software Reliability. Lecture Notes in Computer Science: Software Engineering for Resilient Systems. 2013. № 8166. P. 156-168.

- [5] Mamta Meena, Anamika RamSing, Vinayak Bharadi. Architecture for Software as a Service Model of CBIR on Hybrid Cloud of Microsoft Azure. *Procedia Computer Science*. 2016. Vol. 79. P. 569–578.
- [6] Martini B., Paganelli F A service-oriented approach for dynamic chaining of virtual network functions over multi-provider software-defined networks. *Future Internet*. 2016. Vol. 8. No. 2. P. 1–21.
- [7] Verlaine B., Jureta I., Faulkner S. How Can ITIL and Agile Project Management Coexist? Exploring Services Science. *IESS*. 2016. Vol. 247. P. 327–342. doi:10.1007/978-3-319-32689-4.
- [8] ITIL Service Transition. AXELOS: Second edition, London: «TSO», 2011. 364 p.
- [9] ITIL Continual Service Improvement. AXELOS: Second edition, London: «TSO», 2011. 262 p.
- [10] ITIL. The key to Managing IT services Office of Government Commerce. London: TSO. 2005. 418 c.
- [11] Felinto R., Rodrigues C., Ayres K. The PDCA cycle in practice: the experience of managing subjects of the administration course of the Federal University of Paraiba International. *Journal of Advanced Operations Management (IJAOM)*. 2014. Vol. 6. No. 4. doi: 10.1504/IJAOM.2014.066826
- [12] Левыкин В. М., Юрьев И. А. Разработка моделей инфраструктуры предприятия и инфраструктуры информационной системы, их оценка и установление соответствия между элементами моделей. *АСУ и приборы автоматизи.* 2016. № 174. С. 4–9.
- [13] Левыкин В. М., Юрьев И.А. Разработка метода оценки качества инфраструктуры предприятия и ее компонентов. *Системи та технології*. 2016. № 1 (55). С. 58–62.
- [14] Левикін В. М., Юр'єв І. О. Модель вибору набору ІТ-сервісів для кінцевих користувачів. *Вісник НТУ «ХПІ»*. Серія: Системний аналіз, управління та інформаційні технології. 2016. № 45 (1217). С. 78–84.
- [15] Levykin V., Iuriev I. Development of a model for determining the alignment of it-services of the information system with the end-user requirements
- [16] Eastern–European journal of enterprise technologies. 2017. Issue 4/2 (88). P. 4–9. doi: 10.15587/1729-4061.2017.108556.

Надійшла до редколегії 20.11.2018



Кириченко І.В.

Харківський національний університет радіоелектроніки,
Харків, Україна, iryna.kyrychenko@nure.ua

КОНЦЕПЦІЯ СИСТЕМИ АДАПТИВНОГО ЕЛЕКТРОННОГО НАВЧАННЯ НА ОСНОВІ ІДЕНТИФІКАЦІЇ НАВЧАЛЬНОГО КОНТЕНТУ

Протягом останніх двох десятиріччів, ми спостерігаємо процес переходу від традиційного навчання до навчання з застосуванням сучасних комп'ютерних технологій. Це стало можливим завдяки глобальному розвитку мережі Інтернет, що дало можливість переміщувати необхідну кількість даних з одного ресурсу в інший, вільно вести дискусії з іншими користувачами мережі в online-режимі та розміщувати різноманітну інформацію на Інтернет-ресурсах, що зробило інформаційні дані високоступними для всіх бажаючих. Сучасні інформаційні технології дають змогу підвищити та вдосконалити ефективність освітнього процесу.

Електронне адаптивне навчання передбачає високотехнологічний підхід до процесу передачі знань і дає можливість створення системи масового безперервного самонавчання, загального обміну інформацією. Саме ця система може адекватно і гнучко реагувати на потреби суспільства щодо підготовки високо-професійних фахівців. Крім того, впровадження дистанційного навчання уможливує на практиці ідею пожиттєвої освіти, навчання через усе життя. Система дистанційної освіти підвищує ефективність і якість навчання завдяки додатковим можливостям пізнання навколишньої дійсності і самопізнання, розвитку особистості студента; управління навчально-виховним процесом, проведенню моніторингу (контролю, корекції результатів навчальної діяльності, комп'ютерного педагогічного тестування і психодіагностики; поширення науково-методичного досвіду; організації інтелектуального дозвілля).

ПОШУК ІНФОРМАЦІЇ, ІНФОРМАЦІЙНІ СИСТЕМИ, ОПРАЦЮВАННЯ ДАНИХ, ЕЛЕКТРОННЕ НАВЧАННЯ, ІДЕНТИФІКАЦІЯ

Кириченко И.В. Концепция системы адаптивного электронного обучения на основе идентификации учебного контента. В течение последних двух десятилетий мы наблюдаем процесс перехода от традиционного обучения к обучению с применением современных компьютерных технологий. Это стало возможным благодаря глобальному развитию сети Интернет, что позволило перемещать необходимое количество данных с одного ресурса на другой, свободно вести дискуссии с другими пользователями сети в online-режиме и размещать различную информацию на Интернет-ресурсах, а также сделало информационные данные высокоступными для всех желающих. Современные информационные технологии позволяют повысить и усовершенствовать эффективность образовательного процесса.

Электронное адаптивное обучение предполагает высокотехнологичный подход к процессу передачи знаний и дает возможность создания системы массового непрерывного самообучения, всеобщего обмена информацией. Именно эта система может адекватно и гибко реагировать на потребности общества по подготовке высокопрофессиональных специалистов. Кроме того, внедрение дистанционного обучения осуществляет на практике идею пожизненного образования, обучение через всю жизнь. Система дистанционного образования повышает эффективность и качество обучения, благодаря дополнительным возможностям познания окружающей действительности и самопознания, а также развития личности студента; управления учебно-воспитательным процессом, проведения мониторинга (контроля, коррекции результатов учебной деятельности, компьютерного педагогического тестирования и психодіагностики, распространения научно-методического опыта; организации интеллектуального досуга).

ПОИСК ИНФОРМАЦИИ, ИНФОРМАЦИОННЫЕ СИСТЕМЫ, ОБРАБОТКА ДАННЫХ, ЭЛЕКТРОННОЕ ОБУЧЕНИЕ, ИДЕНТИФИКАЦИЯ

Kyrychenko I.V. Concept of adaptive e-learning system based on content identification. Over the past two decades, we are witnessing the process of transition from traditional teaching to the use of modern computer technology. This became possible due to the global development of the Internet, which enabled to move the necessary amount of data from one resource to another, to freely conduct discussions with other users of the network in online mode and to place various information on the Internet resources, which made the information data highly accessible to all interested persons. Modern information technology enables to improve and improve the efficiency of the educational process.

Electronic adaptive learning involves a high-tech approach to the transfer of knowledge and provides the opportunity to create a system of mass continuous self-education, general information exchange. It is this system that can most adequately and flexibly respond to the needs of society in the training of highly qualified professionals. In addition, the introduction of distance learning enhances the idea of life-long learning, life-long learning. The system of distance education increases the efficiency and quality of learning due to the additional possibilities of knowledge of the surrounding reality and self-knowledge, the development of the student's personality; management of the educational process, monitoring (control, correction of the results of educational activities, computer pedagogical testing and psychodiagnosics, dissemination of scientific and methodological experience, organization of intellectual leisure).

INFORMATION SEARCH, INFORMATION SYSTEMS, DATA PROCESSING, E-LEARNING, IDENTIFICATION

Вступ

Досягнення в області інформаційних та комунікаційних технологій надають багато перспектив, зокрема в сфері освітніх технологій, особливо через наявність мережевих та Інтернет технологій. Такі технології дозволили розвинути різноманітні педагогічні підходи, що суттєво вплинуло на навчальний процес за рахунок створення нових можливостей для обміну та надання знань. Наявність різноманітних постачальників програмного забезпечення для підтримки та супроводу великої кількості ресурсів та сервісів дозволило багатьом університетам та освітнім закладам використовувати їх для управління навчальним процесом, студентами та розподіленням навчальних матеріалів.

Для навчальних систем адаптивність надає таку важливу перевагу як індивідуалізація навчального процесу. В процесі класичного навчання у форматі «людина-людина» досвідчений викладач може виявити сильні та слабкі сторони того, кого навчають, оцінити його загальну ерудицію та рівень знань конкретної дисципліни – і, виходячи з цих даних, визначити обсяг та оптимальну форму подачі матеріалу [1]. Теорія адаптивних систем виникла внаслідок необхідності вирішення великої кількості прикладних задач, для яких неможливо використати традиційні методи, які потребують знання адекватної математичної моделі об'єкту дослідження.

Навчання, засноване на використанні комп'ютерних систем, в силу своєї специфіки спочатку було позбавлено можливості враховувати особисті індивідуальні особливості кожного конкретного учня, являючи собою перекладення паперового підручника в цифровому вигляді з додаванням елементів інтерактивності. Вирішенням цієї проблеми зайняті розробники адаптивних систем навчання. Ідея таких систем полягає у створенні алгоритмів та інструментів накопичення інформації про користувача, з метою використання цієї інформації при визначенні навчальної траєкторії. Ці системи намагаються бути більш пристосованими шляхом побудови моделі цілей, переваг та рівня знань кожного окремого студента, якого навчають, та використання цієї моделі під час взаємодії, для адаптації до його потреб. Крім того, з урахуванням особливостей знань учня за курсом, системи можуть регулювати об'єм навчального матеріалу, що подається, а також форму його представлення: текст, гіпермедіа, зображення, презентація тощо.

Підготовка та розвиток персоналу є важливим у всіх компаніях, незалежно від їх розміру [2]. Підготовлені співробітники забезпечують кращу продуктивність, що призводить до більшого прибутку. Оскільки навчання є важливим для всіх компаній, дуже важливо мати програмне забезпечення,

яке може автоматизувати цей процес. Саме тому багато компаній намагаються створити автоматизований навчальний процес для своїх співробітників.

Системи управління навчанням (LMS або Learning Management System) стали невід'ємною частиною процесу розробки контенту електронного навчання. Система управління навчанням – це програмне забезпечення, яке допомагає керувати електронним навчанням, дозволяючи керувати контентом, доставляти курс та відслідковувати прогрес [3, 4]. Такі системи надають електронні модулі навчання через мережу Інтернет або за допомогою спеціального програмного забезпечення.

Відповідно до статистики [5, 6], завдяки автоматизації системи навчання та тренінгів, збільшуються такі показники, як: задоволеність співробітників роботою зростає на 38%, покращується продуктивність роботи на 42%, покращується виконання робочих зобов'язань на 32.5%. Перевагами використання системи управління навчанням є [5–9]:

1 Контент в одному місці. Система управління навчанням дозволяє зберігати весь зміст у одному місці, зменшуючи ризик втрати важливих даних. Кожен учасник команди керування e-Learning або інші користувачі можуть отримати доступ до інформації, даних або будь-чого іншого через систему хмарного зберігання.

2 Легкий доступ до навчального матеріалу. Завантажений на LMS навчальний матеріал стає постійно доступним для всіх користувачів. Студенти легко можуть отримувати курси з будь-якого місця, своєчасно та з використанням будь-якого мобільного пристрою.

3 Легко відстежувати навчальний процес та розвиток учнів. LMS дає можливість відстежувати прогрес студентів у матеріалах курсу. Точніше, можливо відстежувати їх розуміння матеріалу, вимірювати їх взаємодію з наданим вмістом, з'ясувати, чи відповідає тренінг всім потребам особи, що навчається.

4 Зниження витрат на навчання та підвищення рівню знань. Система управління навчанням забезпечує велику мобільність, оскільки фактична присутність необов'язкова.

5 Зниження тривалості навчання. Особа, що навчається може придбати навички або знання певного змісту запропонованого курсу. У цьому плані LMS дозволяє отримати прямий доступ до курсів. Система тестів може дозволити бути в змозі підтвердити, що певні елементи, розділи або частини можуть бути звільнені від подальшого вивчення.

6 Легко оновити навчальний матеріал. Якщо потрібно додати, видалити або оновити онлайн-модулі для електронного курсу навчання, можна

просто отримати доступ до системи управління навчанням та внести необхідні зміни.

7 Легкість інтеграції соціальних навчальних навичок. Система управління навчанням дозволяє інтегрувати досвід соціального навчання в стратегію електронного навчання, де можна включати посилання на Facebook, Twitter, LinkedIn та онлайн-форуми, які, безумовно, будуть корисні.

1. Постановка задачі

Аналіз науково-методичних основ електронного навчання доводить, що в останні десятиліття швидко розвиваються підходи надання знань дистанційно. Проблемам з питань розвитку електронної освіти присвячені роботи багатьох зарубіжних науковців, серед яких: Р. Деллінг, Г. Рамбле, Д. Кіган, М. Сімонсон, М. Мур, А. Кларк, М. Томпсон та ін., і відповідно вітчизняних, таких як: Н. Шаронова, Н. Клокар, І. Козубовська, А. Волокита, Є. Полат, А. Хуторський. Незважаючи на велику кількість наукових досліджень, сучасна електронна освіта в Україні нагадує традиційні форми заочного навчання, без застосування всіх можливостей принципово нових форм і методів навчання. Сьогодні немає одностайного рішення серед науковців навіть щодо визначення поняття «дистанційне навчання». Також багато праць зарубіжних науковців відводять особливу роль телекомунікаціям в організації дистанційного навчання і визначають його як «теленавчання» [5-13].

Однією з тенденцій сучасної освіти – як загальної, так і вузівської, є можливість побудови персональної освітньої траєкторії. В освітніх стандартах четвертого покоління йдеться, що «навчальний заклад зобов'язаний забезпечити тих, хто навчається, можливістю брати участь в створенні своєї програми навчання, включаючи можливу розробку індивідуальних освітніх програм».

Освітні стандарти сучасного покоління допускають персональний підхід до кожного учня. У програмі навчання повинні міститися необхідні дисципліни, а також дисципліни з можливістю вибору. Таким чином, виникає можливість в підтримці індивідуальних навчальних планів. Так само сучасним напрямком в області систем електронного навчання є адаптивні навчальні системи. Основною відмінністю таких систем є можливість адаптації навчального матеріалу до індивідуальних особливостей того, хто навчається. Завдання адаптивних систем навчання – оптимізація навчального процесу шляхом надання

користувачу навчального матеріалу в найбільш відповідній формі. Результатом такого підходу є безумовне підвищення якості результатів навчального процесу.

Для реалізації адаптації на рівні змісту навчального матеріалу, необхідно виконати значну роботу з формування навчального контенту з кожної дисципліни, з можливістю вибору кращої форми подання інформації. Даний підхід вимагає серйозних витрат інтелектуальної праці та спеціалізованих ресурсів, але в результаті дозволяє отримати абсолютно нову та сучасну, високоефективну систему навчання, яка відповідає вимогам споживача.

2. Результати досліджень

Аналіз показав, що перевагами використання електронних технологій для інтенсифікації навчального процесу є: методична різноманітність форм, методів і прийомів навчання; можливість застосування графічних, аудіовізуальних та анімаційних об'єктів для оформлення навчального матеріалу; забезпечення різних видів зворотного зв'язку для контролю, аналізу та оцінки результатів навчальних досягнень студентів; моделювання професійно значимого контенту навчальної діяльності; формування індивідуальних стратегій. Для систематизації інформації про предметну область було складено онтологію предметної області, а саме виділено основні концепти та відношення між ними. На рис. 1 представлено фрагмент онтології у вигляді UML-діаграми класів. До основних концептів даної предметної області можна віднести:

- 1 Навчальний курс.
- 2 Результати навчання.
- 3 Форма подання контенту.
- 4 Модель навчальних матеріалів.
- 5 Метаопис.

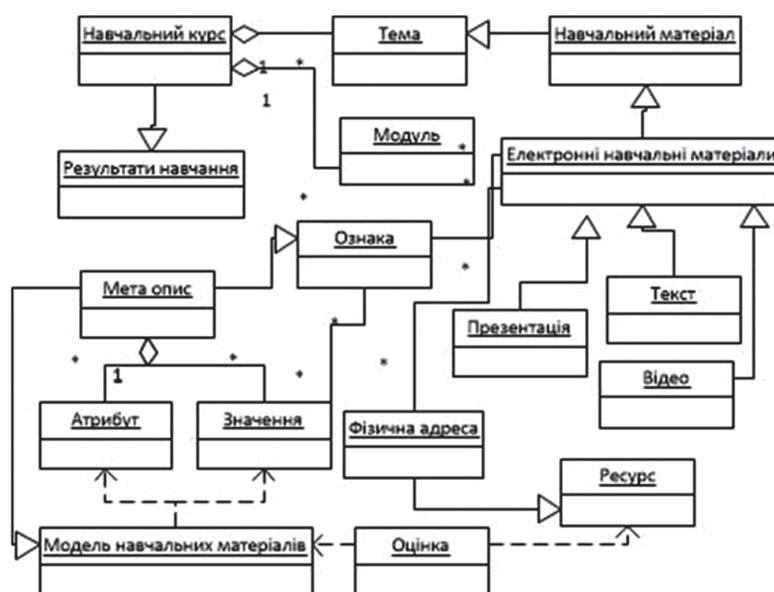


Рис. 1. Фрагмент онтології у вигляді UML-діаграми

Різні науковці досліджують різноманітність методів, спрямованих на застосування адаптивності до стилів навчання студентів. Ці дослідження підтверджують, що надання адаптивного електронного навчання сприятиме покращенню процесу навчання, яке полегшує педагогічна модель, надана кожною методикою адаптації.

У 2000 р. ініціативна група ADL (Advanced Distributed Learning) розробила стандарт SCORM (Sharable Content Object Reference Model), створений для систем дистанційного навчання. Даний стандарт містить ряд вимог до організації навчального контенту і всієї системи навчання. SCORM заснований на стандарті XML і дозволяє підтримувати сумісність компонентів і можливість їх багаторазового використання: навчальний матеріал представлений невеликими розділами, які можуть включатися в навчальні курси і використовуватися системою електронного навчання незалежно від того, ким, де, та за допомогою чого були створені. Структура пакету SCORM приведена на рис. 2.

SCORM — це набір специфікацій і стандартів, які представлені на множині розділів:

- Модель зберігання змісту;
- Середовище поточного виконання;
- Пошук і навігація.

За результатами кількох досліджень були впроваджені модулі для забезпечення адаптивних освітніх гіпермедіа систем. Загальна мета цих досліджень полягала в пошуку рішення проти однієї величини, що відповідає усім, а також для компенсації дефіциту знань, мінімізації втоми та дискомфорту учнів з вивчення теми, що забезпечує більш легке введення в тему. Загальна структура адаптивної системи дистанційного навчання представлена на рис. 3.

Центральне місце на цій схемі займає створення навчального курсу (рис. 4). Модель адаптивних систем навчання заснована на використанні навчальних об'єктів (LO – Learning Objects). Навчальний об'єкт має велику кількість визначень, більшість з яких зосереджуються на загальних принципах, наприклад: багаторазове використання в різних умовах навчання та незалежність контексту [12]. Навчальний об'єкт також визначається як одиниця змісту навчання, незалежна і автономна, яка може повторно використовуватися в декількох навчальних контекстах. Для комітету зі стандартів навчання в галузі технологій навчання IEEE [14] навчальні об'єкти визначаються як будь-який суб'єкт, цифровий або нецифровий, який може бути використаний, повторно використаний



Рис. 2. Структура SCORM-пакету

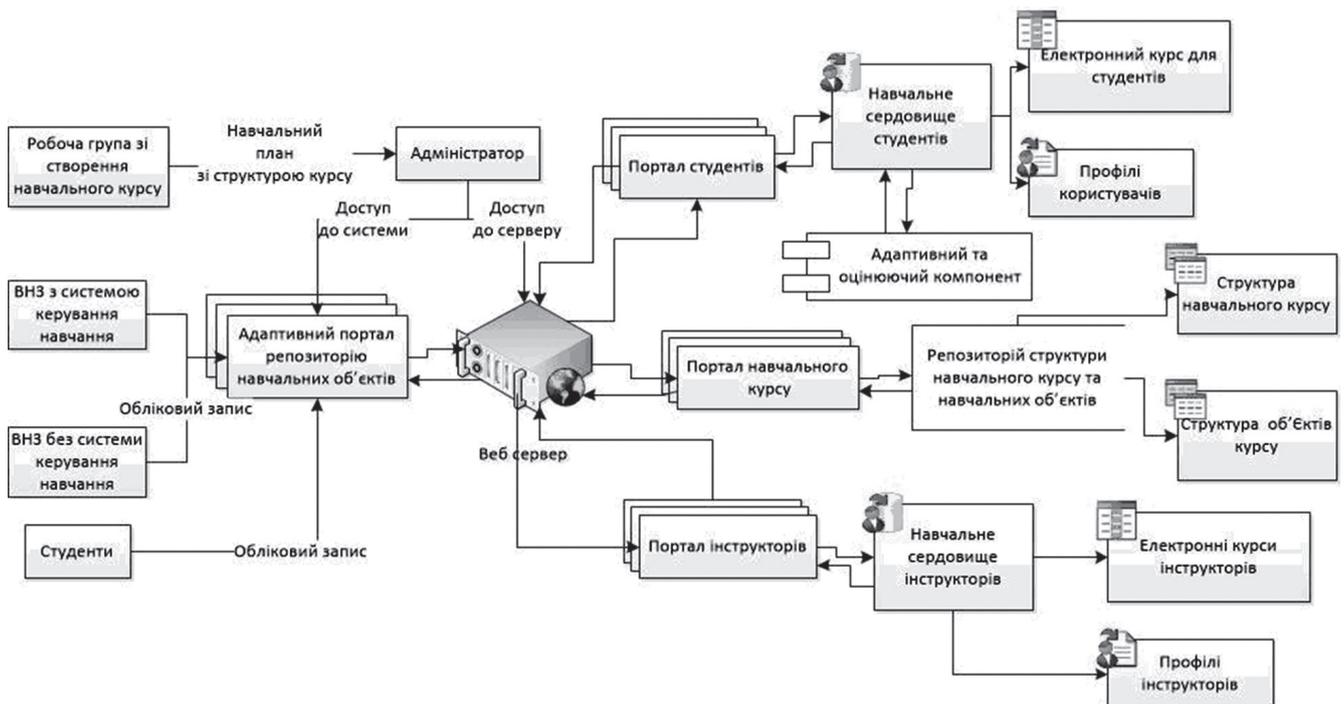


Рис. 3. Структура адаптивної системи електронного навчання

або на який можна посылатися під час навчання, що підтримується технологією.

Модель змісту або контенту описує компоненти, які використовуються для побудови навчального процесу з навчального об'єкта. Модель змісту також визначає, як ці ресурси для вивчення нижчого рівня агреговані та організовані в навчальні підрозділи вищого рівня [10-13]. Використання уніфікованої структури адаптованого електронного навчання надає змогу підвищити стандарти та результати навчання, подолавши перешкоди, що виникають внаслідок відсутності гнучкості, що перешкоджає впровадженню технології електронного навчання. Система може бути використана для забезпечення адаптивного електронного навчання, її структура може бути модифікована з урахуванням інших адаптивних методологій.

Модель контенту на основі специфікації SCORM об'єднує набір пов'язаних технічних стандартів, специфікацій та інструкцій, розроблених для задоволення вимог високого рівня SCORM: доступний, сумісний, тривалий та багаторазовий вміст [12]. Модель SCORM має три основні компоненти: модель агрегації вмісту, середовище виконання, навігація та послідовність.

Модель агрегації вмісту визначає структуру з трьома рівнями агрегації [10]:

- активи – це є основний будівельний блок навчального ресурсу; активи можуть бути текстом, анімацією, зображеннями, відео тощо;

- Object Sharable Content (SCO) – це компіляція одного або декількох активів; SCO являє собою найнижчий рівень деталізації одного ресурсу навчання, який зв'язується з LMS, використовуючи SCORM Run Time Environment;

- Content Organization (CO) – це представлення, яке визначає навмисне використання вмісту через структуровані одиниці навчання; представлення показує, як одна діяльність пов'язана з іншою.

Модель домену (Domain model) характеризується своєю компетентністю щодо представлення концепцій для навчання, ресурсів, доступних учням, та структурування різних елементів поля. Зазвичай модель домену розділена на дві частини [12]:

1 Всі концепти домену, які користувач (учень, студент, співробітник компанії) може отримати в процесі навчання, незалежно від різних видів ресурсів.

2 Навчальні ресурси, які використовуються для реалізації концепцій, разом з їх метаданими. Щоб забезпечити повторне використання навчальних об'єктів, необхідно кваліфікувати їх, використовуючи метадані та необхідні дескриптори, які є важливими для характеристики кожного навчального об'єкту.

Модель ідентифікації навчальних об'єктів спирається на дослідження наукової школи проф. Ю. П. Шабанова-Кушнарєнка та його послідовників в області теорії інтелекту, із застосуванням методу компараторної ідентифікації для моделювання інтелектуальних функцій людини. У роботі приймається, що алгебра скінченних предикатів, як основний інструментарій теорії інтелекту, має універсальний характер для моделювання інтелектуальних процесів обробки даних.

Модель адаптації стосується створення адаптивного контенту, який надалі буде представлений учням/співробітникам. Цей компонент має три підкомпоненти:

1 Навігаційна модель (визначає структуру системи гіпермедіа і описує, як пройти через різні вузли системи).

2 Модель презентації (використовується для адаптації макета до візуальної лінії з перевагами або потребами користувача).

3 Модель вмісту (для надання додаткового контенту, аналогічного контенту, альтернативного контенту або приховування контенту).



Рис. 4. Процес формування курсу

Проведений аналіз науково-практичних розробок з математичного забезпечення моделей навчання в сучасних розподілених системах доводить необхідність створення спеціалізованого математичного та програмного забезпечення, яке має бути спроможним описувати процеси управління навчальним контентом. Для досягнення мети адаптації система повинна мати інформацію для аналізу інтересів та вподобань користувача, історії його взаємодії з системою, будь-яку інформацію, до якої вона може адаптуватися. Окремим питанням постають види та варіанти надання інформації, вирішення питання релевантності навчальної інформації. За своєю структурою адаптивні системи дуже різноманітні, від надзвичайно складних до більш простих, з меншою кількістю компонентів та параметрів, що враховуються при побудові сценарію навчання. Основними компонентами адаптивних навчальних систем є моделі користувача та предметної галузі, а також, в залежності від типу системи, база знань чи інші математичні моделі.

Існує велика кількість робіт, де визнано необхідність дослідження моделей та методів, створення сучасних навчальних систем на основі штучного інтелекту для впровадження їх при створенні інтелектуальних систем безперервного навчання. Основою для даного дослідження з точки зору вибору математичного апарату з моделювання процесів обробки даних, як основи побудови інформаційної технології, обрано метод компараторної ідентифікації та його інструментарій – алгебру скінченних предикатів [15].

Таким чином, було розглянуто базові засади розв'язання задачі ідентифікації навчальних об'єктів для систем електронного адаптивного навчання. Вирішення задачі ідентифікації навчального контенту доцільно на основі побудови відповідних алгебро-логічних моделей.

Висновки

У статті проаналізовано існуючі підходи до електронного навчання, яке надає здобувачам освіти доступ до сучасних джерел інформації, підвищує ефективність самостійної роботи, дає абсолютно нові можливості для творчого самовираження, знаходження та закріплення різних професійних навичок, а викладачам, в свою чергу, дозволяє реалізувати абсолютно нові форми і методи навчання із застосуванням концептуального і математичного моделювання явищ і процесів. На основі аналізу вимог до систем дистанційного навчання запропонована структура адаптивної системи електронного навчання, описані компоненти даної системи. Визначено інформаційні процеси, які протікають в даній системі.

З урахуванням наведеного, про електронне навчання можна говорити як про педагогічну технологію, що заснована на принципі самостійного навчання і ставить за мету творчий розвиток особистості, обов'язковою умовою ефективного використання якого є якісне інформаційне наповнення, що забезпечує підтримку процесу навчання.

Можна підкреслити наступні переваги від використання адаптивної системи електронного навчання: різні університети можуть використовувати цю систему; учні мають однакові шанси мати різні навчальні об'єкти, незалежно від того, в якому університеті вони відвідують навчальну дисципліну; навчальні об'єкти будуть надаватися адаптивним способом; інструктори матимуть краще середовище для обміну навчальними об'єктами між собою. Система буде корисною для користувачів з обмеженими можливостями. Учень має свій власний навчальний простір, який він може контролювати та керувати багатьма способами, змінюючи зміст навчання та цілі навчання.

Список літератури:

- [1] Концепція розвитку дистанційної освіти в Україні (затверджено Постановою МОН України В.Г. Кременем 20 грудня 2000 р.)
- [2] Про Національну програму інформатизації : Закон України від 4 лют. 1998 р. № 74/98-ВР // Відомості Верховної Ради України. – 1998. – №27-28. – Ст. 181.
- [3] Положення про дистанційне навчання (Затверджено наказом Міністерства освіти і науки України 21.01.2004 № 40) [Електронний ресурс] // Режим доступу : <http://zakon4.rada.gov.ua/laws/show/z0703-13#n18>
- [4] Що таке дистанційна освіта: як вона працює? [Електронний ресурс] // Режим доступу : <http://www.vsemisto.info/osvita/2355-sho-take-vysha-osvita-jakvona-prazjuje>
- [5] N. Khairova and N. Sharonova, “Modeling a logical network of relations of semantic items in superphrasal unities,” 2011 9th East-West Design & Test Symposium (EWDTS), Sevastopol, 2011, pp. 360-365. doi: 10.1109/EWDTS.2011.6116585
- [6] Клокар Н. Методологічні основи запровадження дистанційного навчання в системі підвищення кваліфікації / Н. Клокар // Шлях освіти. – 2012. – № 4 (46). – С. 38-41.
- [7] Відділ дослідження і проектування навчального середовища ІТЗН АПН України [Електронний ресурс]. – Режим доступу : <http://www.ime.eduua.net/nauk.html>
- [8] Plummer D. C. Cloud Computing Confusion Leads to Opportunity / Daryl C. Plummer, David W. Cearley, David Mitchell Smith – Report № G00159034. – Gartner Group, 2008 [Electronic resource]. – Access mode : http://www.gartner.com/it/content/868800/868812/cloud_computing_confusion.pdf

- [9] Волокита А. Специфіка інформаційних систем на основі технології cloud computing [Електронний ресурс] / А. Волокита, В. Мухін, В. Стешин. – Режим доступу : http://archive.nbuv.gov.ua/portal/natural/vcndtu/2011_53/29.htm.
- [10] Заячук Я.І. Використання дистанційних віртуальних практикумів у навчальному процесі ВНЗ / Я.І. Заячук, О.В. Мойсеєнко, В.В. Бойчук // Сучасні інформаційні технології в дистанційній освіті: IV Всеукр. наук.-практ. семінар, Івано-Франківськ, 21-23 вересня 2015р
- [11] Anderson, T., Rourke, L., Garrison, D. R., Archer, W. Assessing Teaching presence in a Computer Conference Environment // Journal of asynchronous learning networks. – 2001. – 5(2), PP. 1–17.
- [12] Dewey, J. (1938). Experience and Education. Toronto: Collier-MacMillan Canada Ltd. Поздняков В.А. Практична реалізація адаптованої системи якості підготовки smart-суспільства/ В.А. Поздняков // Вісник наукових досліджень. – 2013. – №6. – С. 70-75.
- [13] Manuel J. Sánchez-Franco. Users' intrinsic and extrinsic drivers to use a webbased educational environment [Electronic resource] / Manuel J. SánchezFranco, Begoña Peral-Peral, Ángel F. Villarejo-Ramos // Computers & Education. – – 2014. – Vol. 74. – May. – P. 81–97. – Access mode: <http://www.sciencedirect.com/science/article/pii/S0360131514000293> (last access: 20.08.18).
- [14] Quendler E. Learning as a lifelong process-meeting the challenges of the changing employability landscape: competences, skills and knowledge for sustainable development / Quendler E., Lamb M. // International Journal of Continuing Engineering Education and Life Long Learning. – № 26.3. – 2016. – p. 273-293.
- [15] Kyrychenko, Iryna. Formal Representation of Knowledge for Infocommunication Computerized Training Systems / Igor Shubin, Iryna Kyrychenko, Petr Goncharov, Stanislav Snisar // PIC S&T'2017, October 10-13, Kharkiv Ukraine, PP. 287-291.

Надійшла до редколегії 14.09.2018

УДК 519.688



І.А. Ревенчук, К.В. Перцьова, О.І. Маренич

ХНУРЕ, м. Харків, Україна, ilona.revenchuk@nure.ua

ХНУРЕ, м. Харків, Україна, karyna.pertsova@nure.ua

ХНУРЕ, м. Харків, Україна, oleh.marenych@nure.ua

ПРОГРАМНА РЕАЛІЗАЦІЯ КЛАСТЕРИЗАЦІЇ ПОШУКОВИХ ЗАПИТІВ

В статті запропонована програмна реалізація кластеризації пошукових запитів мовою Python, кластеризація за складом фраз виконана на базі алгоритма k-means, ієрархічна кластеризація представлена по SERP. Цей аналіз корисний для власників інтернет магазинів, яким важливо знати, що шукають в електронних магазинах клієнти та за якими пріоритетами.

МАШИННЕ НАВЧАННЯ, КЛАСТЕРИЗАЦІЯ ПОШУКОВИХ ЗАПИТІВ, КЛАСТЕРИЗАЦІЯ ЗА СКЛАДОМ ФРАЗ, АЛГОРИТМОМ K-MEANS, ІЄРАРХІЧНА КЛАСТЕРИЗАЦІЯ SERP, PYTHON

Ревенчук І.А., Перцева К.В., Маренич О.І. Програмная реализация кластеризации поисковых запросов. В статье предложена программная реализация кластеризации поисковых запросов на языке Python, кластеризация по составу фраз выполнена на базе алгоритма k-means, Иерархическая кластеризация представлена по SERP. Этот анализ полезен для владельцев интернет магазинов, которым важно знать, что ищут в электронных магазинах клиенты и по каким приоритетам.

МАШИННОЕ ОБУЧЕНИЕ, КЛАСТЕРИЗАЦИЯ ПОИСКОВЫХ ЗАПРОСОВ, КЛАСТЕРИЗАЦИИ ПО СОСТАВУ ФРАЗЫ, АЛГОРИТМ K-MEANS, ИЕРАРХИЧЕСКАЯ КЛАСТЕРИЗАЦИЯ SERP, PYTHON

Revenchuk I.A., Pertseva K.V., Marenych O.I. Software implementation of search query clustering. A software implementation of the clustering of search queries by Python were proposes in this article. Clustering based on the composition of phrases is based on the k-means algorithm. Hierarchical clustering is represented by SERP. This analysis is useful for owners of online stores, for whom it is important to know what customers are looking for in electronic stores and by what priorities.

MACHINE LEARNING, CLUSTERIZATION OF SEARCH REQUESTS, FRAZ COMPOSITION CLUSTERIZATION, K-MEANS ALGORITHM, SERP PERSONAL EXPRESSION, PYTHON.

Вступ

За повідомленнями [1] дослідних фірм США, практично 97% досліджень, спрямовані на виявлення маркетингових проблем щодо оцінки потенціалу ринку, частки ринку, характеристик ринку. Це дослідження базуються на статистичній інформації споживачів електронних ринків та соціальних медіа. Тому що за аналізом [2] кожна друга особа матимете акаунт в соціальних медіа, що призведе до збільшення прибутку, що буде надходитиме від користувачів соціальних мереж.

Дослідження продажів дозволяють вчасно виявити ті чи інші тенденції ринку. Цей аналіз особливо корисний, якщо він проводиться в різних напрямках. Наприклад, аналіз дозволяє виявити вузькі місця в мережі поширення, щоб своєчасно вжити заходів щодо підвищення ефективності каналів розповсюдження товару.

Прогнозування, дослідження ринкових тенденцій дозволяє оцінити ситуацію трендів в майбутньому та домогтися серйозних ринкових переваг.

Для власників е-ринків важливо знати, що шукають в електронних магазинах клієнти та за якими пріоритетами.

Для вирішення цієї проблеми необхідно проаналізувати всі пошукові запити а також використати алгоритм k-means для кластеризації за складом фраз.

При зборі семантичного ядра для сайту, найбільш важливим етапом є обробка, чистка та структуризація отриманого списку запитів. Це дозволяє відсіяти слова-пустушки, нецільові запити та інше «сміття», використання якого знижує віддачу від роботи і не несе будь-які цінні папери для проекту.

Кластеризація пошукових запитів — це процес формування на основі загального списку ключових фраз окремих груп (кластерів), які пов'язані між собою описом об'єкта, здоровим глуздом і контекстом використання.

Потім, кожній групі ключів призначається окрема сторінка на сайті (нова або існуюча), під якою і здійснюється її подальша оптимізація. Просування кожної з посадочних тільки під одну групу фраз, дозволяє зробити їх максимально релевантними, що позитивно позначається на видимості і позиціях в пошуку.

Основний метод, який лежить в основі алгоритмів SEO-інструментів, призначених для вирішення такого завдання, є кластеризація на основі пошукової видачі. Аналізуються сторінки з пошукових систем, наприклад Google показують у відповідь на конкретний запит, а потім, ґрунтуючись на кількості збігів в SERP, ключові фрази об'єднуються в групи.

Кластеризація семантичного ядра дозволяє об'єднати різні по входженням слова — «викуп

автомобіля, викуп авто, викуп машини» — але однакові за змістом запити (синоніми), в одну групу. Таким чином, можна уникнути ситуації, коли на один і той же запит відповідають кілька сторінок в рамках сайту, що в підсумку тільки заважає просуванню.

Крім того, запити діляться на комерційні та інформаційні. Видача для них буде різною і не варто намагатися просувати їх на одній сторінці, тому що результати будуть не дуже гарні.

Звичайно, розраховувати на ідеальний результат не варто, і завжди потрібна фінальна перевірка з боку SEO-фахівця. Але в цілому, економія часу для оптимізатора величезна, особливо, якщо семантичне ядро налічує тисячі і десятки тисяч запитів.

1. Кластеризація пошукових запитів на Python

Векторне подання ключових слів. Щоб зробити наші ключові слова зручними для обробки, необхідно провести їх векторизацію, тобто всі ключові фрази розбиваються на унікальні слова і кодуються. По суті вони перетворюються в великий список і далі замість кожного ключового слова ми розміщуємо довгий рядок цифр, що відповідає нашому списку всіх унікальних слів. Якщо слово є в фразі — то ставимо 1, якщо немає, то 0. Виходить щось на зразок таблиці 1.

Для отримання цієї матриці скористаємося готовими бібліотеками. Таку матрицю можна отримати за допомогою класу `CountVectorizer` з бібліотеки `scikit-learn` (рис. 1).

Таблиця 1

Перелік ключових слів у пошукових запитах

	Магазин	машин	моделей	моделі	Україні	радіокеровані	радіокерованих
Магазин радіокеровані моделі	1	0	0	1	0	1	0
Магазин радіокерованих моделей	1	1	0	0	0	0	1
Магазин радіокерованих моделей в Україні	1	0	1	0	1	0	1

Тут ми створюємо два різних векторизатора, які відрізняються лише одним параметром — мінімальною частотою слова. Для векторизатора `cv` мінімальна частота слова, яке він враховує дорівнює 1, а для `cv2` — 2, тобто все слова з частотою менше двох — не враховуються при побудові вектора. Якщо слово зустрічається тільки 1 раз у всіх ключових фразах — то немає сенсу займати оперативну пам'ять подібною інформацією. Ми не зможемо знайти будь-які відповідні перетини по слову, у якого ніде немає пари.

Результатом обробки списку `phrases`, на якому ми будемо розбирати приклади векторизації видно, як змінилися підсумкові таблиці для порівнянь (рис. 2).

Слова, які використовувалися лише 1 раз, були відкинута, і наші багатослівні ключові фрази перетворилися в двусловні.

```
In [1]: from sklearn.feature_extraction.text import CountVectorizer
import pandas as pd

In [2]: cv = CountVectorizer(min_df= 1)
cv2 = CountVectorizer(min_df= 2)

In [3]: CountVectorizer()

Out[3]: CountVectorizer(analyzer='word', binary=False, decode_error='strict',
dtype=<class 'numpy.int64'>, encoding='utf-8', input='content',
lowercase=True, max_df=1.0, max_features=None, min_df=1,
ngram_range=(1, 1), preprocessor=None, stop_words=None,
strip_accents=None, token_pattern='(?u)\\b\\w+\\b',
tokenizer=None, vocabulary=None)

In [4]: phrases = [
'магазин радиоуправляемые модели',
'магазин радиоуправляемых машин',
'магазин радиоуправляемых моделей в Украине'
]

In [5]: X = cv.fit_transform(phrases)
X2 = cv2.fit_transform(phrases)
```

Рис. 1. Приклад коду класу `CountVectorizer` з бібліотеки `scikit-learn` для створення векторизації

Можна помітити, що багато значущих слова були порізані налаштуваннями, векторизатора, і мали смислову пару, але машина сама по собі не здатна побачити що «моделей» і «моделі» це просто різні відміни одного і того ж слова. Для того, щоб усунути подібні помилки ми попередньо проведемо процедуру, яка називається лематизації — приведення словоформи в її нормальну форму (рис. 3).

Таким нескладним способом можна трансформувати всі ключові фрази в нормалізований вид і перевчити векторизатор (рис. 4).

Загалом отримуємо такі матриці (рис. 5).

Після застосування другого векторизатора, вдалося зберегти набагато більше корисної інформації. Він добре бачить слово «модель» і різні форми «радіокерованого».

Оскільки SEO-фахівці знають, що кількість слів рідко переходить в якість замість CountVectroizer краще використовувати TfidfVectorizer. Він дуже схожий на попередній векторизатор, але замість числа 1 або 0 проставляє значимість кожного слова розраховуючи її по Tf-Idf.

Використовується він так само як і звичайний

```
In [6]: data = pd.DataFrame(data=X.toarray(), columns=cv.get_feature_names(), index = phrases)
data2 = pd.DataFrame(data=X2.toarray(), columns=cv2.get_feature_names(), index = phrases)

In [8]: data
Out[8]:
```

	магазин	машин	моделей	модели	Украине	радиоуправляемые	радиоуправляемых
магазин радиоуправляемые модели	1	0	0	1	0	1	0
магазин радиоуправляемых машин	1	1	0	0	0	0	1
магазин радиоуправляемых моделей в Украине	1	0	1	0	1	0	1

```
In [7]: data2
Out[7]:
```

	магазин	радиоуправляемых
магазин радиоуправляемые модели	1	0
магазин радиоуправляемых машин	1	1
магазин радиоуправляемых моделей в Украине	1	1

Рис. 2. Підсумкові таблиці з ключовими словами для порівнянь

```
import pymorphy2
m = pymorphy2.MorphAnalyzer()

m.parse('радиоуправляемые')

[Parse(word='радиоуправляемые', tag=OpencorporaTag('ADJF plur,nomn'), normal_form='радиоуправляемый', score=0.5, methods_stack=((DictionaryAnalyzer, 'радиоуправляемые', 10, 20))),
 Parse(word='радиоуправляемые', tag=OpencorporaTag('ADJF inan,plur,accs'), normal_form='радиоуправляемый', score=0.5, methods_stack=((DictionaryAnalyzer, 'радиоуправляемые', 10, 24)))]

m.parse('радиоуправляемых')[0].normal_form
'радиоуправляемый'

m.parse('радиоуправляемые')[0].normal_form
'радиоуправляемый'
```

Рис. 3. Приведення словоформи в її нормальну форму

```
lemm_phrases = [' '.join([m.parse(word)[0].normal_form for word in x.split()]) for x in phrases]

lemm_phrases
['магазин радиоуправляемый модель',
 'магазин радиоуправляемый машина',
 'магазин радиоуправляемый модель в москва']

X = cv.fit_transform(lemm_phrases)
X2 = cv2.fit_transform(lemm_phrases)

data = pd.DataFrame(data=X.toarray(), columns=cv.get_feature_names(), index = phrases)
data2 = pd.DataFrame(data=X2.toarray(), columns=cv2.get_feature_names(), index = phrases)
```

Рис. 4. Трансформація ключових фраз в нормалізований вид

CountVectorizer, але повертає інші результати (рис. 6).

Отже необхідний сумісний лемматизатор і TfidfVectorizer, як один клас, який дозволить проводити перетворення одразу.

Взагалі, для задач кластеризації в цьому немає необхідності, але такий код, виглядає більш зручним (рис. 7). в порівнянні з прикладами (рис. 1–6).

У всіх розібраних векторизатор є кілька загальних налаштувань:

– stop-words — список слів, які не враховуватимуться при векторизації;

– token_pattern — регулярний вираз, за яким рядок розбивається на токени. Зазвичай це просто поділ на слова, але можуть бути виділені і інші сутності;

– max_df — токени мають частотність вище цього значення не будуть враховуватися. Можна вказати відсоток через коефіцієнт — 0.9 означитиме що 10% найбільш частих слів будуть відкинуті;

– min_df — токени мають частотність нижче цього значення не будуть враховуватися. Можна вказати відсоток через коефіцієнт — 0.1 означитиме що 10% найбільш рідкісних слів будуть відкинуті.

data					
	магазин	машина	модель	Україна	радиоуправляемый
магазин радиоуправляемые модели	1	0	1	0	1
магазин радиоуправляемых машин	1	1	0	0	1
магазин радиоуправляемых моделей в Украине	1	0	1	1	1

data2			
	магазин	модель	радиоуправляемый
магазин радиоуправляемые модели	1	1	1
магазин радиоуправляемых машин	1	0	1
магазин радиоуправляемых моделей в Украине	1	1	1

Рис. 5. Остаточні матриці

```
from sklearn.feature_extraction.text import TfidfVectorizer

tfidf_v = TfidfVectorizer(min_df=1)

X = tfidf_v.fit_transform(lemm_phrases)

data = pd.DataFrame(data=X.toarray(), columns=tfidf_v.get_feature_names(), index = phrases)
data
```

	магазин	машина	модель	Україна	радиоуправляемый
магазин радиоуправляемые модели	0.522842	0.000000	0.673255	0.000000	0.522842
магазин радиоуправляемых машин	0.453295	0.767495	0.000000	0.000000	0.453295
магазин радиоуправляемых моделей в Украине	0.391484	0.000000	0.504107	0.66284	0.391484

Рис. 6. Приклад використання TfidfVectorizer

```
class LemmTfidfVectorizer(TfidfVectorizer):
    def build_analyzer(self):
        analyzer = super(TfidfVectorizer, self).build_analyzer()
        return lambda x: (m.parse(word)[0].normal_form for word in analyzer(x))

tfidf_lemm_v = LemmTfidfVectorizer(min_df=1)
X=tfidf_lemm_v.fit_transform(phrases)
data = pd.DataFrame(data=X.toarray(), columns=tfidf_lemm_v.get_feature_names(), index = phrases)
data
```

	магазин	машина	модель	Україна	радиоуправляемый
магазин радиоуправляемые модели	0.522842	0.000000	0.673255	0.000000	0.522842
магазин радиоуправляемых машин	0.453295	0.767495	0.000000	0.000000	0.453295
магазин радиоуправляемых моделей в Украине	0.391484	0.000000	0.504107	0.66284	0.391484

Рис. 7. Сумісний лемматизатор і TfidfVectorizer

2. Кластеризація за складом фраз алгоритмом k-means

Кластеризація за складом фраз дозволить виділити ті словосполучення, які між собою сильно пов'язані і мають велику кількість перетинів значущих слів.

Для подальшої роботи використовуватиме клас KMeans з бібліотеки scikit-learn. Нижче скріншот (рис. 8) підготовчої частини, де ми довантажимо ключові слова з зовнішнього текстового файлу, створюємо клас для векторизації і заповнюємо список стоп-слів, якщо він у вас є.

Для налаштування алгоритму k-means зручно використовувати такі параметри:

- `n_clusters` — кількість кластерів, на які будуть ділитися дані. Мінусом даного алгоритму є необхідність спочатку вказати, на яку кількість груп будуть поділені слова;

- `max_iter` — максимальна кількість ітерацій. Робота буде примусово зупинено при досягненні цього числа. Іноді алгоритм «залипає» на деяких видах даних, і вказується конкретне статичне число, яке дозволяє уникнути нескінченної роботи;

- `n_init` — кількість повторних ініціалізованих алгоритмів з різними початковими центроїдами. Чим більше ініціалізацій — тим краще кластера описують реальну структуру даних і тим довше працює алгоритм;

- `tol` — довірчий ліміт, при досягненні якого кластеризація буде зупинена;

- `n_jobs` — кількість паралельних потоків роботи алгоритму;

- `random_state` — параметр дозволяє домогтися відтворюваності одних і тих же результатів при різних запусках алгоритму.

Сама кластеризація багато місця не займає (рис. 9).

```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.cluster import KMeans
from pymorphy2 import MorphAnalyzer
import numpy as np
m = MorphAnalyzer()
```

Завантажуємо ключові слова

```
keywords = [x.rstrip() for x in open('keywords.txt')]
```

Створюємо клас для підготовки ключових слів

```
class LemmTfidfVectorizer(TfidfVectorizer):
    def build_analyzer(self):
        analyzer = super(TfidfVectorizer, self).build_analyzer()
        return lambda x: (m.parse(word)[0].normal_form for word in analyzer(x))
```

За необхідністю можна використовувати перелік стоп-слів, які будуть відброшені при векторизації

```
stop_words = ['и', 'в', 'во', 'не', 'что', 'он', 'на', 'я', 'с', 'со', 'как', 'а', 'то', 'все', 'она', 'так', 'его', \
              'но', 'да', 'ты', 'к', 'у', 'же', 'вы', 'за', 'бы', 'по', 'только', 'ее', 'мне', 'было', 'вот', 'от', \
              'меня', 'еще', 'нет', 'о', 'из', 'ему', 'теперь', 'когда', 'даже', 'ну', 'вдруг', 'ли', 'если', 'уже', \
```

Рис. 8. Підготовчий етап створення класу векторизації

Векторизуємо ключові слова

```
tfidf_lemm_vec = LemmTfidfVectorizer(min_df=2, stop_words= stop_words)
X = tfidf_lemm_vec.fit_transform(keywords)
```

Створюємо об'єкт кластеризатора і проводимо пошук груп

```
k_means = KMeans(n_clusters=int(np.round(np.divide(len(keywords), 5))),
                 init='k-means++',
                 n_init=10,
                 max_iter=300,
                 tol=0.0001,
                 n_jobs=4)
k_means.fit(X)
```

Збираємо ключові слова у групи і виводимо для аналізу

```
dct = {}
for key, label in zip(keywords, k_means.labels_):
    dct[label] = dct.get(label, [])+[key]
```

Рис. 9. Приклад кластеризації

За допомогою останнього сніппета зберігаємо результати в cvs (рис. 10).

CVS (Concurrent Versions System, «Система одночасних версій») — програмний продукт, що відноситься до розряду систем управління версіями (англ. Version control system).

Зберігає історію змін певного набору файлів, як правило, вихідного коду програмного забезпечення, і полегшує спільну роботу групи людей (часто — програмістів) над одним проектом. CVS популярна в світі відкритого ПЗ. Поширюється на умовах ліцензії GNU GPL.

Крім того, використовуючи цей алгоритм можна вирішити ще одну задачу.

Наприклад, маємо об'єкти — інформація про частіші пошукові запити на aliexpress, 2178 об'єктів, 3 ознаки: частота покупок, геолокація покупця, сума покупки за середнім чеком.

За допомогою кластеризації в цій базі можна знайти інформацію про більш активних покупців за географічною ознакою та частотою покупок (рис. 11).

Це допоможе уявити геолокацію активних покупців aliexpress.com.

3. Ієрархічна кластеризація SERP

Ієрархічні алгоритми будують не одне розбиття вибірки на непересічні кластери, а систему вкладеного розбиття. На виході отримуємо дерево кластерів, коренем якого є вся вибірка, а листям — найбільш дрібні кластери.

Для початку потрібно зібрати ТОП з будь-якої пошукової системи. Для прикладу був обраний топ Google, тому що з ним зручно працювати через XML. За глибиною — топ 10, хоча нормальні результати виходять і на топ 20-30, але от далі

```
for x in (0, 3, 9, 13, 37):
    print(dct[x])
    print('='*8)

['запчасти для самолетов', 'запчасти для радиоуправляемых самолетов', 'пропеллер 7x8 для самолета']
=====
['где купить машину для дрифта', 'машины на радиоуправлении для дрифта', 'гс модели для дрифта купить', 'машины на пульте д
ля дрифта', 'радиоуправляемые машины для дрифта', 'купить радиоуправляемую машину для дрифта', 'радиоуправляемые машины для
дрифта цена', 'купить радиоуправляемые машины для дрифта', 'дешевые радиоуправляемые машины для дрифта']
=====
['машины на пульте управления дс', 'модели машин на пульте управления', 'гоночные машины на пульте управления', 'машины на
пульте управления на бензине', 'бензиновые машины на пульте управления', 'большие машины на пульте управления', 'игрушки м
ашины на пульте управления', 'машины на пульте', 'машина на пульте управления цена', 'машины на пульте управления', 'продаж
а машин на пульте управления']
=====
['купить радиоуправляемую подводную лодку', 'радиоуправляемые модели подводных лодок', 'радиоуправляемые подводные лодки',
'купить подводную лодку', 'куплю подводную лодку', 'подводные радиоуправляемые модели', 'где купить подводную лодку', 'моде
ль подводной лодки купить', 'радиоуправляемая игрушка подводная лодка']
=====
['двигатели kyosho', 'двигатели для автомоделей', 'двигатели для моделей', 'двигатели для самолетов', 'двигатели дс для мо
делей', 'двигатели для авиамodelей']
=====
```

Збереження результатів в csv

```
with open('k-means.csv', 'w') as f:
    for cln in dct:
        for keyword in dct[cln]:
            f.write('{};{}\n'.format(cln, keyword))
```

Рис. 10. Приклад збереження результатів в CVS

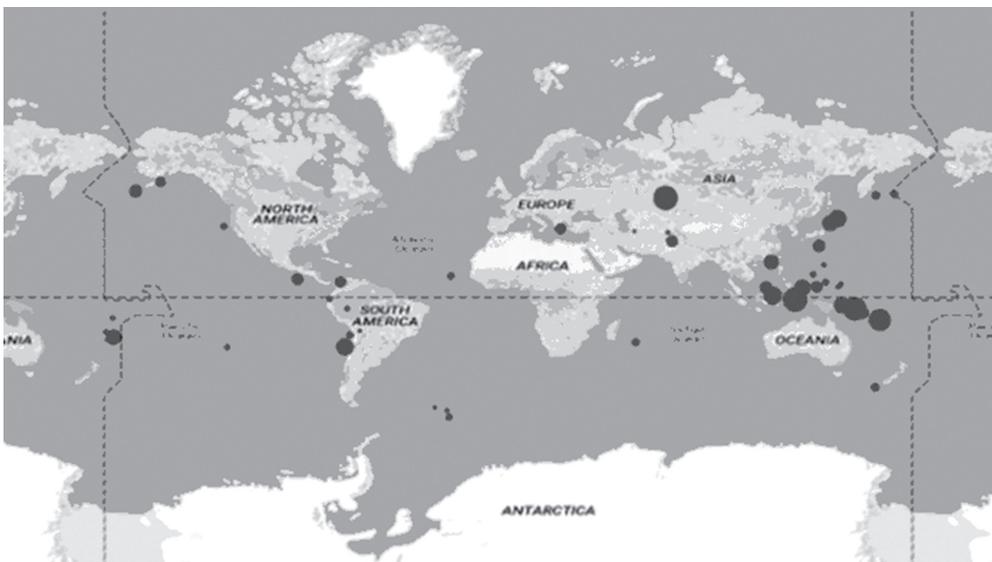


Рис. 11. Метод K-means і 50 кластерів

починається хаос в угрупованні. Після того, як зібрали серпи у нас виходять такі конструкції:

- keywords — список в якому лежать наші ключові слова;
- serps — список в якому лежать на відповідних ключам рядках серпи, в яких все url серпа просто розділені між собою пробілом.

Тут необхідним буде можливість вказувати власне регулярний вираз для розбиття фрази на специфічні маркери. Для нас кожним унікальним

токеном буде будь-якої унікальний url (рис. 12).

Далі проводимо кластеризацію, обчислюємо групування та виводимо кластери (рис. 13).

Використання даного алгоритму так само дозволяє отримати адекватні результати угруповання (рис. 14).

За бажанням можна візуалізувати дерево, яке вийшло в процесі кластеризації (рис. 15).

Візуалізація даних — це наочне уявлення масивів різної інформації. Існує кілька типів візуалізації:

```
from sklearn.feature_extraction.text import CountVectorizer
from scipy.cluster.hierarchy import linkage, dendrogram, fcluster
import numpy as np
```

Збираємо для кожного ключового слова серп через xml

```
keywords = []
serps = []
for key in open('keywords.txt'):
    key = key.lower().rstrip()
    serp = get_serp(key, service='xmlproxy')
    keywords.append(key)
    serps.append(' '.join(serp))
```

Преобразуємо строки url в матрицю ознак

```
vectorizer = CountVectorizer(min_df=2, token_pattern='[^\ ]+')
X = vectorizer.fit_transform(serps)
```

Рис. 12. Кластеризація SERP. Перетворення url в матрицю ознак

```
link_M = linkage(X.toarray(), method='ward')
```

```
clusters = fcluster(link_M,
                    int(np.round(np.divide(len(keywords), 5))),
                    criterion='maxclust')
```

```
dct = {}
for key, label in zip(keywords, clusters):
    dct[label] = dct.get(label, []) + [key]
```

Рис. 13. Кластеризація SERP. Обчислення групування та вивів кластерів

```
for x in (1, 5, 16, 26, 50):
    print(dct[x])
    print('='*8)
```

```
['радиоуправляемые автомобили с двс', 'радиоуправляемые модели автомобилей двс', 'радиоуправляемые модели автомобилей с двс', 'радиоуправляемые модели с двс авто', 'модели автомобилей с двс']
=====
['бензиновые радиоуправляемые модели', 'радиоуправляемые бензиновые модели автомобилей', 'радиоуправляемые модели автомобилей на бензине', 'бензиновые радиоуправляемые машины', 'радиоуправляемые автомобили на бензине', 'радиоуправляемые модели на бензине']
=====
['магазин радиоуправляемых моделей', 'магазины радиоуправляемых моделей', 'магазин радиоуправляемых моделей', 'радиоуправляемые модели', 'управляемые модели', 'радиоуправляемые модели купить', 'радиоуправляемые модели', 'радиоуправляемых моделей']
=====
['вертолет радиоуправляемый', 'купить радиоуправляемый вертолет', 'купить радиоуправляемые вертолеты', 'куплю радиоуправляемый вертолет', 'радиоуправляемые вертолеты дешево', 'радиоуправляемые вертолеты', 'купить радиоуправляемый вертолет', 'радиоуправляемые вертолеты']
=====
['радиоуправляемая модель самолета', 'купить радиоуправляемый самолет', 'куплю радиоуправляемый самолет', 'радиоуправляемые самолеты где купить', 'самолет на радиоуправлении где купить', 'радиоуправляемые самолеты купить', 'где купить радиоуправляемый самолет', 'радиоуправляемые самолеты', 'купить радиоуправляемый самолёт', 'радиоуправляемый самолет']
=====
```

Рис. 14. Кластеризація SERP. Результати угруповання

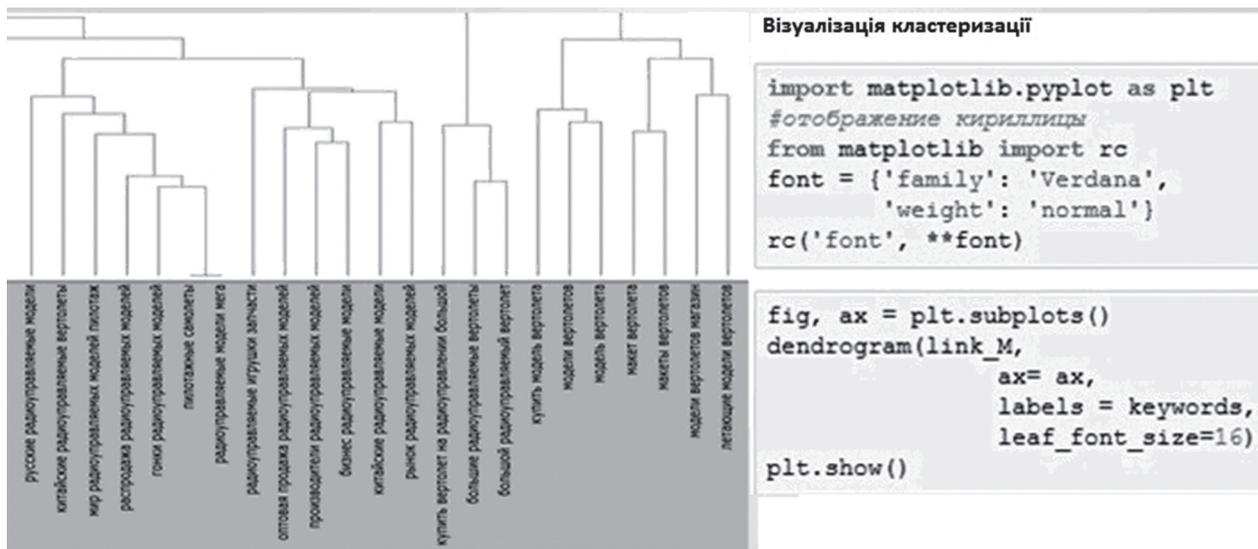


Рис. 15. Візуалізація кластеризації

— звичайне візуальне уявлення кількісної інформації в схематичній формі. До цієї групи можна віднести всім відомі кругові та лінійні діаграми, гістограми і спектрограми, таблиці і різні точкові графіки;

— дані при візуалізації можуть бути перетворені в форму, посилює сприйняття і аналіз цієї інформації. Наприклад, карта і полярний графік, тимчасова лінія і графік з паралельними осями, діаграма Ейлера;

— концептуальна візуалізація дозволяє розробляти складні концепції, ідеї і плани за допомогою концептуальних карт, діаграм Ганта, графів з мінімальним шляхом та інших подібних видів діаграм;

— стратегічна візуалізація переводить в візуальну форму різні дані про аспекти роботи організацій. Це всілякі діаграми продуктивності, життєвого циклу і графіки структур організацій;

— графічно організувати структурну інформацію за допомогою пірамід, дерев і карт даних допоможе метафорична візуалізація;

— комбінована візуалізація дозволяє об'єднати кілька складних графіків в одну схему.

Візуалізація кластеризації відноситься до графічно організованої структурної інформації, що дозволяє краще сприймати і ефективно донести ідею.

Висновки

В статті запропонована програмна реалізація кластеризації пошукових запитів мовою Python, кластеризація за складом фраз виконана на базі алгоритма k-means, ієрархічна кластеризація представлена по SERP.

Отримані дані з кластерів будуть подальше використовуватися власниками електронного бізнесу з метою більш ефективного просування товарів та послуг мережею інтернет.

Це прогнозування дозволить керівникам електронного бізнесу, завдяки отриманій інформації, отримувати уявлення про клієнтів та за допомогою швидкого, послідовного, і інтерактивного доступу до інформації підвищити ефективність ведення бізнесу для визначеного сегменту споживачів, у тому числі і користувачів соціальних мереж та направити PR стратегію на утримання клієнтів, які знаходяться у зоні відтоку користувачів компанії та покупців.

Список літератури:

- [1] Digital Market Outlook. — оновлення: 5.12.2018. URL:<https://www.statista.com/outlook/digital-markets>.
- [2] Social Networking Reaches Nearly One in Four Around the World.-оновлення: 5.12.2018. URL:<https://www.emarketer.com/Article/Social-Networking-Reaches-Nearly-One-Four-Around-World/1009976>

Надійшла до редколегії 16.10.2018

УДК 53.082.9



Ю.С. Курской

Харьковский национальный университет радиоэлектроники,
пр. Науки, 14, 61166 Харьков, Украина, yurii.kurskyi@nure.ua

МОДЕЛЬ ИЗМЕРЕНИЯ ДЛЯ СИСТЕМ С ФУНКЦИЕЙ САМООРГАНИЗАЦИИ

К важнейшим научным задачам относятся задачи исследований и измерений в эволюционирующих нелинейных динамических системах с функцией самоорганизации. В работе представлена модель измерения для таких систем. В рамках модели проводятся измерения динамических переменных системы в равновесном и возбуждённом состояниях. Результатом измерения являются интервалы значений динамических переменных, фрактальная размерность, энтропия Шеннона и время предсказания динамики. Для оценки эволюционных процессов в модель введены энтропийные и временные шкалы. В качестве количественной оценки состояния системы используется время её возврата в состояние равновесия после прекращения внешнего нормированного воздействия. Предложен энтропийный метод оценки нормированного внешнего воздействия.

НЕЛИНЕЙНАЯ ДИНАМИЧЕСКАЯ СИСТЕМА; МОДЕЛЬ ИЗМЕРЕНИЯ; ФРАКТАЛЬНАЯ РАЗМЕРНОСТЬ; ЭНТРОПИЯ ШЕННОНА

Ю.С. Курський. Модель вимірювання для системи з функцією самоорганізації. До найважливіших наукових завдань відносяться завдання досліджень та вимірювань у нелінійних динамічних системах із функціями еволюції та самоорганізації. У роботі представлена модель вимірювання для таких систем. В рамках моделі виконуються вимірювання динамічних змінних системи у рівноважному та збудженому станах. Результатом вимірювання є інтервали значень динамічних змінних, фрактальна розмірність, ентропія Шеннона і час передбачення динаміки. Для оцінки еволюційних процесів в модель введені ентропійна та часова шкали. В якості кількісної оцінки стану системи використовується час її повернення до стану рівноваги після припинення зовнішнього нормованого впливу. Запропоновано ентропійний метод оцінки нормованого зовнішнього впливу.

НЕЛІНІЙНА ДИНАМІЧНА СИСТЕМА; МОДЕЛЬ ВИМІРЮВАННЯ; ФРАКТАЛЬНА РОЗМІРНІСТЬ; ЕНТРОПІЯ ШЕННОНА

Yu.S. Kurskoy. Measurement model for system with selforganization function. The tasks of research and measurement of evolving nonlinear dynamical systems with a self-organization function is one of the most important scientific tasks. The paper presents a measurement model for such systems. The measurement model includes the measurements of dynamic variables of the system in the equilibrium and excited states. The results of the measurement are: the intervals of the dynamic variables values, the fractal dimension, the Shannon entropy and the dynamics prediction time. For assess of the evolutionary processes the model has the entropy and time scales. As a quantitative assessment of the system state the model offers the time of return to the equilibrium state after the normalized external influence. An entropy method for estimating normalized external influence is proposed too.

NONLINEAR DYNAMIC SYSTEM, MEASUREMENT MODEL, FRACTAL DIMENSION, SHANNON ENTROPY

Введение

К важнейшим научным задачам относятся задачи теоретического и практического обеспечения исследований, управления и прогнозирования динамики сложных физических, биологических и нелинейных динамических систем (НДС) иного происхождения. Примерами НДС являются: лазеры, солитоны, оптические системы конфиденциальной связи, климат, океан, геосфера, биопопуляции и живые организмы.

Несмотря на различную природу происхождения НДС демонстрируют общность свойств, таких как:

- нелинейность динамики и интервальность значений динамических переменных (ДП) — величин, характеризующих состояние НДС;
- сильная зависимость ДП от начальных условий и внешних воздействий;
- диссипативность;
- возможность хаотичной динамики;

– функция самоорганизации.

Общность свойств НДС позволила сформировать единые позиции к их исследованиям, которые отображены в таких междисциплинарных теориях как теория открытых систем, теория динамического хаоса, синергетика и ряде других [1].

Успешная реализация задач исследования, прогнозирования и управления НДС зависит от корректности измерений ДП. Несмотря на достигнутый прогресс в исследовании и описании НДС вопрос измерений и анализа их результатов долгое время оставался не проработанным. Данное обстоятельство позволяет считать создание методов, методик и моделей измерения в НДС важной теоретико-прикладной задачей, важной для развития теорий исследования НДС.

Системный подход к вопросу измерениям в НДС впервые был предложен в рамках нелинейной теории измерений (НТИ) [2-9]. В НТИ была обоснована необходимость дополнения классической

теории измерения принципами и моделями, соответствующими процессам, протекающим в реальных НДС. Было предложено использовать топологические инструменты теорий НДС для оценки результатов измерения величин со сложной динамикой [2]. НТИ содержит: принципы проведения измерений, физико-математические модели измерений и анализа их результатов (МИ); принципы оценки неопределённости измерения и представления результатов измерения [3, 4]. Подходы и модели НТИ предложено использовать для оценки характеристик электрических сетей [5], лазеров [6, 7], процесса лазерного охлаждения [8], измерения физического состояния человека [9].

Наиболее сложными для исследования являются эволюционирующие НДС с функцией самоорганизации. С течением времени, вследствие эволюции ДП, характеризующие возбуждённые и равновесные состояния, меняют свои значения. Меняется и время перехода системы из одного состояния в другое. В эволюционирующих системах время перехода между состояниями системы является маркером и численной мерой её эволюционных изменений, как со знаком плюс, так и минус.

Цель данной работы состоит в разработке модели измерения для систем с функцией самоорганизации на основе физико-математических положений теорий исследований и измерений в нелинейных динамических системах.

1. Постановка задачи

Рассмотрим открытую, диссипативную, эволюционирующую НДС с функцией самоорганизации. Её состояние характеризуются n -мерным вектором:

$$X[X_1(t), \dots, X_n(t)],$$

где $X_i(t)$ — i -я ДП; t — время.

Значения ДП связаны с начальными значениями $[X_1(t_0), \dots, X_n(t_0)]$ функцией эволюции F [10]:

$$F[X_1(t_0), \dots, X_n(t_0)] \rightarrow [X_1(t), \dots, X_n(t)]. \quad (1)$$

С течением времени наблюдения значения ДП $X_i(t)$ меняются (1), но находится в интервале $X_i^{\min} \leq X_i \leq X_i^{\max}$. Динамика ДП носит сложный, нелинейный характер. В ходе фазовых переходов ДП могут менять динамику с случайной на регулярную, хаотичную, и наоборот. У системы есть неустойчивые состояния равновесия, она обменивается энергией и информацией с окружающей средой. Система подвержена влиянию внешних факторов, вследствие которых значения ДП выходят за рамки $X_i^{\min} \leq X_i \leq X_i^{\max}$ и система переходит их устойчивого состояния в возбуждённое. После прекращения воздействия, благодаря функции самоорганизации, за некоторое время T , система возвращается в устойчивое состояние. Эволюция системы приводит к изменению интервалов

возможных значений ДП $[X_i^{\min}; X_i^{\max}]$ и времени возврата T .

Измерение состояния НДС представляется как эксперимент, целью которого является: определение интервалов возможных значений ДП $X_i^{\min} \leq X_i \leq X_i^{\max}$; классификация динамики процессов; определение времени T возвращения системы в устойчивое состояние после прекращения нормированного воздействия Q .

Для исследования и измерений ДП НДС в НТИ [3] используются топологические и информационные характеристики НДС: фрактальная размерность D [11], энтропия Шеннона H , время предсказуемости T_{for} (временной горизонт прогноза) [12].

Фрактальная размерность — одна из топологических характеристик геометрических объектов, процессов или фазового портрета НДС. Согласно определению Б. Мандельброта фрактал представляет собой объект, фрактальная размерность D (или размерность Хаусдорфа-Безиковича) которого больше его топологической размерности [11]. При проведении физических исследований фрактальные признаки могут быть обнаружены в структуре сигналов и полей, поведении функций, характеризующих распределение физических величин во времени и пространстве. Из этого следует возможность поиска фрактальной размерности, как особенного признака процессов или изображений.

Фрактальная размерность D используется для анализа временных рядов результатов измерений ДП. Для определения D ряда $\{x_i\}$ (где x_i — i -е значение величины x , $i=1, \dots, n$) используется метод нормированного размаха, выведенный эмпирическим путем П. Хёрстом. Анализ ряда $\{x_i\}$ позволяет получить показатель Хёрста H , связанный с D :

$$D = 2 - H \quad (2)$$

Значения показателя Хёрста находятся в интервале $0 \leq H \leq 1$ и определяются через отношение:

$$R/\sigma = (n/2)^H,$$

где: R — размах между максимальным и минимальным значениями функции приращения $x(i, n)$, n — количество измерений (состояний ДП), величина σ — среднее квадратичное отклонение:

$$R = \max_{1 \leq i \leq n} x(i, n) - \min_{1 \leq i \leq n} x(i, n); \quad x(i, n) = \sum_{i=1}^n (x_i - \bar{x}_i),$$

где \bar{x}_i — среднее арифметическое значений $\{x_i\}$ [11].

Фрактальная размерность (2) и фрактальная шкала с ключевыми значениями $D=1$, $D=1,5$, $D=2$, используются для классификации процессов:

- при $D=1$ процесс — строго детерминированный;
- при $D=2$ разброс значений очень велик, что не позволяет использовать методы обработки результатов измерений;

- при $D=1,5$ процесс является случайным;
- при $1 < D < 1,5$ процесс персистентный;
- при $1,5 < D < 2$ процесс — антиперсистентный.

В этих случаях процесс немарковский, хаотичный, для обработки результатов измерения должен быть применен специальный математический аппарат [3].

Ранее, в работах [3] и [5] фрактальный анализ был успешно использован для анализа динамики частоты излучения лазера и напряжения электрической сети соответственно.

Энтропия Шеннона H — ключевое понятие теории информации. Для ДП X_i с плотностью распределения вероятности $p(X_i)$ определяется как:

$$H_i = -p(X_i) \ln p(X_i). \quad (3)$$

Энтропия Шеннона используется в информационной теории измерения для оценки разброса результатов измерения. Если ряд значений X_i ограничен, то энтропия максимальна для равномерного закона распределения:

$$(H \rightarrow \ln(X_i^{\max} - X_i^{\min})) \\ \text{при } p(X_i) \rightarrow 1 / (X_i^{\max} - X_i^{\min}))$$

и минимальна при нормальном законе распределения, при отсутствии неопределённости состояний:

$$p(X_i) \rightarrow 1 \quad H \rightarrow 0.$$

Энтропия Шеннона характеризует и степень упорядоченности динамики НДС. Энтропия хаотичной и случайной динамики выше энтропии детерминированной динамики. Изменение её значения свидетельствует об изменении динамики или интервала значений ДП. Энтропия может быть применена для оценки отклонения системы от состояния равновесия.

В работе [3] автором было получено выражение связи энтропии Шеннона (3) и фрактальной размерности (2) временного ряда результатов измерений $\{x_i\}$ в виде:

$$H = \ln[2kR(n/2)^{D-2}].$$

где k — энтропийный коэффициент.

Согласно этому выражению энтропия Шеннона результатов измерения ДП НДС зависит от функции распределения, значения размаха Хёрста, количества измерений ДП и фрактальной размерности временного ряда:

$$H = H(k, R, n, D).$$

В значение энтропии вносят вклад качество измерительного эксперимента и разнообразие значений ДП НДС. При этом, вклад свойств НДС является определяющим. Согласно полученному выражению энтропия принимает максимальное значение в случае нормального распределения результатов измерения, при этом $k = 2,06$; энтропия принимает минимальное значение в случае

арксинусоидального распределения, при этом $k = 1,11$.

На значение энтропии влияет число состояний ДП НДС n . Чем большее количество состояний может принимать ДП НДС, тем выше значение энтропии. Если все состояния ДП равновероятны, то энтропия монотонно растёт с увеличением числа n .

Связь фрактальной размерности D со значением энтропии Шеннона выглядит следующим образом: энтропия максимальна для случая $D=2$, при этом ДП НДС ведет себя регулярным образом, но разброс измеряемых значений очень велик; энтропия минимальна для случая $D=1$, при этом поведение ДП НДС строго детерминированное и может быть описано уравнением.

Таким образом, разнообразие состояний ДП НДС является основным фактором влияния на значение энтропии.

Временя предсказуемости T_{for} — отрезок времени, на котором возможно корректное прогнозирование динамики НДС, определяется как:

$$T_{for}(\lambda) \sim \frac{1}{\lambda_{\max}} \log \frac{1}{\varepsilon}, \quad (4)$$

где λ_{\max} — максимальный показатель Ляпунова; ε — расстояние между близкими траекториями фазового портрета НДС (рис. 1).

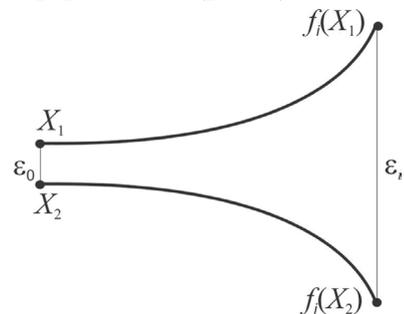


Рис. 1. Геометрия разбегания близких траекторий:

$\varepsilon_0, \varepsilon_i$ — расстояния между точками X_1, X_2 близких траекторий в моменты времени t_0 и t_i соответственно; $f_i(X_1), f_i(X_2)$ — отображения точек X_1, X_2 соответственно в момент времени t_i .

Показатели Ляпунова λ_i используют для исследования НДС в окрестности произвольной траектории. Если две близкие траектории $x_i(t)$ и $x_{i+1}(t)$ выбрать так что:

$$x_{i+1}(t) = x_i(t) + \xi(t), \text{ а } \xi(0) = \varepsilon, \varepsilon \rightarrow 0,$$

то функция вида:

$$\Xi[\xi(0)] = \lim_{t \rightarrow \infty} \frac{1}{t} \ln \left[\frac{\xi(t)}{\xi(0)} \right] \quad (5)$$

принимает конечный ряд значений показателей Ляпунова $\{\lambda_i\}, i = 1, 2, \dots, n$ [12].

В статье [9] авторы данной работы использовали интервалы значений ДП, фрактальную размерность (2) и энтропию Шеннона (3) при проведении

исследования человека, регулярно занимающегося физкультурой. В течение 21 недели с дискретностью $\Delta t=7$ дней выполнялись измерения параметров:

- нижняя часть бедра (X_1);
- верхняя часть бедра (X_2);
- бёдра (X_3);
- нижняя часть живота (X_4);
- талия (X_5);
- верхняя часть живота (X_6);
- масса (X_7).

Были получены семь временных рядов результатов измерения, рассчитаны интервальные значения ДП, значения фрактальной размерности и энтропии Шеннона (таблица).

Таблица

Результаты измерения ДП, уровень доверия $p=0,95$

ДП	$[X_i^{\min}; X_i^{\max}]$	D	H
X_1	[0,430; 0,450] м	1,47	0,891
X_2	[0,615; 0,660] м	1,2	1,117
X_3	[1,010; 1,070] м	1,2	1,117
X_4	[0,840; 0,910] м	1,02	2,117
X_5	[0,735; 0,760] м	1,2	1,417
X_6	[0,755; 0,780] м	1,00	1.235
X_7	[67; 70] кг	1,1	0,712

Анализ результатов измерений позволяет утверждать, что все ДП демонстрировали практически детерминированную динамику, что характерно для НДС с функцией самоорганизации. Значения фрактальной размерности всех ДП находятся в интервале $1 < D < 1,5$. При этом для ДП $X_2...X_7$ $D \rightarrow 1$, а для $D \rightarrow 1,5$. То есть, динамика X_1 близка к случайному процессу. Анализ значений энтропии Шеннона говорит о том, что наиболее переменными являются ДП X_4 ($H \approx 2$) и X_5 ($H \approx 1,4$), а наименее переменными ДП X_1 ($H \approx 0,9$) и X_7 ($H \approx 0,7$).

2. Модель измерения для системы с функцией самоорганизации

В рамках МИ измерение состояния НДС рассматривается как двухэтапный активный эксперимент. На первом этапе проводятся измерения ДП НДС в равновесном состоянии. На втором этапе система, под действием нормированного внешнего воздействия Q переходит в возбуждённое состояние, в котором также выполняются измерения значений ДП. После прекращения внешнего воздействия система за время T возвращается в равновесное состояние (рисунок). В случае нарушений в функционировании системы или эволюционных изменений, время возврата возрастает, или НДС вовсе не может вернуться в устойчивое состояние самостоятельно.

По результатам измерений ДП в равновесном состоянии определяются: n временных рядов

результатов измерений значений ДП $\{X_i(t)\}$ (n — количество ДП) при этом $X_i(t) \in [X_i^{Stab\min}; X_i^{Stab\max}]$; фрактальная размерность D_i^{Stab} (2); энтропия H_i^{Stab} (3); время прогноза динамики ДП $T_{for\ i}^{Stab}$ (4).

По результатам измерения ДП в возбуждённом состоянии определяются:

- n временных рядов результатов измерений значений ДП $\{X_i(t)\}$ при этом $X_i(t) \in [X^{\min}; X^{\max}]$;
- фрактальная размерность D_i ;
- энтропия H_i ;
- время прогноза $T_{for\ i}$;
- время T_i возврата значения ДП $X_i(t)$ из возбуждённого в равновесное состояние после прекращения нормированного воздействия Q (рис. 2).

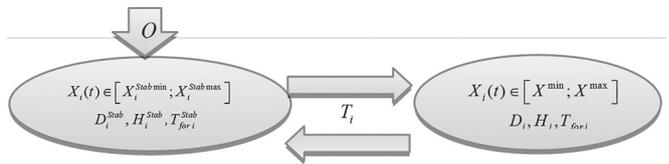


Рис. 2. Схема НДС с функцией самоорганизации

В МИ предложена энтропийная шкала оценки степени хаотичности динамики с использованием нормированной энтропии:

$$\|H\| = \sum_{i=1}^n \frac{-p(X_i) \ln p(X_i)}{\ln [X_i^{\max} - X_i^{\min}]} \quad (6)$$

Нормированная энтропия (6) принимает значения от 0 (регулярное протекание процессов) до 1 (максимальный хаос). Значения $\|H_{Stabi}\| = 0$ и $\|H_{Stabi}\| = 1$ являются граничными точками энтропийной шкалы. Значение энтропии (6) для равновесного состояния (норма энтропии) является третьей ключевой точкой шкалы, при этом $0 < \|H_{Stabi}\| < 1$.

В МИ норма энтропии H_{Stab} определена исходя из критерия «S-теоремы» Ю. Климонтовича. Являясь характеристикой состояния НДС, энтропия эволюционирующих систем меняется с течением времени вследствие внешних воздействий и эволюции (2). Результат её измерения необходимо сравнить с H_{Stab} :

$$\Delta H = H - H_{Stab} \quad (7)$$

Значение ΔH является показателем степени хаотичности.

На практике значение нормы энтропии H_{Stab} определяется как усредненное значение результатов измерения энтропии ряда схожих НДС. Однако, такой подход противоречит задаче индивидуального подхода к диагностике состояния систем. Исходя из этого, в МИ сформулированы принципы и правила определения нормы энтропии эволюционирующих НДС с функцией самоорганизации:

1) норма энтропии H_{Stab} меняется при эволюционных изменениях НДС, поэтому её значение должно периодически определяться заново;

2) по экспериментальным данным определяет значение энтропии H (3);

3) по формуле (6) определяются нормированные значения H и H_{Stab} ;

4) из выражения (7) определяется показатель степени хаотичности ΔH .

5) в соответствии с энтропийной шкалой выполняется анализ смещения динамики в сторону хаотичности или регулярности.

Также в МИ предложена временная шкала оценки состояния НДС. Шкала ограничена минимальным значением $T_i = 0$, максимального значения не имеет. Контрольной точкой шкалы является T_{Stabi} — время возврата в равновесное состояние «эталонной», идеально функционирующей системы.

В результате измерений, составляется цифровой портрет состояния НДС:

$$\left. \begin{array}{l} [X_{Stabi}^{\min}(t_0), X_{Stabi}^{\max}(t_0)], \|H_{Srab i}\|, T_{for Stabi} \\ [X_i^{\min}(t), X_i^{\max}(t)], \|H_i\|, T_{for i} \\ T_i, i = 1 \dots n. \end{array} \right\} \quad (8)$$

Оценка динамики состояния НДС (8) выполняется при помощи энтропийных и временных шкал:

$$\left. \begin{array}{l} 0 \dots \|H_{Stabi}\| \dots 1; \\ 0 \dots T_{Stabi} \dots \infty. \end{array} \right\} \quad (9)$$

В МИ предложен принцип определения нормированного воздействия Q (рисунок). Его значение должно быть таким, чтобы система, выйдя из устойчивого состояния, могла самостоятельно в него вернуться. То есть, значение Q не должно превышать некоего критического значения Q_{\max} , $Q \leq Q_{\max}$. Для определения Q_{\max} величину внешнего воздействия Q следует связать с показателем степени хаотичности ΔH (7). Для этого введём значение энтропии H_{Crit} , соответствующее возбуждённому состоянию, из которого система не может самостоятельно вернуться в равновесное состояние. Значение Q должно обеспечивать выполнение условия:

$$\Delta H < H_{Crit} - H_{Stab}. \quad (10)$$

Таким образом, разработанная МИ (8)-(10) учитывает изменение ДП НДС во времени (эволюцию системы), является динамической моделью. Она может быть использована для исследования, измерения и управления эволюционирующими НДС с функцией самоорганизации физической, биологической и иной природы. МИ позволяет оценить эволюционные изменения. Выбор в качестве количественной оценки состояния НДС времени возврата системы в состояние равновесия позволит создать надлежащее метрологическое обеспечение подобных исследований и прослеживаемость результатов измерений, опираясь на эталон времени.

Выводы

В работе представлена модель измерения динамических переменных нелинейных динамических систем с функцией самоорганизации. Модель построена на физико-математических основах теорий открытых динамических систем, синергетики и теории измерений в нелинейных динамических системах.

В рамках модели измерения проводятся измерения динамических переменных системы в равновесном и возбуждённом состояниях.

Результатом измерения являются интервалы значений динамических переменных, фрактальная размерность, энтропия Шеннона и время предсказания динамики.

Для оценки эволюционных процессов в модель введены энтропийные и временные шкалы.

В качестве количественной оценки состояния системы в модели используется время её возврата в состояние равновесия после прекращения внешнего нормированного воздействия.

Предложен энтропийный метод оценки нормированного внешнего воздействия.

Список литературы:

- [1] Nicolis, G., Prigogine, I. (1979). *Self-organization in nonequilibrium systems from dissipative structures to order through fluctuations*. Moscow: Mir.
- [2] Machekhin, Yu., Kurskoy, Yu. (2014). Fractal-entropy analysis of measurement results in nonlinear dynamical systems. *Measuring technique*. 57(6), 609-704.
- [3] Machekhin, Yu., Kurskoy, Yu. (2014). *Fundamentals of nonlinear metrology*. LAP. (in Russian).
- [4] Machekhin, Yu., Kurskoy, Yu. (2015). Compilation of the Shannon entropy measurement equation for nonlinear dynamical systems using interval analysis methods. *Pribory i metody izmereniy*. 6(2), 257-263. (in Russian).
- [5] Kurskoy, Yu. (2014). Fractal analysis of electrical measurements. *Energetika, energoberezhenie i energoaudit*. 119(1), 52-57. (in Russian).
- [6] Gnatenko, A., Machekhin, Yu., Kurskoy, Yu., Obozna, V. (2018). Providing mode locking in fiber ring lasers. *J. Nano-Electron. Phys.* 10(2), 2033-2041. (in Russian).
- [7] Machekhin, Yu., Kurskoy, Yu., Gnatenko, A., Tkachenko V. (2018). Nanolaser Superradiation In Information And Measuring Procedures. *Telecommunications and Radio Engineering*. 77(13), 1179-1186.
- [8] Machekhin, Yu., Kurskoy, Yu. (2015). The evaluation of cooled particles parameters by instruments of nonlinear metrology. *Metrologiya ta priladi*. 56(6), 50-53.
- [9] Machekhin, Yu., Kurskoy, Yu., Prisch, E. (2017). A human as a research object for metrology of nonlinear dynamic systems. *Metrologiya ta priladi*. 63(1), 63-66. (in Russian).
- [10] Haken, H.. (1984). *The science of structure: synergetics*. New York: Van Nostrand Reinhold.
- [11] Schröder, M. (2001). *Fractals, chaos, power laws, Regular and chaotic dynamics*. Izhevsk. (in Russian).
- [12] Loskutov, V. (2007). Charm of Chaos. *UFN*, 177(6), 989-1014. (in Russian).

Поступила в редколлегию 23.11.2018



Ю.В. Жмаєва, Л.Е. Чала

ХНУРЕ, м. Харків, Україна, larysa.chala@nure.ua

Досліджено сучасні методи оцінювання складності проектування програмних продуктів. Розглянуто процедури визначення меж, запитів та транзакцій взаємодії додатків. Запропонований метод дозволяє корегувати результати, отримані за допомогою методу функціональних точок, з використанням алгоритмів корекції оцінок складності проектування. Визначено перспективи використання методу.

СКЛАДНІСТЬ ПРОГРАМНИХ ДОДАТКІВ, МЕТОД ФУНКЦІОНАЛЬНИХ ТОЧОК, КОРЕКЦІЯ ОЦІНОК

Ю.В. Жмаєва, Л.Э. Чала. Гибридный метод оценивания сложности ИТ проектов. Исследованы современные методы оценки сложности проектирования программных продуктов. Рассмотрены процедуры определения границ, запросов и транзакций взаимодействия приложений. Предложенный метод позволяет корректировать результаты, полученные с помощью метода функциональных точек, с использованием алгоритмов коррекции оценок сложности проектирования. Определены перспективы использования метода.

СЛОЖНОСТЬ ПРОГРАММНЫХ ПРИЛОЖЕНИЙ, МЕТОД ФУНКЦИОНАЛЬНЫХ ТОЧЕК, КОРЕКЦИЯ ОЦЕНОК

Y.V. Zhmaeva, L.E. Chala. A hybrid method for estimating the complexity of IT projects. The modern methods of evaluating the complexity of designing software products are investigated. Considered procedures for determining the boundaries, queries and transactions of interaction of applications. The proposed method allows you to adjust the results obtained using the method of functional points, using algorithms for correcting the estimates of the complexity of the design. The possibilities of using the method are determined.

COMPLEXITY OF SOFTWARE APPLICATIONS, METHOD OF FUNCTIONAL POINTS, CORRECTION OF EVALUATING

Вступ

Визначення складності розробки програмного забезпечення є одним з найбільш критичних аспектів в управлінні ІТ проектами [1]. Точне визначення розміру програмного забезпечення допомагає менеджерам проєктів ефективно планувати і виконувати поставлені завдання. Дотримання правил, що диктуються існуючими методиками оцінок складності ІТ проєктів, є надзвичайно важливим. Розробникам програмного забезпечення доводиться виробляти значні обсяги вихідного продукту в умовах постійного розширення вимог користувача, внаслідок чого метод вимірювання розміру програми повинен бути максимально уніфікований. Поширеним методом вирішення цієї проблеми є функціональний точковий аналіз, який дозволяє здійснювати функціональну декомпозицію процесу проектування програмного забезпечення [2]. Основою даного підходу є класичний метод оцінювання програмного продукту з використанням ієрархічної структури робіт.

Застосування базового методу функціональних точок (МФТ) дозволяє вимірювати масштабність програмного забезпечення шляхом кількісної оцінки його функціональності, що надається користувачеві насамперед на момент реалізації логічного дизайну. Однак, безпосереднє використання МФТ має ряд недоліків, які не тільки вимагають додаткових витрат на проектування, але і можуть негативно позначитися на взаємодію проєктувальників та замовників.

У даній роботі досліджуються сучасні методи оцінювання процесу проектування програмних продуктів, а також пропонується модифікований гібридний підхід до оцінки складності ІТ проєктів на основі удосконалення методу функціональних точок за рахунок застосування механізмів ризик-менеджменту та прогнозування зусиль, пов'язаних з розробкою програмних систем.

1. Базовий метод функціональних точок

Згідно з базовим методом функціональних точок, трудомісткість програмного продукту розраховується з урахуванням функціональності системи, що розробляється, яка в свою чергу визначається на основі логічних груп взаємопов'язаних даних, підтримуваних і використовуваних додатком, а також простих процесів, пов'язаних з виведенням і введенням інформації. Безсумнівною перевагою методу є те, що можливість його застосування не залежить від технологічної платформи, на якій буде розроблятися продукт.

Для використання МФТ слід виконати послідовність наступних кроків: визначення типу виконуваної оцінки; оцінку області та меж розробки програмного продукту; підрахунок функціональних точок, пов'язаних з даними; підрахунок функціональних точок, пов'язаних з транзакціями; визначення сумарної кількості невіривняних функціональних точок (UFP); визначення значення фактору вирівнювання (FAV); розрахунок

кількості вирівняних функціональних точок (AFP – Adjusted Functional Point). Метод функціональних точок зазвичай передбачає оцінки наступних типів:

– «проект розробки» – проект, який є новим для розробника;

– «проект розвитку» – проект, розробка якого тривала певний час (оцінюється проект доопрацювання в функціональних точках: видалення, додавання, зміна та доопрацювання функціоналу);

– «продукт» – проект, якому надається оцінка незалежною стороною (оцінюються обсяги вже існуючого і розробленого продуктів).

Для оцінювання меж розробки програмного продукту мають бути визначені:

– внутрішні логічні файли (ILFs – Internal Logical File), що виділяються користувачем як логічно пов’язані групи даних або блоки керуючої інформації, які підтримуються всередині продукту;

– файли зовнішнього інтерфейсу (EIFs), що виділяються користувачем як логічно пов’язані групи даних або блоки керуючої інформації, на які посилається продукт, але які підтримуються поза продукту.

Прикладами логічних даних (інформаційних об’єктів) можуть бути клієнт, рахунок, тарифний план, послуга. Для підрахунку функціональних точок, пов’язаних з даними, спочатку визначається складність даних за такими показниками:

– DET (Data Element Type), тобто повторюване унікальне поле даних, наприклад, ім’я клієнта – 1 DET; адреса клієнта (індекс, країна, область, район, місто, вулиця, будинок, корпус, квартири) – 9 DET’s;

– RET (Record Element Type) – логічна група даних, наприклад, адреса, паспорт, телефонний номер.

Оцінка кількості невіривняних функціональних точок залежить від складності даних, яка визначається згідно з матрицею складності (таблиця 1).

Таблиця 1

Матриця складності даних

Тип	1-19 DET	20-50 DET	51+ DET
1 RET	Low	Low	Average
2-5 RET	Low	Average	High
6+ RET	Average	High	High

Оцінка даних в невіривняних функціональних точках розраховується для внутрішніх логічних файлів (ILFs) і для зовнішніх інтерфейсних файлів (EIFs) в залежності від їх складності.

Крім того здійснюється підрахунок функціональних точок, пов’язаних з транзакціями. Транзакція – це елементарний неподільний

замкнутий процес, який пре і переводить продукт з одного консистентного стану в інший. У МФТ розрізняються кілька типів транзакцій, представлених в таблиці 2.

Таблиця 2

Функції та типи транзакцій

Функція	Тип транзакції		
	EI	EO	EQ
Зміна поведінки системи	Основна	Додаткова	–
Підтримка одного або кількох файлів інтерфейсу ILF	Основна	Додаткова	–
Надання інформації клієнту/користувачу	Додаткова	Основна	Основна

Оскільки більшість комп’ютерних систем взаємодіють з іншими системами, то мають бути визначені межі поділу між розроблюваним проектом або додатком і зовнішніми додатками або домом користувача. Після встановлення таких меж, компоненти розроблюваного програмного забезпечення можуть бути класифіковані, ранжовані і підраховані.

2. Визначення меж, запитів та транзакцій взаємодії додатків

Межі можуть бути встановлені на ранньому етапі життєвого циклу програмного забезпечення. Якщо ж команда працює з додатком повторно, наприклад, додавання нового функціоналу до релізу, то межа проекту повинна бути аналогічною, навіть ідентичною попередній. Якщо додаток новий, межі інших додатків, які будуть взаємодіяти з розроблюваною системою, повинні бути переглянуті та перераховані [3].

Існує ряд технічних особливостей, які дозволяють точно визначити межі додатків. Межа для веб-додатку, яка наведена на рисунку 1, визначена аналогічним чином як для традиційних додатків. Для традиційних додатків кордон не малюється тільки навколо призначеного для користувача інтерфейсу або групи екранів, а навколо всього програми. Часто такі додатки є просто розширеннями для існуючих десктопних додатків.

Межі для клієнт-серверного додатку, приклад якого схематично наведено на рисунку 2, повинні бути орієнтовані як на клієнта, так і на сервер. Це пов’язано з тим, що ні клієнт, ні сервер не підтримують інтерфейс, тобто жоден компонент не представляє собою сам додаток.

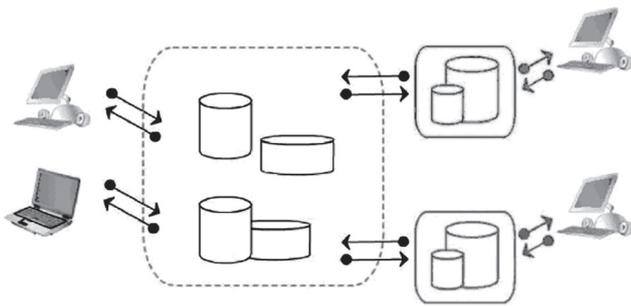


Рис. 1. Визначення меж веб-додатку

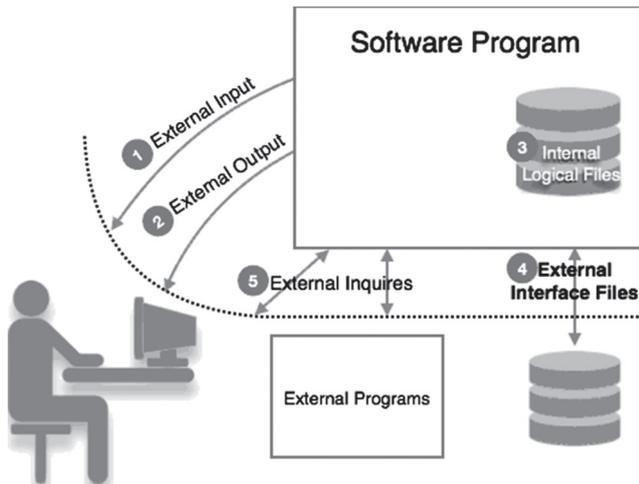


Рис. 2. Визначення меж клієнт-серверного додатку

Зовнішні вхідні транзакції EI (External Inputs) – це елементарний процес, в якому дані перетинають межу ззовні всередину. Ці дані можуть надходити з екрана введення даних або іншої програми. Вони можуть використовуватися для підтримки одного або декількох внутрішніх логічних файлів. Дані транзакції EI можуть бути пов'язані з керуючою або діловою інформацією. Для даних, пов'язаних з керуючою інформацією, не потрібно оновлювати внутрішній логічний файл. На рис. 3 наведено приклад простої транзакції EI, яка оновлює два файли ILF [4].

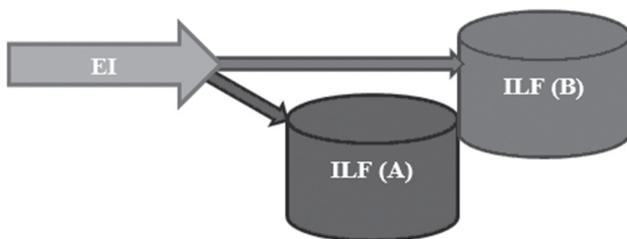


Рис. 3. Оновлення внутрішніх логічних файлів за допомогою зовнішніх вхідних транзакцій

Якщо зовнішні вхідні транзакції додають, змінюють і видаляють інформацію у внутрішньому логічному файлі, то така взаємодія представляється за допомогою трьох зовнішніх входів. Зовнішнім входам (особливо змінам і видаленням) може передувати зовнішній запит.

Як і всі компоненти, зовнішні вхідні транзакції можуть бути оцінені. Рейтинг вхідних транзакцій заснований на підрахунку кількості неповторюваних унікальних полів даних (DET) і пов'язаних з ним типів файлів (FTR). У таблиці 3 наведено матрицю рівнів складності транзакцій (низький, середній або високий) і відповідних балів (3, 4 або 6).

Таблиця 3

Матриця складності зовнішніх вхідних транзакцій

Типи файлів (FTR)	Рівень складності (DET)		
	1-4	5-15	Більше 15
0 – 1	Низький (3)	Низький (3)	Середній (4)
2	Низький (3)	Середній (4)	Високий (6)
Більше 2	Середній (4)	Високий (6)	Високий (6)

Зовнішні вхідні транзакції можуть бути бізнес-даними, даними управління і даними на основі правил. Прикладами бізнес-даних можуть бути ім'я клієнта, адреса, телефон тощо. Прикладами керуючих даних є дані, які викликають транзакцію або змінюють поведінку програми. Кожен прапорець представляє елемент даних. Крім того, кнопки для сортування співробітників і визначення тимчасового формату являють собою один елемент даних. Керуюча інформація змінює стан (поведінку) додатків, а також визначає, як і коли дані будуть оброблятися [5].

Наведемо приклади унікальних наборів даних, які дозволяють відрізнити зовнішні вхідні транзакції від інших транзакцій: поля введення даних; обчислені значення; повідомлення про помилки; повідомлення про підтвердження; рекурсивні поля, які враховуються як один DET; клавіші дії (командні кнопки); функціональні клавіші, які враховуються як один DET.

Унікальний FTR (File Types Referenced) також допомагає відрізнити оцінювану зовнішню вхідну транзакцію від інших транзакцій. FTR повинен бути або внутрішнім логічним файлом, або файлом зовнішнього інтерфейсу. Кожен внутрішній логічний файл, підтримуваний зовнішньою вхідною транзакцією, вважається FTR. Наприклад, зовнішній вхід може оновлювати внутрішній логічний файл, але також повинен посилатися на «файл безпеки», щоб переконатися, що у користувача є відповідні рівні безпеки. Це приклад двох FTR.

Існує ряд технічних особливостей, які дозволяють точно визначити кількість зовнішніх вхідних транзакцій: кнопки, де кожен набір перемикачів вважається одним DET, так як за один раз можна вибрати тільки одне значення перемикача; списки, що випадають можуть бути зовнішнім запитом, але результатом запиту має бути унікальне поле даних для зовнішнього введення; чекбокси – кожен прапорець, який може бути перевірений одночасно, є

унікальним DET; кнопки, які можуть вважатися унікальним полем даних для пов'язаної транзакції. Наступні типи документації можуть використовуватися для підрахунку зовнішніх вхідних транзакцій: документація з описом бізнес цілей; діаграма потоку даних; специфікація.

Зовнішні вихідні транзакції EO (External Outputs) – це елементарний процес, в якому похідні дані проходять через межу зсередини назовні. Крім того, вони можуть служити для поновлення файлів ILF. Вихідні дані створюють звіти або вихідні файли, що відправляються в інші додатки. Ці звіти і файли створюються з одного або декількох внутрішніх логічних файлів і файлу зовнішнього інтерфейсу.

Похідні дані – це дані, які обробляються за межами прямого пошуку і редагування інформації з внутрішніх логічних файлів або файлів зовнішнього інтерфейсу. Похідні дані зазвичай є результатом алгоритмів або розрахунків. Вони виникають, коли один або кілька елементів даних об'єднані з формулою для генерації або отримання додаткового елемента даних. Ці похідні дані не відображаються ні в одному FTR (внутрішній логічний файл або файл зовнішнього інтерфейсу).

Алгоритм визначається як механічна процедура для виконання заданого обчислення або покрокового вирішення проблеми. Розрахунок визначається як рівняння, яке має один або кілька операторів. Оператор є математичною функцією (додавання, віднімання, множення і ділення). Транзакції між додатками слід позначати інтерфейсами. Розроблюваний додаток може мати тільки зовнішній висновок або зовнішній запит даних по відношенню до додатка.

Як і всі компоненти, транзакції EO можуть бути оцінені. Рейтинг їх оцінювання заснований на кількості елементів даних (DET) і пов'язаних з ними типів файлів (FTR). Оцінка заснована на загальній кількості унікальних (комбінованих унікальних вхідних і зовнішніх сторін) елементів даних (DET) і типів файлів (FTR). У таблиці 4 наведено матрицю рівнів складності транзакцій (низький, середній або високий) і відповідних балів (4, 5 або 7).

Таблиця 4

Матриця складності зовнішніх вихідних транзакцій

Типи файлів (FTR)	Рівень складності (DET)		
	1-5	6-19	Більше 19
Менше 2	Низький (4)	Низький (4)	Середній (5)
2-3	Низький (4)	Середній (5)	Високий (7)
Більше 3	Середній (5)	Високий (7)	Високий (7)

Зовнішні вихідні транзакції містять інформацію, яка виводить дані через межу зсередини

програми в зовнішній рівень взаємодії з користувачем. Деяка плутанина може виникнути через те, що EO має вхідну сторону. Вхідна сторона EO – це критерії пошуку та параметри. Мета інформації, що надходить ззовні додатку (вхідна сторона), полягає не в тому, щоб підтримувати файли ILF. Наприклад, на відміну від інших компонентів, EO майже завжди містить бізнес-дані. Базові дані про правила і «вихідні дані» на основі управління майже завжди вважаються зовнішніми запитами через те, що правила і типи керування не виводяться.

Попереджуючі повідомлення також вважаються транзакціями EO. Вони відрізняються від повідомлень про помилку. Даний вид повідомлень є елементарним процесом, в той час як повідомлення про помилку (або повідомлення підтвердження) є частиною елементарного процесу. Попереджуючі повідомлення є результатом обробки бізнес-логіки (наприклад, торговий додаток може повідомити брокера про те, що клієнт, який намагається розмістити замовлення, не має достатніх коштів на своєму рахунку).

Елементарний процес, пов'язаний із зовнішнім виходом, може оновлювати внутрішній логічний файл або файл зовнішнього інтерфейсу. Наприклад, елементарний процес, який проводить перевірку нарахування заробітної плати, може включати оновлення файлу, щоб встановити прапорець і вказати, що була проведена перевірка заробітної плати. Підтримується процес зміни даних (додавання, зміна та видалення) за допомогою елементарного процесу (через зовнішній вхід).

Зовнішні запити EQ (External Inquiries) – це елементарний процес з вхідними та вихідними компонентами, які призводять до вилучення даних з одного або декількох внутрішніх логічних файлів і файлів зовнішнього інтерфейсу. Процес введення не оновлює внутрішні логічні файли, а вихідна сторона не містить похідних даних. На рисунку 4 представлений приклад зовнішнього запиту з двома внутрішніми логічними файлами.

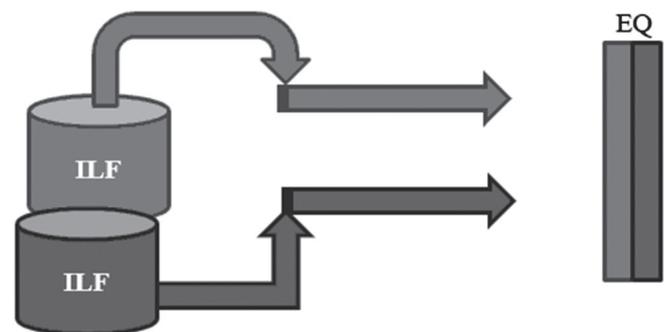


Рис. 4. Формування зовнішнього запиту

Як і всі компоненти, зовнішні запити EQ можуть бути оцінені. В цілому, EQ оцінюється (низький,

середній або високий), як EO, але запитами присвоюються значення типу EI. Оцінка заснована на загальній кількості унікальних елементів даних (DET) і типів файлів (FTR). Якщо один і той же FTR або DET використовується як на вхідний, так і на вихідній стороні, він підраховується тільки один раз. Матриця складності зовнішніх запитів наведена в таблиці 5.

Таблиця 5

Матриця складності зовнішніх запитів

Типи файлів (FTR)	Рівень складності (DET)		
	1-5	6-19	Більше 19
Менше 2	Низький (3)	Низький (3)	Середній (4)
2-3	Низький (3)	Середній (4)	Високий (6)
Більше 3	Середній (4)	Високий (6)	Високий (6)

Запити EQ можуть містити бізнес-дані, дані управління і дані, засновані на правилах. Прикладами бізнес-даних є імена клієнтів, адреси, номер телефону, список, що випадає (наприклад, список клієнтів по імені). Приклади зовнішніх запитів: натискання миші; значення пошуку; командні кнопки; скролінг; значення, що надійшли з внутрішнього логічного файлу або файлу зовнішнього інтерфейсу; зміна кольору або шрифту на екрані; рекурсивні поля.

Внутрішні логічні файли (ILF) – це ідентифікована користувачем група логічно пов'язаних даних, яка знаходиться на межі додатку і підтримується через зовнішні входи. Внутрішній логічний файл має деяку логічну структуру і зберігається в файлі. Хоч це і не обов'язково, але зазвичай файл ILF має як мінімум один зовнішній вивід або один зовнішній запит. Тобто, принаймні один зовнішній висновок або зовнішній запит повинні включати ILF як FTR. Інакше кажучи, інформація зберігається в ILF, тому її можна використовувати пізніше. Відзначимо, що транзакції EO або запити EQ можуть приходити з іншої програми.

Як і всі компоненти, файл ILF може бути оцінений. Рейтинг такої оцінки заснований на кількості елементів даних (DET) і типах записів (RET). У таблиці 6 зазначені рівні складності (низький, середній або високий) і відповідні бали (7, 10 або 15).

Таблиця 6

Матриця складності внутрішніх логічних файлів

Групи даних (RET)	Рівень складності (DET)		
	1-19	20-50	Більше 51
1	Низький (7)	Низький (7)	Середній (10)
2-5	Низький (7)	Середній (10)	Високий (15)
Більше 6	Середній (10)	Високий (15)	Високий (15)

Файли ILF можуть містити бізнес-дані, дані управління і дані, засновані на правилах. Тип

даних, що містяться в ILF, являє собою той же тип даних, який містить і підтримує EI. Зазвичай керуючі дані мають тільки одне входження в ILF. Скажімо, файл керуючих даних може містити тільки параметри або налаштування стану. Наприклад, частина бортової автомобільної системи містить тільки поточну інформацію (тиск масла, температуру двигуна тощо). Цей конкретний процес бортової системи дбає не про історичні дані, а тільки про поточні екземпляри. Коли статус змінюється, файл оновлюється поточною інформацією, а історична інформація відсутня. Бортова система може відстежувати історичні зміни в файлах діагностики, але це буде абсолютно окремий процес. Цей процес не використовується, щоб підтримувати роботу автомобіля, а тільки щоб допомогти механіку зрозуміти, що відбувається з двигуном.

Ідея застосування RET (реального часу) полягає в кількісному визначенні складних відносин між даними, які підтримуються в одному FTR. Прикладом систем RET є, зокрема, система телефонної комутації, де використовуються три типи даних: бізнес-дані, дані правил і дані управління. В цьому випадку, бізнес-дані – це фактичний виклик, дані правил – це те, як виклик повинен бути направлений через мережу, а дані управління – це те, як комутатори взаємодіють один з одним. Подібно керуючим файлам, в звичайних системах реального часу буде тільки одне входження у внутрішній логічний файл. Логічні групи даних – одна з найскладніших концепцій в аналізі функціональних точок. Більшість логічних груп залежать від відносин між батьками і дочірніми елементами.

Файли зовнішнього інтерфейсу (EIF) – це ідентифікована користувачем група логічно пов'язаних даних, яка використовується тільки для довідкових цілей. Дані цих файлів повністю знаходяться за межами додатку і підтримуються іншими додатками зовнішніх входів. Файл зовнішнього інтерфейсу є внутрішнім логічним файлом для іншої програми. Кожен EIF повинен мати принаймні один зовнішній файл, одну транзакцію, зовнішній вхід, зовнішній вихід або зовнішній запит.

Як і всі компоненти, складність EIF може бути оцінена. Оцінка заснована на кількості елементів даних (DET) і типах записів (RET). У таблиці 7 зазначені рівні складності EIF (низький, середній або високий) і відповідні бали (5, 7 або 10).

Таблиця 7

Матриця складності зовнішніх інтерфейсів

Групи даних (RET)	Рівень складності (DET)		
	1-19	20-50	1-19
1	Низький (5)	Низький (5)	Середній (7)
2-5	Низький (5)	Середній (7)	Високий (10)
Більше 6	Середній (7)	Високий (10)	Високий (10)

3. Визначення кількості невіривняних та вирівняних функціональних точок

Загальний обсяг продукту в невіривняних функціональних точках (UFP) визначається шляхом підсумовування для всіх інформаційних об'єктів і елементарних операцій:

$$MFP = \sum_{i=1}^{ILF} MFP_i + \sum_{i=1}^{EI} MFP_i + \sum_{i=1}^{EO} MFP_i + \sum_{i=1}^{EQ} MFP_i. \quad (1)$$

Коефіцієнт вирівнювання (VAF) враховує 14 загальних характеристик системи, які оцінюють загальну функціональність програми. Ступені впливу при цьому варіюються від нуля до п'яти: 0 – вплив відсутній; 1 – випадковий вплив; 2 – помірний вплив; 3 – середній вплив; 4 – значний вплив; 5 – сильний вплив.

Коефіцієнт вирівнювання обчислюється наступним чином:

$$VAF = 0.65 + 0.01 \sum_{i=1}^{14} C_i, \quad (2)$$

де C_i – ступінь впливу на кожну системну характеристику продукту; i – системні характеристики продукту.

Основним недоліком використання коефіцієнта вирівнювання є частковий суб'єктивізм, так як різні фахівці можуть по-різному оцінити той чи інший параметр. Для уніфікації процесу оцінки слід користуватися докладним описом кожного з факторів і відповідної оцінки.

До основних системних характеристик програмного продукту слід віднести: обмін даними; розподілена обробка даних; продуктивність; обмеження по апаратних ресурсах; ступінь транзакційного навантаження; інтенсивність взаємодії з користувачем; ергономічність; ступінь ефективності роботи кінцевих користувачів; інтенсивність зміни даних користувачами; складність обробки даних; можливість повторного використання; зручність інсталяції; зручність адміністрування; кількість портів (процесорів), що можуть бути залучені до обробки даних; гнучкість. Для прикладу у таблиці 8 наведено опис однієї з цих характеристик (ступеня впливу обміну даними на розроблювану програмну систему).

Подальша оцінка вирівняних функціональних точок (AFP) залежить від типу оцінки. Наприклад, початкова оцінка кількості вирівняних функціональних точок (AFP) для програмного додатка, що враховує тільки нову функціональність, яка реалізується в продукті, визначається так:

$$AFP = UFP * VAF. \quad (3)$$

Проект, який знаходиться в стадії розробки оцінюється в DFP (Development Functional Point) за такою формулою:

$$DFP = (UFP + CFP) * VAF, \quad (4)$$

де CFP – функціональні точки, підраховані для додаткової функціональності, яка буде потрібна при установці продукту (наприклад, при міграції даних).

Таблиця 8

Матриця складності зовнішніх інтерфейсів

Оцінка	Опис для визначення ступеня впливу
0	Продукт є автономним додатком
1	Додаток не є пакетним, але має віддалений ввід даних або віддалений друк
2	Додаток є пакетним
3	Додаток передбачає онлайн-збір даних для пакетного процесу або системи запитів
4	Логіка програмного додатку винесена на рівень інтерфейсів, що підтримують лише один тип протоколу зв'язку
5	Продукт обмінюється даними більш ніж по одному телекомунікаційному протоколу

Проект доопрацювання і вдосконалення продукту оцінюється за такою формулою:

$$EFP = (ADD + CHGA + CFP) * VAF + (DEL * VAFB), \quad (5)$$

де ADD – функціональні точки для доданої функціональності; $CHGA$ – функціональні точки для змінених функцій, розраховані після модифікації; VAF – величина фактора вирівнювання, розрахованого після завершення проекту; DEL – обсяг вилученої функціональності; $VAFB$ – величина фактора вирівнювання, розрахованого до початку проекту.

Сумарний вплив процедури вирівнювання знаходиться в межах $\pm 35\%$ відносно обсягу, розрахованого в UFP .

4. Корекція оцінки складності проекту з урахуванням ризиків

Наступним кроком запропонованого методу є визначення допустимого порогу ризиків, з якими розробник буде готовий взяти проект в розробку. Для кожного розробника існує граничне значення, в рамках якого менеджмент-команда зможе продуктивно управляти ризиками. Коригування МФТ-оцінки передбачає ідентифікацію позитивних і негативних ризиків на кожному з етапів проекту: аналіз і опрацювання вимог, проектування і дизайн, розробка, тестування і підтримка. Далі, кожен ризик аналізується за параметрами ймовірності його виникнення та його впливу на проект. Це дозволяє визначити пріоритет для обчислення діапазонів допустимих значень по кожному з етапів розробки програмного забезпечення.

Для того щоб почати пом'якшувати або усувати ризики потрібно прояснити які саме з них небезпечні. Це можна зрозуміти шляхом розрахунку

пріоритету ризиків. Щоб розрахувати пріоритет ризику, потрібно мати два основних параметри — коефіцієнт впливу ризику (Impact) і ймовірність виникнення ризику (Probability). Розрахунок пріоритету ризику здійснюється за такою формулою:

$$S = I * P, \quad (6)$$

де I — коефіцієнт впливу ризику; P — ймовірність виникнення ризику.

Коефіцієнти впливу ризику і ймовірності виникнення ризику можуть набувати значення в діапазоні [0.01; 1]. Отже, пріоритет ризику може приймати ті ж значення. Пріоритетність ризиків визначається наступними діапазонами:

- [0.01; 0.05] — низький пріоритет;
- [0.06; 0.17] — середній пріоритет;
- [0.18; 1.00] — високий пріоритет.

Найчастіше, розробка програмного забезпечення складається з декількох етапів, які займають ту, чи іншу частину робочого часу. Умовно, розробку програмного засобу можна розділити на наступні фази: аналіз і опрацювання вимог (20% часу реалізації проекту); проектування та дизайн (25% часу реалізації проекту); розробка (40% часу реалізації проекту); тестування (10% часу реалізації проекту); підтримка (5% часу реалізації проекту).

Згідно з методикою PERT та методом Монте-Карло, тривалість кожної операції має межі, які виходять з статистичного розподілу. PERT використовує 3 оцінки розрахунку часу для кожного завдання або проекту. Це означає, що тривалість кожного завдання має межу від оптимістичного (найкращого) до песимістичного (найгіршого), і середній показник можна розрахувати для кожної операції. Для вираження тривалості операції в PERT використовується апроксимація бета-розподілу. Згідно з цим можна зробити висновок, що різниця між оптимістичною і найбільш вірогідною оцінкою в 1,75 разів менше ніж між найбільш ймовірною і песимістичною. Дана ситуація пов'язана з тим, що в другій відрізок закладається ймовірність виникнення ризиків. У зазначеній різниці враховуються ризики з усіма стратегіями, крім стратегії усунення, так як превентивні заходи по обробці даного ризику повинні бути впроваджені в проект до його старту. Цей фактор доцільно брати до уваги для визначення меж діапазонів допустимих значень по кожній фазі на етапі корекції МФТ-оцінок складності проекту.

Для попередження виникнення максимально негативного сценарію проектування, слід визначити допустимий рівень ризиків, з яким ІТ — компанія зможе взяти проект в роботу, і скорегувати фінальну оцінку відповідним чином. Для цього пропонується використання принципу ALARP (As Low As Reasonably Practicable, що означає

мінімальний практично прийнятний ризик. Цей принцип дозволяє визначити поріг допустимості ризику шляхом зниження ризику аж до того моменту, коли прийняття додаткових заходів щодо зниження ризику, будучи технічно здійсненним, буде визнано непропорційно витратним. Ситуація, коли менеджери зможуть звести на нуль або зменшити ймовірність і ступінь впливу абсолютно всіх ризиків практично неможлива. Принцип ALARP вважає за доцільне прийняти наявність тих чи інших ризиків і не витрачати час і бюджет на зниження їх ймовірності і мінімізацію впливу ризиків. Згідно з принципом ALARP, всі ризики можна оцінити і розділити на три категорії: високопріоритетні (пріоритет 0.66 – 1.00); середньопріоритетні (пріоритет 0.33 – 0.65); фонові (пріоритет 0.01 – 0.32).

5. Тестування гібридного методу оцінки складності проекту

Запропонований метод було протестовано на прикладі оцінки складності проектування додатку «Hostel Issue Reporting», яке дозволяє відправляти запит з описом проблеми, що виникла з обладнанням в готелі. Опис функцій програми: користувач вибирає одну з категорій виниклих проблем на головному вікні; завантажується майстер «звіт про проблеми», де користувач вибирає адреса будівлі, номер поверху (необов'язково), службу, проблему, надає опис проблеми, вказує своє ім'я, номер телефону і відправляє запит; запис про нову проблему автоматично створюється в довідковій службі адміністратора; запис про підтвердження запиту відображається користувачеві зі створеним номером запиту, який може використовуватися пізніше для довідки. Також додаток містить прихований екран «Налаштування адміністратора», що використовується при першій установці. Для першого запуску додатка адміністратор повинен встановити адресу будівлі, в якому пристрій буде встановлено. Крім того, адміністратор може додавати додаткові категорії, крім тих, які встановлені за замовчуванням. Основне екранне вікно додатку містить коротку інструкцію про те, як використовувати додаток, і пропонує вибрати категорію проблем.

Згідно з запропонованим методом була сформована модель функціональних точок для програмного засобу, що реалізує інформаційну систему управління запитів про несправність обладнання, та визначені межі проектного програмного засобу. Очевидно, що цей засіб є локальним і обмін даними з іншими програмними засобами не передбачає.

Розроблюваний програмний засіб буде працювати з локальною базою даних, що складається з 7

таблиць – «User», «Category», «Request», «Problem», «Address», «e-mail», «RequestStatus». Таким чином, цей засіб буде мати 7 внутрішніх логічних файлів (ILF). Всі внутрішні логічні файли містять від 1 до 3 типів елементів записів (RET): символічний формат, числовий формат і формат дата / час. Число типів елементів даних внутрішніх логічних файлів DET – від 2 до 5. Таким чином, рівень складності внутрішніх логічних файлів низький [6].

У програмному засобі є 19 зовнішніх входів (EI), що характеризуються різними рівнями складності. Крім того, засіб має: 15 зовнішніх виходів, що відповідають різним повідомленням при додаванні або видаленні записів (12 мають низький рівень складності, 3 – середній); 9 зовнішніх запитів, що представляють собою поля введення для пошуку, виведення результату пошуку тощо. (7 мають низький рівень складності, 2 – високий). Основними характеристиками для даної системи є: обмін даними (оцінка 2); розподілена обробка даних (оцінка 3); (інтенсивність взаємодії з користувачем (оцінка 4); ергономічність (оцінка 5; гнучкість (оцінка 2). Інші характеристики не мають суттєвого значення для даного програмного продукту. Сумарне значення ваг всіх розглянутих характеристик дорівнює 14. Кількість рядків коду додатку *SLOC* була визначена за методом, заснованим на використанні так званого «мовного множника», який представляє собою середню кількість рядків вихідного коду конкретної алгоритмічної мови, що припадає на одну нормовану функціональну точку:

$$SLOC = AFP * LM, \quad (7)$$

де *LM* – мовний множник.

Згідно з розрахунками за запропонованим методом були отримані наступні значення показників складності розробки додатку «Hostel Issue Reporting»: *UFP* – 202; *VAF* – 0.81; *AFP* – 163.62; *SLOC* – 8671.86.

Фінальна оцінка складності розробки продукту була представлена в людино-годинах (МН), кожна одиниця яких має відповідний грошовий еквівалент:

$$MH = AFP * C, \quad (8)$$

де *C* – коефіцієнт, що визначає, скільки людино-годин еквівалентно одній функціональній точці (зазвичай приймається рівним 6).

Таким чином, згідно з розрахунками, для розробки додатку «Hostel Issue Reporting» *MH* = 982 люд.-год.

Використання розрахунків за методом Монте-Карло (для *MH* = 982 люд.-год.) дозволило зробити висновок, що аналіз і опрацювання вимог будуть виконані за 196 люд.-год., проектування і дизайн за 246 люд.-год., розробка за 393 люд.-год., тестування за 98 люд.-год., підтримка за 43 люд.-год. Було

проаналізовано вплив ризиків по кожному із значених етапів для визначення діапазонів допустимих значень. Наприклад, в рамках фази аналізу і опрацювання вимог існує ризик того, що вимоги не будуть з'ясовані в повній мірі, що призведе до незапланованого обсягу робіт. Імовірність виникнення даного ризику 40%, ступінь впливу на проект 80%, отже, пріоритет ризику дорівнює 32%.

З оглядом на вплив негативних ризиків (32%) в межах фази аналізу і опрацювання вимог, можемо зробити висновок, що різниця між оптимістичною і найбільш вірогідною оцінкою становитиме 18%. За результатами розрахунків були визначені граничні значення по кожній фазі розробки програмного продукту з урахуванням впливу негативних і позитивних ризиків.

Адаптуючи дану модель за принципом ALARP, можна зробити висновок, що фінальна оцінка складності проекту повинна бути завищена з урахуванням часу, необхідного на обробку ризиків і застосування відповідної стратегії. У базовому експерименті порогове значення у вигляді оцінки складності програмного продукту становило 982 людино-години. Збільшення цієї оцінки (за допомогою створення буфера на обробку високопріоритетних і середньопріоритетних ризиків) до 1055 людино-годин дозволить дати максимально об'єктивну, а головне конкурентоспроможну на ринку оцінку, що суттєво збільшить ймовірність успішного закінчення проекту.

Висновки

У статті був розглянутий модифікований гібридний підхід до оцінки складності ІТ проектів на основі удосконалення методу функціональних точок за рахунок застосування механізмів ризик-менеджменту та прогнозування зусиль, пов'язаних з розробкою програмних систем.

Оскільки більшість комп'ютерних систем взаємодіють з іншими системами, то мають бути визначені межі поділу між розроблюваним проектом або додатком і зовнішніми додатками або доменом користувача. Після встановлення таких меж, компоненти розроблюваного програмного забезпечення можуть бути класифіковані, ранжовані і підраховані.

Загальний обсяг продукту в невіривняних функціональних точках (*UFP*) визначається шляхом підсумовування для всіх інформаційних об'єктів і елементарних операцій. Це дозволяє визначити кількості невіривняних та вирівняних функціональних точок в програмному продукті, що розроблюється.

Наступним кроком запропонованого методу є визначення допустимого порогу ризиків, з якими

розробник буде готовий взяти проєкт в розробку. Коригування МФТ-оцінки передбачає ідентифікацію позитивних і негативних ризиків на кожному з етапів проєкту. Кожен ризик аналізується за параметрами ймовірності його виникнення та його впливу на проєкт. Це дозволяє визначити пріоритет для обчислення діапазонів допустимих значень по кожному з етапів розробки програмного забезпечення.

В статті наведено результати тестування запропонованого методу на прикладі оцінки складності проєктування програмного додатку «Hostel Issue Reporting», яке дозволяє відправляти запит з описом проблеми, що виникла з обладнанням в готелі. Результати тестування свідчать про те, що метод дозволяє формувати обґрунтовані рекомендації щодо оцінювання реальної складності ІТ проєктів.

Список літератури:

- [1] Филипенко О. М. Управління проєктами: навч. посібник / О. М. Филипенко, Т. С. Колеснік. – Харків : ХДУХТ, 2016. – 161 с
- [2] Herron D. E. Function Point Analysis: Measurement Practices for Successful Software Projects [Текст] / D. E. Herron, D. Garmus, – Т.: Addison-Wesley Professional, 2001. – 39 p.
- [3] Жмаєва Ю.В. Адаптивное прогнозирование оптимального количества ресурсов IT проекта по методологии Agile [Текст] / Ю.В. Жмаєва, Л.Э. Чалай, С.Г. Удовенко// АСУ и приборы автоматики. – 2015. – № 173. – С. 4–13.
- [4] Capers, J. The Economics of Software Quality [Текст] / J. Capers– N.: Pearson Education, 2011. – 62 p.
- [5] Cooper, B. The Lean Entrepreneur: How Visionaries Create Products, Innovate with New Ventures, and Disrupt Markets [Текст] / Cooper, B. – London: Wiley, 2016. – 133 p.
- [6] Жмаєва Ю.В. Оцінювання складності розробки програмного забезпечення з використанням модифікованого методу функціональних точок/ Ю.В. Жмаєва, Л.Е. Чала, С.Г. Удовенко // Праці Міжнародної науково-практичної конференції «Математичне моделювання процесів в економіці та управлінні проєктами і програмами (ММР-2018)». – Харків-Миколаїв, 2018. – С.58-61

Надійшла до редколегії 10.10.2018

УДК 519.62



Божко І.К.¹, Козел Н.Б.², Четвериков Г.Г.³

¹ Харківський національний університет радіоелектроніки,
Харків, Україна, ivan.bozhko@nure.ua

² Харківський національний університет радіоелектроніки,
Харків, Україна, natalia.kozel1@nure.ua

³ Харківський національний університет радіоелектроніки,
Харків, Україна, grigoriy.chetverykov@nure.ua

ІНСТРУМЕНТИ КВАНТОВИХ ОБЧИСЛЕНЬ

В даній роботі досліджується поточний стан існуючих інструментів квантових обчислень, розглядаються такі засоби як IBM Q Experience, ProjectQ, Rigetti Forest та особливу увагу приділяється мові програмування Q# як найбільш розвиненому інструменту квантових обчислень в даний час. Оскільки квантові обчислення сьогодні є однією з основних областей досліджень, створюються відповідні інструменти, які покликані спростити розробку квантових програм, з одного боку, і надати платформу для тестування і запуску їх, з іншого боку. Тому автори дослідили наявні в даний час інструменти і представили результати в даній роботі.

КВАНТОВІ ОБЧИСЛЕННЯ, КВАНТОВИЙ КОМП'ЮТЕР, IBM Q EXPERIENCE, RIGETTI FOREST, PROJECTQ, Q#, ІНСТРУМЕНТИ КВАНТОВИХ ОБЧИСЛЕНЬ, MICROSOFT QUANTUM DEVELOPMENT KIT

Ivan Bozhko, Natalia Kozel, Grygorij Chetverykov. Quantum computing tools. In this paper, the current state of existing quantum computing tools is investigated, such tools as IBM Q Experience, ProjectQ, Rigetti Forest are considered, and special attention is given to the programming language Q# as the most advanced quantum computing tool currently available. As quantum computing today is one of the main areas of research, relevant tools are created to simplify the development of quantum programs, on the one hand, and provide a platform for testing and launching them, on the other hand. Therefore, the authors examined existing tools and presented the results in this paper.

QUANTUM COMPUTING, QUANTUM COMPUTER, IBM Q EXPERIENCE, RIGETTI FOREST, PROJECTQ, Q#, QUANTUM COMPUTING TOOLS, MICROSOFT QUANTUM DEVELOPMENT KIT

Божко И.К., Козел Н.Б., Четвериков Г.Г. Инструменты квантовых вычислений. В данной работе исследуется текущее состояние существующих инструментов квантовых вычислений, рассматриваются такие средства как IBM Q Experience, ProjectQ, Rigetti Forest и особое внимание уделяется языку программирования Q# как наиболее развитому инструменту квантовых вычислений в настоящее время. Поскольку квантовые вычисления сегодня являются одной из основных областей исследований, создаются соответствующие инструменты, которые призваны упростить разработку квантовых программ, с одной стороны, и предоставить платформу для тестирования и запуска их, с другой стороны. Поэтому авторы исследовали имеющиеся в настоящее время инструменты и представили результаты в данной работе.

КВАНТОВЫЕ ВЫЧИСЛЕНИЯ, КВАНТОВЫЙ КОМПЬЮТЕР, IBM Q EXPERIENCE, RIGETTI FOREST, PROJECTQ, Q#, ІНСТРУМЕНТИ КВАНТОВИХ ОБЧИСЛЕНЬ, MICROSOFT QUANTUM DEVELOPMENT KIT

Вступ

Квантові обчислення являють собою альтернативний класичному підхід до виконання обчислень на комп'ютері і потребують спеціальний пристрій для їх виконання – квантовий комп'ютер.

Ідея цих обчислень не є новою, незважаючи на її популярність сьогодні, і була висунута радянським математиком Ю.І. Манін ще в 1980 році у своїй монографії «Обчисленне і необчисленне». Проте найбільший інтерес до цього виду обчислень виник тільки у 1982 році, після того, як американський фізик-теоретик Річард Фейнман помітив, що не всі квантово-механічні операції можуть бути точно змодельованими на класичному комп'ютері і для більш ефективного виконання потребують окремого пристрою здатного виконувати квантові операції [1].

У 1994 році проблема квантових обчислень стала ще більш актуальною, коли Пітер Шор, американський математик, запропонував алгоритм, що дозволяє розкласти N-значні числа на прості множники з поліноміальною складністю. Ця задача є значно більш складною для класичних комп'ютерів і не вирішується на них протягом задовільного часу.

Оскільки ця задача є основою багатьох популярних асиметричних криптографічних алгоритмів (наприклад, RSA) [2], розвиток квантових комп'ютерів може впливати на безпеку обміну даними в мережі, а після появи реального прототипу квантового комп'ютера проблема може набути глобального характеру.

За останні кілька десятиліть впровадження квантового комп'ютера досягло значного прогресу.

Існуючі дослідження з даної тематики розглядаються у літературі [2-10].

Точність одно- і двокубітних квантових венти-лів перевищує 99% і досягає межі відмовостійкості квантових обчислень [3]. Сьогодні кількість кубітів у сучасних квантових обчислювальних процесорах зростає і перевищує 20, за даними останнього дослідження IBM [4].

Дана проблема важлива не тільки для інформаційної безпеки, але квантові комп'ютери також мають потенціал у наукових дослідженнях, таких як, наприклад, моделювання фізичних та хімічних процесів [5].

З огляду на сучасні дослідження зрозуміло, що квантові комп'ютери можуть стати доступними найближчим часом, а для того, щоб дозволити науковцям і інженерам використовувати їх, створюються спеціальні інструменти, що спрощують написання квантових програм. Деякі з таких інструментів було досліджено і представлено в даній роботі.

1. Огляд існуючих інструментів квантових обчислень

В останні роки зі збільшенням кількості досліджень у галузі квантових обчислень з'явилися нові інструменти та емулятори квантових комп'ютерів, що дозволяє їх користувачам спробувати працювати з квантовими програмами. Авторами було обрано чотири різні популярні інструменти для вивчення та порівняння їх можливостей.

1.1. IBM Q Experience

IBM Q Experience – це онлайн-платформа для вивчення квантових обчислень та забезпечення доступу до емуляторів і реальних прототипів квантових процесорів для дослідницьких цілей. Вона включає в себе набір навчальних матеріалів для користувачів, Quantum Composer (інтерфейс користувача для створення квантових схем), емулятор для тестування, а також прототипи квантових процесорів [6].

У ньому представлені серверні пристрої, що включають в себе два процесори з 5 надпровідними кубітами (ibmqx2 і ibmqx4), один 16-кубітний процесор (ibmqx5) і один 20-кубітний процесор (QSI 1). Нещодавно IBM оголосила про те, що вони успішно побудували і випробували 20-кубітний і 50-кубітний машини. Квантова хмарна служба IBM забезпечує високу точність операцій і вимірювань квантових воріт. Таким чином, після запуску IBM Q, багато розробників перевірили його і виконали квантові обчислювальні експерименти у хмарі [4].

1.2. Microsoft Quantum Development Kit

У 2017 році Microsoft випустила попередню версію Quantum Development Kit, яка є платформою для створення квантових програм, включаючи нову

мову квантового програмування Q#, інтеграцію з середовищем розробки Visual Studio, платформу .NET, емулятори, що працюють як з локальною системою, так і з потужною хмарною платформою Azure, а також бібліотеками та зразками коду, які можна використовувати як конструктивні блоки [7].

Незважаючи на існування інших квантових обчислювальних засобів, головною перевагою Microsoft Quantum Development Kit є Q#. Це не єдина мова квантових обчислень (існують інші нещодавно представлені мови Quipper [8], LIQUi |> [9], ProjectQ [10] і т.д.), але відрізняється від інших у підході до написання програм.

На відміну від Quipper (вбудований в Haskell), LIQUi |> (вбудований в F #) і ProjectQ (вбудований в Python), Q# не орієнтований на маніпулювання квантовими схемами, а є мовою визначення алгоритму і, природно, представляє алгоритми без маніпуляцій з схемами. Таким чином, Q# дозволяє писати квантові алгоритми, не обробляючи низькорівневу логіку, тому її легше використовувати без вивчення деталей низького рівня. Крім того, оскільки вона має інтеграцію із платформою .NET, вона легко поєднується з класичною програмою (наприклад, написаною на C#) [7].

Інша відмінність Q# полягає в тому, що вона має власну систему типів, розроблену спеціально для квантових обчислень, що дає можливість використовувати мову Q# самостійно.

Підсумовуючи, переваги даного інструменту - низька складність використання; Платформа .NET, Visual Studio, інтеграція Azure; орієнтація на визначення алгоритмів замість деталей низького рівня і власної системи типу, що дозволяє використовувати Q# без будь-якої іншої мови.

З наведеного вище тексту маємо також кілька недоліків. По-перше, Q# не забезпечує простої можливості написання низькорівневої логіки, наприклад, маніпулювання квантовими схемами. По-друге, на даний момент вона інтегрована тільки з інструментами Microsoft.

1.3. ProjectQ

ProjectQ – це програмне забезпечення з відкритим вихідним кодом для квантових обчислень, що використовується в Python. Це дозволяє користувачам реалізовувати свої квантові програми в Python, використовуючи потужний і інтуїтивно зрозумілий синтаксис. ProjectQ потім може транслювати ці програми на будь-який сервер: емулятор, який працює на класичному комп'ютері, або квантовий комп'ютер (наприклад, за допомогою IBM Q Experience). Інші апаратні платформи наразі не підтримуються [9].

1.4. Rigetti Forest

Пакет Rigetti Forest складається з квантової мови на основі інструкцій Quil, відкритої

бібліотеки Python для побудови програм Quil під назвою pyQuil, квантової бібліотеки Grove і середовища моделювання QVM (Quantum Virtual Machine). pyQuil і Grove – це програми з відкритим кодом, доступні на Github. Користувачі можуть розробляти свої програми за допомогою pyQuil і Grove на своєму комп'ютері, а потім передавати їх у QVM для моделювання через веб-портал, доступний зареєстрованим користувачам [10].

На додаток до цих інструментів, є інші, такі як Cirq, Quirk, QuTiP, але вони менш потужні, ніж описані вище.

Як видно з опису існуючих рішень, більшість з них доповнюють існуючі мови програмування (зокрема, Python), але, враховуючи нову парадигму обчислень, це може викликати труднощі в алгоритмах програмування з використанням цих інструментів, тому гнучке і потужне рішення – це окрема мова програмування, яку зараз пропонує тільки Microsoft.

Також необхідно виділити рішення Intel з існуючим експериментальним квантовим комп'ютером, але це вимагає виконання запитів до окремих кубітів через API, що ускладнює його використання.

2. Мова Q#

Як обговорювалося вище, однією з частин Microsoft Quantum Development Kit є мова Q#, спеціально розроблена для квантових обчислень. Цей інструмент є основним, який досліджується в даній роботі. З точки зору розробки програмного забезпечення, це рішення є найбільш цікавим, оскільки дозволяє абстрагуватися від парадигми класичних обчислень та класичних мов програмування, та описувати квантовий алгоритм, використовуючи спеціальний синтаксис.

Розглянемо мову Q# більш докладно.

2.1. Модель обчислень

Згідно з опублікованими Microsoft даними [6], використовується модель для квантових обчислень полягає в тому, щоб розглядати квантовий комп'ютер як співпроцесор, подібний до GPU, FPGA та інших допоміжних процесорів. Первинною є логіка класичної програми, що виконує код звичайною мовою програмування на класичному комп'ютері (хост). Коли це доречно і необхідно, хост-програма може викликати підпрограму, яка працює на додатковому процесорі. Після завершення підпрограми хост-програма отримує доступ до результатів підпрограми.

У цій моделі існують три рівні обчислення:

- Класичні обчислення, які зчитують вхідні дані, викликає квантову програму, ініціює квантові обчислення, обробляє результати і представляє результати користувачеві.

- Квантові обчислення, які відбуваються безпосередньо в квантовому пристрої та реалізують квантовий алгоритм.

- Класичні обчислення, які необхідні квантовому алгоритму під час його виконання.

Не існує внутрішньої вимоги, щоб ці три рівні були написані однією мовою. Дійсно, квантові обчислення мають дещо інші структури управління і потреби в управлінні ресурсами, ніж класичні обчислення, тому використання спеціальної мови програмування дозволяє виразити більш моделі в квантових алгоритмах зручніше.

Збереження класичних обчислень окремо означає, що мова квантового програмування може бути дуже обмеженою. Ці обмеження дозволяють краще оптимізувати або прискорити виконання квантового алгоритму.

Q# (Q-sharp) – це специфічна для галузі мова програмування, яка використовується для описання квантових алгоритмів. Вона повинна використовуватися для написання підпрограм, які виконуються на додатковому квантовому процесорі, під контролем класичної хост-програми та комп'ютера.

Q# забезпечує невеликий набір примітивних типів, а також два засоби (масиви і кортежі) для створення нових, структурованих типів. Він підтримує базову процедурну модель для написання програм, з циклами і операторами if / then. Конструкції верхнього рівня у Q# – це типи, операції та функції, визначені користувачем.

2.2. Система типів

Мова Q# забезпечує невеликий набір примітивних типів, а також два методи (масиви і виправлення) для створення нових типів даних (це означає, що мова має трохи більше обмежень, ніж класичні мови, які були зроблені для цілей оптимізації). Взагалі, мова зазвичай підтримує модель процесуального умовного програмування (if -this) і цикли.

Розглянемо примітивні типи даних, з яких складаються інші типи:

- int – 64-бітне число зі знаком;
- double – число з плаваючою точкою подвійної точності;

- bool – логічне значення, true або false;

- qubit являє собою кубіт або квантовий біт.

Вони непрозорі для користувача; єдина можлива операція з ними, крім передачі їх іншій операції, є перевіркою на ідентичність (рівність). Зрештою, дії на Qubits реалізуються викликом операцій у стандартній бібліотеці Q#;

- Тип Pauli являє собою елемент групи Паулі з одним кубітом. Цей тип використовується для позначення базової операції для ротацій і для визначення базису вимірювання. Цей тип є дискримінованим об'єднанням з чотирма

можливими значеннями: PauliI, PauliX, PauliY і PauliZ;

- Result відображає результат вимірювання.

Цей тип є дискримінованим об'єднанням з двома можливими значеннями: One і Zero. Zero вказує на те, що було виміряне значення +1, One вказує на значення -1;

- Range являє собою послідовність цілих чисел;
- String — це послідовність символів Unicode,

що непрозора для користувача після створення. Цей тип використовується для повідомлення повідомлень класичній хост-програмі.

Слід також зазначити, що, маючи описану вище систему типів, існує набір зарезервованих ключових слів: true, false, PauliI, PauliX, PauliY, PauliZ, Zero і One.

За межами примітивних типів існують і інші типи, які ми далі розглянемо більш детально.

Маючи будь-який допустимий тип T у Q#, існує тип, який представляє масив значень типу T. Цей тип масиву представлений як T[], наприклад, Qubit[] або Int [][].

У другому прикладі необхідно зауважити, що тип Int[][] являє собою потенційно нерівний масив масивів, а не прямокутний двовимірний масив. Q# не має підтримки прямокутних багатовимірних масивів.

Враховуючи будь-які допустимі типи Q# T1, T2, T3 і т.д., існує тип, який представляє кортеж значень типів T1, T2, T3 і т.д., відповідно. Цей тип кортежу представляють у вигляді (T1, T2, T3,...). До кортежу може бути додано будь-яке число типів. Порожній кортеж, (), еквівалентний одиниці у F#.

Можна створювати масиви кортежів, кортежі масивів, кортежі підкортежів і т.д.

Екземпляри типу Tuple є незмінними. Q# не надає механізму для зміни вмісту кортежу, що був створений.

Можна також створити одинарний кортеж, (T1), такий як (5) або ([1; 2; 3]), проте Q# розглядає одноелементний кортеж як повністю еквівалентний значенню вкладеного типу. Тобто, між 5 і (5), або між 5 і ((5)), або між (5, (6)) і (5, 6) немає різниці.

Ця еквівалентність може бути використана для всіх цілей, включаючи призначення та вирази. Це так само справедливо для записів (5) + 3 або 5 + 3, і обидва вирази будуть рівні 8. Цю властивість називають еквівалентністю одинарного кортежу.

Також існує можливість створення користувачьких типів. В мові Q# можна визначити новий тип на основі стандартного. Будь-який існуючий тип може використовуватися як основа для типу користувача.

Визначені користувачем типи можуть бути використані в будь-якому місці будь-якого іншого

типу. Зокрема, можна визначити масив визначеного користувачем типу і включити даний тип як елемент кортежу.

Створення структур рекурсивного типу неможливо. Тобто, тип, який визначає визначений користувачем тип, не може бути типом кортежу, який включає елемент визначеного типу. Більш загально кажучи, визначені користувачем типи не можуть мати циклічних залежностей один від одного.

Екземпляри визначених користувачем типів можуть змінюватись так само, як екземпляри базового типу, визначеного користувачем. Зокрема, екземпляри визначених користувачем типів на основі кортежів є незмінними; екземпляри визначених користувачем типів на основі масивів є потенційно змінюваними.

Фактично користувальницький тип є підтипом базового типу. Таким чином, значення визначеного користувачем типу може використовуватися в будь-якому місці, де очікується значення базового типу. Це застосовується рекурсивно.

Наприклад, припустимо, що тип IntPair є визначеним користувачем з базовим типом (Int, Int), а тип IntPair2 — це визначений користувачем тип з базовим типом IntPair. Значення типу IntPair2 може використовуватися в будь-якому місці, коли очікується значення типу IntPair2, IntPair або (Int, Int). Значення типу IntPair може використовуватися в будь-якому місці, де очікується значення типу IntPair або (Int, Int).

Різні користувачькі типи, що базуються на одному і тому ж базовому типі, розглядаються як різні і не пов'язані між собою. У попередньому прикладі, якщо IntPair3 — також визначений користувачем тип з базовим типом (Int, Int), то IntPair і IntPair3 не пов'язані між собою, і значення одного не може використовуватися там, де очікується значення іншого.

Операція Q# — квантова підпрограма. Тобто, це підпрограма, що викликається, яка містить квантові операції.

Функція Q# є класичною підпрограмою, що використовується в квантовому алгоритмі, вона може містити класичний код, але не квантові операції. Функції не можуть виділяти або захоплювати кубіти, а також не можуть викликати операції. Однак можна передавати їм операції або кубіти для обробки.

Разом операції та функції називаються callable.

Всі callable в Q# розглядаються як такі, що приймають одне значення у вхідних даних і повертають одне значення у вихідних. І вхідні, і вихідні значення можуть бути кортежами. Callable, які не мають результату, повертають порожній кортеж, (); callable, які не мають вхідних даних, приймають порожній кортеж як вхідний.

Основна сигнатура для будь-якого callable виглядає наступним чином: (Tinput => Tresult) або (Tinput -> Tresult), де обидва Tinput і Tresult є іменами типу. Перша форма використовується для операцій; друга – для функцій. Наприклад, ((Qubit, Pauli) => Result) представляє запис для можливої операції вимірювання в один кубіт.

Типи функцій повністю задаються їх підписом. Наприклад, функція, яка обчислює синус кута, має тип (Double -> Double).

Операції (але не функції) дозволяють застосувати один або більше функторів. Функтори – це мета-операції, які генерують варіант базової операції. Типи операцій визначаються їх сигнатурою і списком функторів, які вони підтримують. Наприклад, операція PauliX має тип (Qubit => ()): Adjoint, Controlled). Тип операції, який не підтримує жодних функторів, визначається тільки його підписом, без кінцевих значень.

Сигнатури callable можуть містити параметри типу. Типові параметри позначаються символом, який префіксується одинарною лапкою; наприклад, 'A - це вірний параметр типу. Функції і операції, параметризовані за типом, подібні до загальних функцій на багатьох мовах програмування, але Q# не забезпечує повноцінної можливості розробляти загальні типи та функції.

Параметр типу може з'являтися більше одного разу в одній сигнатурі. Наприклад, функція, яка застосовує іншу функцію до кожного елементу масиву і повертає зібрані результати, має підпис (('A[], 'A-> A) ->'A[]). Аналогічно, функція, яка повертає композицію з двох операцій, може мати підпис (('A =>' B), ('B =>' C)) -> ('A =>' C)).

При виклику параметризованого за типом callable всі аргументи, що мають один і той же параметр типу, повинні бути одного типу або бути сумісними з тим же типом.

Q# не забезпечує механізмів обмеження можливих типів, які можуть бути замінені параметром типу. Таким чином, параметри типу є в першу чергу корисними для функцій на масивах і для створення callable.

Робота з додатковими підтримуваними функторами може бути використана в будь-якому місці для роботи з меншою кількістю функторів, але в цьому випадку однакова сигнатура. Наприклад, операція типу (Qubit => ()): Adjoint) може використовуватися в будь-якому місці, де очікується операція типу (Qubit => ()).

Q# є ковариантним по відношенню до типів, що повертаються з callable: callable, який повертає тип 'A, сумісний з callable з тими же вхідними і вихідними типами, з яким сумісний 'A.

Q# є контриваріантним по відношенню до типів вхідних даних: callable, що приймає тип 'A як вхідний, сумісний з callable з тими же вхідними і вихідними типами, що сумісні з 'A.

Функтор у Q# є фабрикою, яка визначає нову операцію з іншої операції. Функтори мають доступ до реалізації базової операції при визначенні реалізації нової операції, тому функтори можуть виконувати більш складні функції, ніж традиційні функції вищого рівня.

Функтор застосовується до операції, повертаючи нову операцію. Наприклад, операція, що виникає в результаті застосування функтора Adjoint до операції Y, записується як (Adjoint Y). Нова операція може бути викликана як будь-яка інша операція. Таким чином, (Adjoint Y) (q1) застосовує приєднаний функтор до операції Y для генерації нової операції і застосовує цю нову операцію до q1. Аналогічно, (Controlled X) (функтор, операція).

Двома стандартними функторами в Q# є Adjoint і Controlled.

У квантових обчисленнях Adjoint операції є складною сполученою транспозицією операції. Для операцій, які реалізують унітарний оператор, приєднана частина є зворотною операцією. Для простої операції, яка просто викликає послідовність інших унітарних операцій на множині кубітів, Adjoint може бути обчислено шляхом застосування сполучених підоперацій на тих же кубітах, у зворотній послідовності.

Якщо маємо вираз з операцією, може бути сформовано новий вираз за допомогою Adjoint-функтора, причому вираз базової операції укладається в дужки (and). Нова операція має таку саму сигнатуру і тип, як і базова операція. Зокрема, нова операція також дозволяє використання Adjoint і Controlled тоді і тільки тоді, коли базова операція також дозволяє.

Наприклад, (Adjoint QFT) позначає сумісну операцію QFT.

Контрольована версія операції є новою операцією, яка ефективно застосовує базову операцію тільки у тому разі, коли всі керуючі кубіти знаходяться в заданому стані. Якщо керуючі кубіти знаходяться в суперпозиції, то базова операція застосовується когерентно до відповідної частини суперпозиції. Таким чином, контрольовані операції часто використовуються для генерації запутаності.

У Q# контрольовані версії завжди приймають масив керуючих кубітів, а заданий стан завжди для всіх керуючих кубітів, що знаходяться в обчислювальному (PauliZ) стані, рівному One $|1\rangle|1\rangle$. Контроль, заснований на інших станах, може бути досягнуто шляхом застосування відповідних кліффордових операцій до кубітів перед даною операцією, а потім застосуванням зворотних операцій після неї. Наприклад, застосування операції X до керуючого кубіту до і після контрольованої операції призведе до того, що операція буде контролюватися в стані Zero $(|0\rangle|0\rangle)$ для цього кубіту; застосовуючи операцію H контроль буде виконуватися у стані PauliX Zero $|+\rangle = (|0\rangle + |1\rangle) / \sqrt{2}$; а не в стані PauliZ Zero .

Якщо є вираз з операцією, можна сформулювати новий вираз з операцією за допомогою функтора Controlled, причому вираз базової операції укладено в дужки (and). Сигнатура нової операції базується на сигнатурі базової. Тип результату є таким самим, як у базової операції, але вхідний тип – це подвійний кортеж, що складається з масива кубітів, що контролюються, та аргумента базової операції. Якщо базова операція не приймала аргументів, то вхідний тип контрольованої версії – це просто масив керуючих кубітів. Нова операція дозволяє використання Controlled і Adjoint тоді і тільки тоді, коли дозволяє базова операція.

Якщо основна функція приймала лише один аргумент, то буде використовуватися еквівалентність одиночного кортежу. Наприклад, Controlled(X) є контрольованою версією операції X. X має тип (Qubit => ()): Adjoint, Controlled), тому Controlled (X) має тип ((Qubit [], (Qubit)) => ()): Adjoint, Controlled); через еквівалентності одинарного кортежу це те ж саме ((Qubit [], Qubit) => ()): Adjoint, Controlled).

Аналогічно, Controlled (Rz) є контрольованою версією операції Rz. Rz має тип ((Double, Qubit) => ()): Adjoint, Controlled), тому Controlled (Rz) має тип ((Qubit [], (Double, Qubit)) => ()): Adjoint, Controlled).

В якості іншого прикладу, CNOT(control, target) може бути реалізована як (Controlled(X)) ([control], target). Якщо ціль повинна контролюватися двома керуючими кубітами (CCNOT), можна використовувати оператор (Controlled (X)) ([control1; control2], target).

Висновки

У рамках даної роботи досліджено основні інструменти квантових обчислень. Основна увага приділяється Microsoft Quantum Development Kit і Q# як найбільш семантично відповідному інструменту для квантових обчислень, так що вони були головною областю дослідження даної статті.

Сьогодні існують різні інструменти для квантових обчислень, окрім Q#, таких як IBM Q Experience, Rigetti Forest, ProjectQ та інші. Вони також заслуговують окремого дослідження, особливо враховуючи факт швидкого розвитку технологій і частих змін в даній галузі.

На сьогодні квантові обчислювальні технології є важливими через проведення наукових досліджень для створення квантового комп'ютера науковими та технологічними компаніями, що може призвести до суттєвих змін у всій галузі (створення нових методів наукового моделювання, зміни підходів до асиметричної криптографії). Також важливим є дослідження існуючих інструментів розвитку квантових програм.

Незважаючи на те, що дана робота містить лише короткий огляд існуючих на сьогодні технологій та

мови Q#, більш глибоке дослідження, включаючи практичне порівняння програм, створених за допомогою різних інструментів, є основним напрямком майбутніх досліджень.

Список літератури:

- [1] M. A. Nielsen and I. L. Chuang, “Quantum computation and quantum information,” Cambridge University Press, 2010.
- [2] Lily Chen, Stephen Jordan, Yi-Kai Liu, Dustin Moody, Rene Peralta, Ray Perlner, Daniel Smith-Tone, “Report on Post-Quantum Cryptography,” NIST Interagency Report 8105, 2016, 15 p.
- [3] Barends, R. et al, “Superconducting quantum circuits at the surface code threshold for fault tolerance,” Nature 508, 2014, pp. 500–503.
- [4] Yuanhao Wang, Ying Li, Zhang-qi Yin, Bei Zeng, “16-qubit IBM universal quantum computer can be fully entangled,” arXiv:1801.03782v3 [quantum-ph], 2018.
- [5] P. J. J. O’Malley, R. Babbush, I. D. Kivlichan, J. Romero, J. R. McClean, R. Barends, J. Kelly, P. Roushan, A. Tranter, N. Ding, B. Campbell, Y. Chen, Z. Chen, B. Chiaro, A. Dunsworth, A. G. Fowler, E. Jeffrey, E. Lucero, A. Megrant, J. Y. Mutus, M. Neeley, C. Neill, C. Quintana, D. Sank, A. Vainsencher, J. Wenner, T. C. White, P. V. Coveney, P. J. Love, H. Neven, A. AspuruGuzik, and J. M. Martinis, “Scalable quantum simulation of molecular energies,” Phys. Rev. X, vol. 6, p. 031007, Jul 2016. [Online]. Available: <http://link.aps.org/doi/10.1103/PhysRevX.6.031007>
- [6] Krysta M. Svore, Alan Geller, Matthias Troyer, John Azariah, Chris Granade, Bettina Heim, Vadym Kliuchnikov, Mariia Mykhailova, Andres Paz, Martin Roetteler, “Q#: Enabling scalable quantum computing and development with a high-level domain-specific language,” arXiv:1803.00652 [quant-ph], 2018.
- [7] A. Green, P. L. Lumsdaine, N. Ross, P. Selinger, and B. Valiron, “Quipper: A scalable quantum programming language,” in Proceedings of PLDI ’13, 2013.
- [8] D. Wecker and K. M. Svore, “LIQUi|: A Software Design Architecture and Domain-Specific Language for Quantum Computing,” 2014. [Online]. Available: [arXiv:1402.4467v1](https://arxiv.org/abs/1402.4467v1).
- [9] Damian S. Steiger, Thomas Häner, Matthias Troyer, “ProjectQ: An Open Source Software Framework for Quantum Computing,” arXiv:1612.08091 [quant-ph], 2018.
- [10] Rigetti Computing, “pyQuil,” 2016, accessed: 2016-08-10. [Online]. Available: <https://github.com/rigetticomputing/pyQuil>.
- [11] Christine C. Moran. “Quintuple: A tool for introducing quantum computing into the classroom,” Frontiers in Physics 6(JUL), 2018, p. 69
- [12] Robert S. Smith, Michael J. Curtis, William J. Zeng, “A Practical Quantum Instruction Set Architecture,” arXiv:1608.03355v2 [quant-ph], 2017.
- [13] Mathias Soeken, Thomas Haener, Martin Roetteler, “Programming Quantum Computers Using Design Automation,” arXiv:1803.01022 [quant-ph], 2018.
- [14] D. Wecker, M. B. Hastings, and M. Troyer, “Progress towards practical quantum variational algorithms,” Physical Review A, vol. 92, no. 4, p. 042303, 2015.

Надійшла до редколегії 21.11.2018



Н.Г. Аксак¹, А.А. Бессонов², И.В. Новосельцев³, О.Г. Руденко⁴

¹ ХНУРЭ, г. Харьков, Украина, nataliia.axak@nure.ua

² ХНЭУ, г. Харьков, Украина, oleksandr.bezsonov@hneu.net;

³ ХНУРЭ, г. Харьков, Украина, i.v.novoseltsev@gmail.com;

⁴ ХНЭУ, г. Харьков, Украина, oleg.rudenko@hneu.net

РАСПОЗНАВАНИЕ ИЗМЕНЕНИЯ РАЗМЕРА И ЦВЕТА ИЗОБРАЖЕНИЯ НА ОСНОВЕ СВЕРТОЧНОЙ НЕЙРОННОЙ СЕТИ

В статье рассмотрено применение сверточной нейронной сети Mask R-CNN для распознавания изменения размера и цвета изображения. Применение данной сети оправдано тем, что в отличие от других типов сверточных нейронных сетей данная сеть предназначена для не только для обнаружения всех объектов указанных классов и определения охватывающей рамки для каждого из них, но и для определения пикселей, принадлежащих каждому объекту каждого класса по отдельности. Описана архитектура этой сети. Предложена модификация алгоритма обучения многослойного перцептрона, входящего в состав Mask R-CNN, представляющая собой матричные варианты процедуры Качмажа (Уидроу-Хоффа). Вследствие наличия ограниченных помех предлагается использовать в алгоритме зону нечувствительности, для которой приведены процедуры настройки. Проведено экспериментальное исследование эффективности работы сети в задаче распознавания заболевания кожи — меланомы. В качестве обучающей выборки использовался набор фотографий из конкурса «ISIC 2016: Skin Lesion Analysis Towards Melanoma Detection». Полученные результаты свидетельствуют об эффективности развиваемого подхода.

РАСПОЗНАВАНИЕ ОБРАЗОВ, КЛАССИФИКАЦИЯ, СВЕРТОЧНАЯ ИСКУССТВЕННАЯ НЕЙРОННАЯ СЕТЬ, АЛГОРИТМ ОБУЧЕНИЯ, АКТИВАЦИОННАЯ ФУНКЦИЯ

Аксак Н.Г., Бессонов О.О., Новосельцев И.В., Руденко О.Г. Розпізнавання зміни розміру і кольору зображення на основі згорткової нейронної мережі. У статті розглянуто застосування згорткової нейронної мережі Mask R-CNN для розпізнавання зміни розміру і кольору зображення. Застосування даної мережі виправдано тим, що на відміну від інших типів згортальних нейронних мереж дана мережа призначена для не лише для виявлення всіх об'єктів заданих класів і визначення охоплюючої рамки для кожного з них, а й для визначення пікселів, що належать кожному об'єкту кожного класу окремо. Описана архітектура цієї мережі. Запропоновано модифікацію алгоритму навчання багатосарового перцептрона, що входить до складу Mask R-CNN, яка представляє собою матричні варіанти процедури Качмажа (Уїдроу-Хоффа). Внаслідок наявності обмежених перешкод пропонується використовувати в алгоритмі зону нечутливості, для якої наведені процедури налаштування. Проведено експериментальне дослідження ефективності роботи мережі в задачі розпізнавання захворювання шкіри - меланоми. В якості навчальної вибірки використовався набір фотографій з конкурсу «ISIC 2016: Skin Lesion Analysis Towards Melanoma Detection». Отримані результати свідчать про ефективність підходу, що розвивається.

РОЗПІЗНАВАННЯ ОБРАЗІВ, КЛАСИФІКАЦІЯ, ЗГОРТКОВА ШТУЧНА НЕЙРОННА МЕРЕЖА, АЛГОРИТМ НАВЧАННЯ, АКТИВАЦІЙНА ФУНКЦІЯ

Aksak N.G. Bezsonov O.O., Novoseltsev I.V., Rudenko O.G. Recognition of image resizing and color based on a convolutional neural network. The article describes the use of the Mask R-CNN convolutional neural network for recognition of changes in image size and color. The use of this network is justified by the fact that, unlike other types of convolutional neural networks, this network is designed not only to detect all objects of the indicated classes and determine the covering frame for each of them, but also to determine the pixels belonging to each object of each class separately. The architecture of this network is described. A modification of the learning algorithm of the multilayer perceptron, which is part of the Mask R-CNN, is a matrix version of the Kacmage (Widrow-Hoff) procedure. Due to the presence of limited interference, it is proposed to use the dead zone in the algorithm, for which tuning procedures are given. An experimental study of the effectiveness of the network in the task of recognizing skin disease - melanoma was conducted. A set of photographs from the competition "ISIC 2016: Skin Lesion Analysis Towards Melanoma Detection" was used as a training sample. The obtained results testify to the effectiveness of the developed approach.

RECOGNITION OF IMAGES, CLASSIFICATION, CLASSIFICATION, ARTIFICIAL NEURAL NETWORK, TRAINING ALGORITHM, ACTIVATION FUNCTION

Введение

На сегодняшний день одним из самых интенсивно развивающихся научных и технологических направлений является обработка и анализ изображений.

Среди важнейших задач, связанных с обработкой изображений и которые относят к задачам компьютерного зрения, следует отметить:

- **Classification** — классификация изображения по типу объекта, которое оно содержит.

- **Semantic segmentation** — определение всех пикселей объектов определённого класса или фона на изображении. Если несколько объектов одного класса перекрываются, их пиксели никак не отделяются друг от друга.

- **Object detection** — обнаружение всех объектов указанных классов и определение охватывающей рамки для каждого из них.

- **Instance segmentation** — определение пикселей, принадлежащих каждому объекту каждого класса по отдельности.

За последние годы был представлен ряд методов и алгоритмов, применяемых для решения данных задач, среди одними из наиболее эффективных являются искусственные нейронные сети (ИНС) [1]. Использование персептрона, радиально-базисных сетей, автоэнкодера, неокогнитрона, вероятностных сетей оказалось весьма эффективным при решении широкого круга задач. Однако новый толчок к интересу исследований в данной области дало появление в 1998 г. нового типа сетей — сверточных ИНС (СИНС).

Сверточная ИНС (Convolution Neural Network, CNN) впервые была предложена в [2,3] как развитие модели неокогнитрона, предназначенного для эффективного распознавания изображений. Впоследствии на основе СНС были построены сети R-CNN (Regions With CNNs) для применения CNN к задаче object detection. R-CNN создает ограничивающие рамки для каждого объекта на изображении или предложения регионов, используя процесс выборочного поиска. Fast R-CNN, увеличившая производительность R-CNN, осуществляет классификацию объектов каждого региона вместе с более жесткими ограничивающими рамками. Следующая сеть Faster R-CNN улучшила механизм генерации используемых в ней регионов-кандидатов за счет вычисления регионов не по изначальному изображению, а по карте признаков, полученных из CNN. Для этого был добавлен модуль под названием Region Proposal Network (RPN). Наконец, сеть Mask R-CNN развивает архитектуру Faster R-CNN путём добавления ещё одной ветки, которая предсказывает положение маски, покрывающей найденный объект, и, таким образом решает уже задачу instance segmentation. При получении изображения сеть выдает объекты (bbox), ограничивающие рамки, классы (class) и маски (mask).

Следует отметить, что Mask R-CNN является наиболее быстродействующей сетью на данный момент.

1. Структура сверточной нейронной сети

Первоначально структура сверточной нейронной сети создавалась с учетом особенностей строения некоторых частей человеческого мозга, отвечающих за зрение. В основу разработки таких сетей заложено три механизма:

- локальное восприятие;
- формирование слоев в виде набора карт признаков (разделяемые веса);
- субдискретизация (подвыборка).

Под локальным восприятием подразумевается, что на вход нейрона поступает не все изображение, а только некоторая его часть. Это позволяет сохранять конфигурацию изображения при переходе от слоя к слою.

Идея разделяемых весов подразумевает, что к большому числу связей применяется небольшой набор весов, т. е. каждая область изображения, на которые оно разделено, будет обработана одним и тем же набором весов. При таком искусственно созданном ограничении весов улучшается свойство сети к обобщению.

СНС состоит из слоев свертки, субдискретизации (подвыборки) и слоев полносвязной нейронной сети.

2. Слои сверточной нейронной сети

CNN получили свое название от оператора «свертки». Основная цель свертки в случае CNN — извлечь элементы из входного изображения. Свертка сохраняет пространственные отношения между пикселями, изучая особенности изображения, используя маленькие квадраты входных данных.

Каждый нейрон в плоскости сверточного слоя получает свои входы от некоторой области предыдущего слоя (локальное рецептивное поле), то есть входное изображение предыдущего слоя сканируется небольшим окном и пропускается сквозь набор весов, а результат отображается на соответствующий нейрон сверточного слоя.

Подвыборочный слой уменьшает масштаб плоскостей путём локального усреднения значений выходов нейронов. Таким образом, достигается иерархическая организация. Последующие слои извлекают более общие характеристики, меньше зависящие от искажений изображения.

За каждым сверточным слоем следует слой субдискретизации (подвыборки), или вычислительный слой, который производит уменьшение размерности изображения путём локального усреднения значений выходов нейронов (рис. 1).

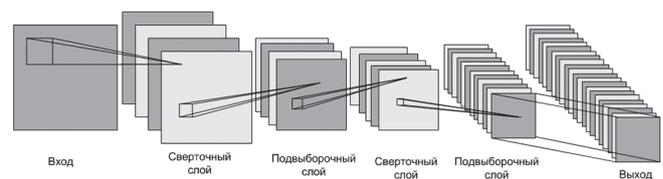


Рис. 1. Структура сверточной нейронной сети

В архитектуре сверточной сети принято считать, что наличие признака важнее информации о его расположении. Поэтому из нескольких соседних нейронов в карте признаков выбирается максимальный и его значение считается одним нейроном в карте признаков меньшей размерности.

Отличие слоя подвыборки от слоя свертки состоит в том, что в последнем области соседних нейронов перекрываются, чего не происходит в слое субдискретизации.

Таким образом, СНС строится путем чередования слоев свертки и субдискретизации. На выходе сети обычно устанавливается несколько слоев полносвязной нейронной сети, на вход которых подаются конечные карты признаков. Каждый нейрон этого слоя представляет собой персептрон, который имеет нелинейную функцию активации.

3. Методы настройки параметров сверточной нейронной сети

Для обучения сверточных нейронных сетей может применяться как стандартный метод обратного распространения ошибки, так и его различные модификации. Рассмотрим один из них.

Рассмотрим обучение многослойного персептрона (МП) с M входами и L выходами, S слоями, числом нейронов выходного слоя P , скрытого – N , все нейроны которого имеют активационную функцию вида

$$f(x) = \frac{1}{1 + e^{-\alpha x}}.$$

Введем следующие обозначения:

$$W^{*s}(k) = (W_1^{*s}(k), W_2^{*s}(k), \dots, W_p^{*s}(k))^T -$$

матрица оптимальных весов выходного слоя $P \times L$;

$$w_i^{*s}(k) = (w_{i1}^{*s}(k), w_{i2}^{*s}(k), \dots, w_{iL}^{*s}(k))^T -$$

вектор оптимальных весов i -го нейрона выходного слоя $P \times 1$;

$$W^s(k) = (W_1^s(k), W_2^s(k), \dots, W_p^s(k))^T -$$

матрица оценок весов выходного слоя МП $P \times L$;

$$w_i^s(k) = (w_{i1}^s(k), w_{i2}^s(k), \dots, w_{iL}^s(k))^T -$$

вектор оценок весов i -го нейрона выходного слоя $P \times 1$;

$$W^{*i}(k) = (W_1^{*i}(k), W_2^{*i}(k), \dots, W_p^{*i}(k))^T -$$

матрица оптимальных весов нейронов i -го скрытого слоя ($i = 1, \dots, S - 1$) $N \times P$;

$$w_i^{*i}(k) = (w_{i1}^{*i}(k), w_{i2}^{*i}(k), \dots, w_{iN}^{*i}(k))^T -$$

вектор оптимальных весов нейронов i -го скрытого слоя $N \times 1$;

$$F^s(x, w^s) = (f_1^s(w_1^{Ts}(k)x(k)) \quad f_2^s(w_2^{Ts}(k)x(k)) \quad \dots \quad f_L^s(w_L^{Ts}(k)x(k)))^T$$

$$F^s(x, w^s) = (f_1^s(w_1^{Ts}(k)x(k)) \quad f_2^s(w_2^{Ts}(k)x(k)) \quad \dots \quad f_L^s(w_L^{Ts}(k)x(k)))^T -$$

вектор активационных функций нейронов выходного слоя $L \times 1$;

$$\nabla f(x, w) = \text{diag}[\nabla f(w_1^T(k), x(k)) \quad \nabla f(w_2^T(k), x(k)) \quad \dots \quad \nabla f(w_L^T(k), x(k))] -$$

матрица производных активационных функций слоя $L \times L$; $\nabla f(w_i^T(k), x(k))$ – первая производная активационной функции i -го нейрона.

Тогда выходные сигналы МП и ошибка аппроксимации могут быть представлены следующим образом:

– требуемый вектор выходных сигналов

$$y^*(k) = W^{*s}(k)F^s(x, w^{*s}(k)) + \xi_1(k); \quad (1)$$

– реальный вектор выходных сигналов

$$y(k) = W^s(k)F(x, w^s(k)) + \xi_2(k); \quad (2)$$

– ошибка

$$e(k) = y^*(k) - y(k) + \xi(k) = \tilde{\theta}^T(k)F(x, w(k)) + W^{*s}(k)(k)\tilde{F}(x, w(k)) + \xi_1(k) + \xi_2(k); \quad (3)$$

$$i = 1, \dots, S - 1.$$

где $\tilde{\theta}(k) = W^{*s}(k) - W^s(k)$ – матрица ошибок обучения $P \times L$; $\tilde{F}(k) = F(x, w^{*s}(k)) - F(x, w^s(k))$ – вектор ошибок активационных функций; $\xi_1(k)$, $\xi_2(k)$ – вектор ошибок выходного сигнала и нейросетевой аппроксимации соответственно.

С учетом этих обозначений градиентные процедуры обучения матриц весов выходного и i -го скрытого слоя МП, линеаризующие квадратичный функционал и представляющие собой процедуру обратного распространения ошибки, могут быть записаны следующим образом:

$$W^s(k) = W^s(k-1) - \gamma_{W^s}(k) \frac{1}{\|F(x, w^s(k-1))\|^2} \frac{\partial E(k)}{\partial W^s(k-1)} =$$

$$= W^s(k-1) + \gamma_{W^s}(k) \frac{F(x, w^s(k-1))e^T(k)}{\|F(x, w^s(k-1))\|^2}; \quad (4)$$

$$W^i(k) = W^i(k-1) -$$

$$- \gamma_{W^i}(k) \frac{1}{\|\nabla f(x(k), W^i(k-1))x(k)\|^2} \frac{\partial E(k)}{\partial W^i(k-1)} =$$

$$= W^i(k-1) + \gamma_{W^i}(k) \frac{\nabla f(x(k), w^i(k-1))e^T(k)x^T(k)}{\|\nabla f(x(k), W^i(k-1))x(k)\|^2}, \quad (5)$$

$$i = 1, \dots, S - 1,$$

т.е. представляет собой матричные варианты процедуры Качмажа (Уидроу-Хоффа).

Здесь $\gamma_{W^s}(k)$ и $\gamma_{W^i}(k)$, $i = 1, \dots, S - 1$ – коэффициенты обучения.

Для повышения вычислительной устойчивости (4), (5) они могут быть модифицированы путем введения в них параметров регуляризации, т.е.

$$W^s(k) = W^s(k-1) + \gamma_{W^s}(k) \frac{F(x, w^s(k-1))e^T(k)}{\beta_{W^s}(k) + \|F(x, w^s(k-1))\|^2}; \quad (6)$$

$$W^i(k) = W^i(k-1) + \gamma_{W^i}(k) \frac{\nabla f(x(k), w^i(k-1))e(k)x^T(k)}{\beta_{W^i}(k) + \|\nabla f(x, W^i(k-1))x(k)\|^2}, i \neq S, \quad (7)$$

где $\beta_{W^s}(k)$, $\beta_{W^i}(k), i=1, \dots, S-1$ – параметры регуляризации.

Следует, однако, отметить, что если статистические свойства помех $\xi_1(k)$ и $\xi_2(k)$ не известны, а известно, что они ограничены по амплитуде, то процедуры (4)-(7) следует модифицировать путем использования в них зон нечувствительности так, как это было сделано для случая линейной Адалины, например, следующим образом:

$$W^s(k) = W^s(k-1) + \gamma_{W^s}(k) \frac{\alpha_{W^s}(k)F(x, w^s(k-1))e^T(k)}{\beta_{W^s}(k) + \|F(x, w^s(k-1))\|^2}; \quad (8)$$

$$\text{где } \alpha_{W^s}(k) = \begin{cases} g(e(k), \Delta_{W^s}(k)), & \text{если } \|e(k)\| > \Delta_{W^s}(k); \\ 0, & \text{если } \|e(k)\| \leq \Delta_{W^s}(k). \end{cases} \quad (9)$$

Так как при вычислении оценок матрицы W используется производная функции активации, то параметр зоны нечувствительности должен содержать коэффициент α , определяющий наклон сигмоиды

$$W^i(k) = W^i(k-1) + \gamma_{W^i}(k) \frac{\alpha_{W^i}(k)\nabla f(x(k), W^s(k-1))e(k)x^T(k)}{\beta_{W^i}(k) + \|\nabla f(x, W^s(k-1))x(k)\|^2}, \quad (10)$$

где

$$\alpha_{W^i}(k) = \begin{cases} g(e(k), \Delta_{W^i}(k)), & \text{если } \frac{\nabla f_{\min}}{\alpha} \|e(k)\| > \Delta_{W^i}(k); \\ 0, & \text{если } \frac{\nabla f_{\min}}{\alpha} \|e(k)\| \leq \Delta_{W^i}(k), \end{cases} \quad (11)$$

где $\nabla f_{\min} = \min[\nabla f_1(k), \nabla f_2(k), \dots, \nabla f_L(k)] > 0$.

Настройка зон нечувствительности для процедур (8) и (10) может осуществляться следующим образом:

$$\Delta_{W^s}(k) = \Delta_{W^s}(k-1) + \frac{\alpha_{W^s}(k)\|e(k-1)\|}{\beta_{W^s}(k-1) + \|F(x, W^s(k-1))\|^2}; \quad (12)$$

$$\Delta_{W^i}(k) = \Delta_{W^i}(k-1) + \frac{\alpha_{W^i}(k)\|e(k-1)\|}{\beta_{W^i}(k-1) + \|\nabla f(x, W^i(k-1))x(k)\|^2}, i \neq S. \quad (13)$$

Эксперимент 1. Решалась задача распознавания заболевания кожи – меланомы, с помощью сверточной нейронной сети MRCNN [6]. Данная сеть является расширенной версией сети Faster RCNN, которая в свою очередь является усовершенствованной версией сети Fast RCNN.

В качестве обучающей выборки использовался набор фотографий с конкурса «ISIC 2016: Skin Lesion Analysis Towards Melanoma Detection» [7]. В данном наборе содержится 900 фотографий различных повреждений кожи и соответствующие им битовые маски, позволяющие определить точный контур повреждения. Данный набор был разбит на 600 обучающих фотографий и 300 проверочных. Примеры изображений из обучающей выборки приведены на рис. 2.

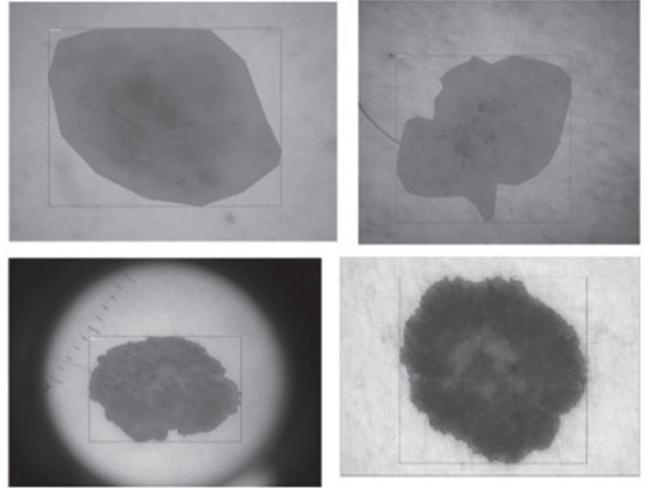


Рис. 2. Примеры изображений с наложенными масками из обучающей выборки

Обучение сети проводилось с использованием весов из предобученной на выборке “Coco” (<http://cocodataset.org>) сети resnet-101. Обучение проводилось на протяжении 30 эпох, и его результаты приведены на рис. 3. На рис. 3а показан график изменения общей ошибки обучения, состоящей из ошибки детектора «Bounding Box» – рис. 3б, ошибки классификатора 3в, ошибки предсказателя маски – 3г и ошибки модуля, предлагающего регионы, содержащие объекты для распознавания. Результаты распознавания показаны на рис. 4. Как видно из результатов, сеть MRCNN осуществляет эффективное распознавание повреждения кожи, указывая область повреждения и предлагая точную попиксельную маску. Данная сеть также позволяет отслеживать рост повреждения. Как видно на одном из фрагментов рис. 4, сеть распознала два региона – первичный и вторичный, являющийся результатом роста новообразования. Это свидетельствует о том, что предлагаемый подход позволяет отслеживать динамику развития заболевания.

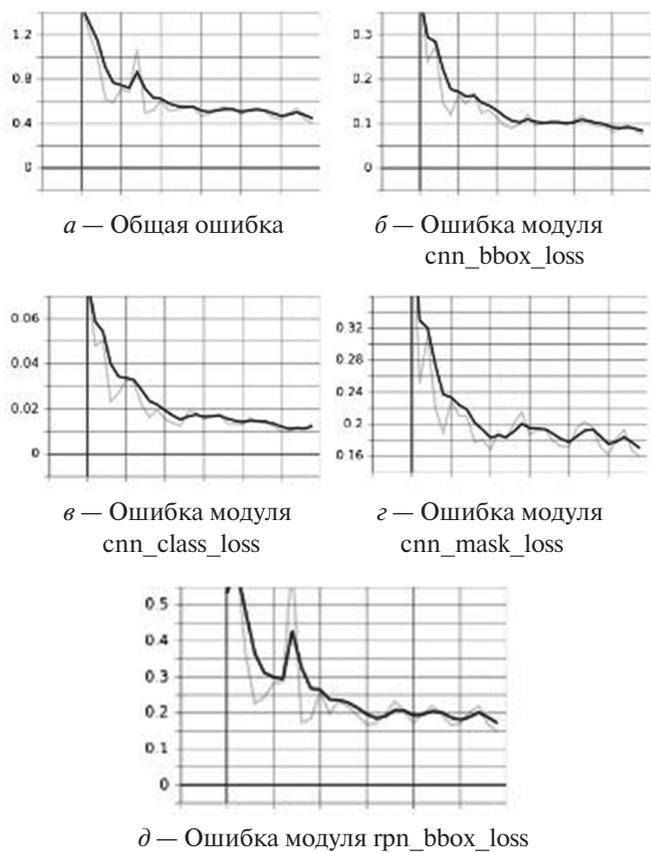


Рис. 3. Результаты обучения сети

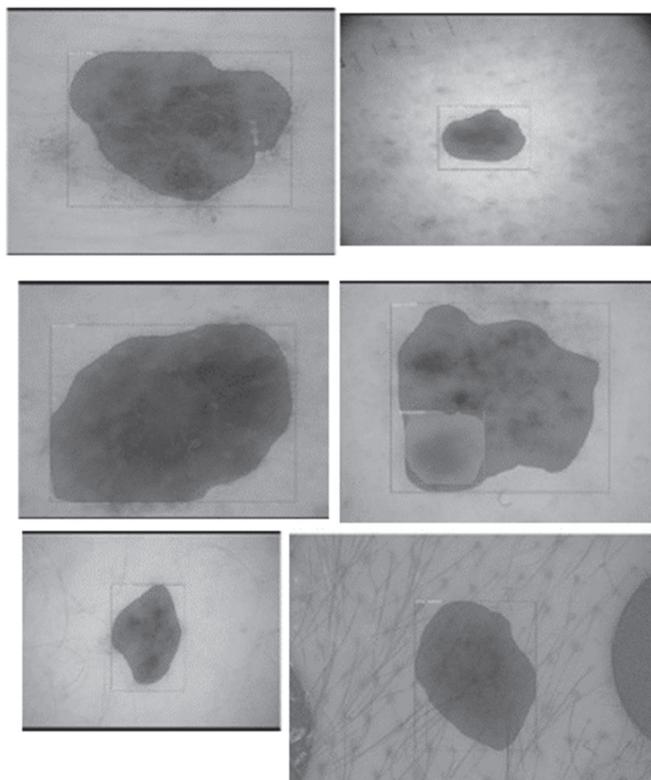


Рис. 4. Результаты распознавания

Эксперимент 2. В данном эксперименте помимо решения задачи обнаружения кожного повреждения решалась задача классификации заболевания (доброкачественное образование или

злокачественное). Для обучения использовалась база фотографий с кожными повреждениями из Эксперимента 1. К данной базе прилагалась маска, позволяющая определить границы повреждения и описание, содержащую информацию о типе заболевания. Примеры изображений, использованных для обучения представлены на рис. 5. Результаты классификации представлены на рис. 6. Зеленым цветом обозначались повреждения, которые не были распознаны сетью как злокачественные, красным — обозначены повреждения, которые с высокой вероятностью являются злокачественными. На некоторых изображениях вероятность злокачественности новообразования была близка к пороговому уровню, но не превышала его. Такие образования обозначались голубым цветом. Точность классификации после 70 эпох обучения составила 91,63%.

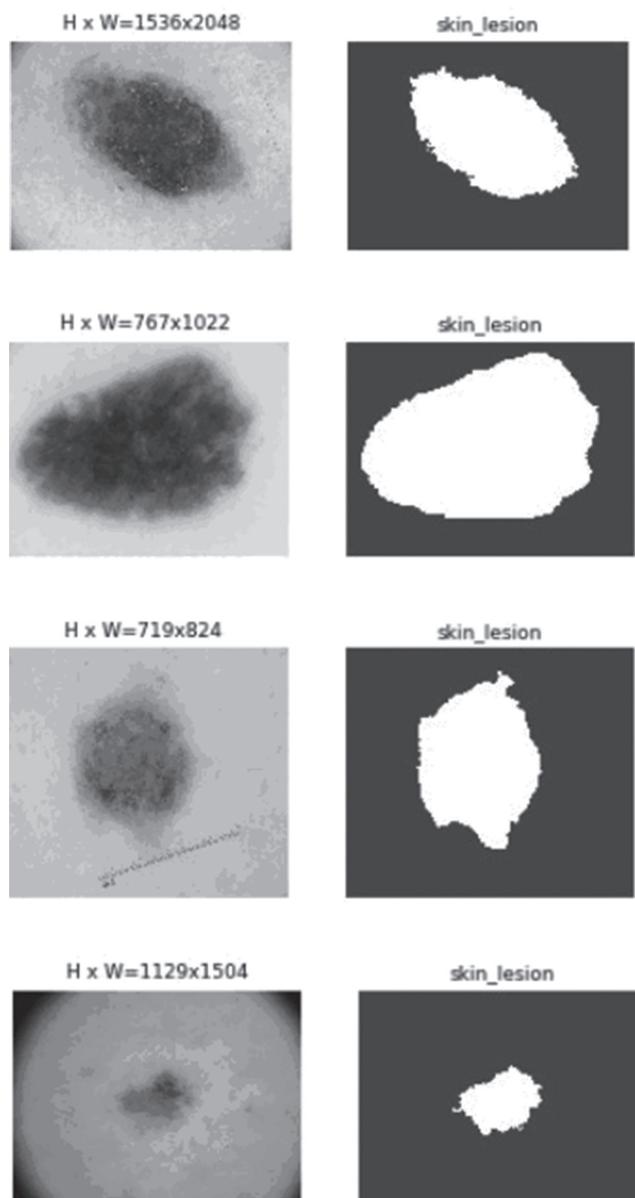


Рис. 5. Примеры изображений, использованных для обучения

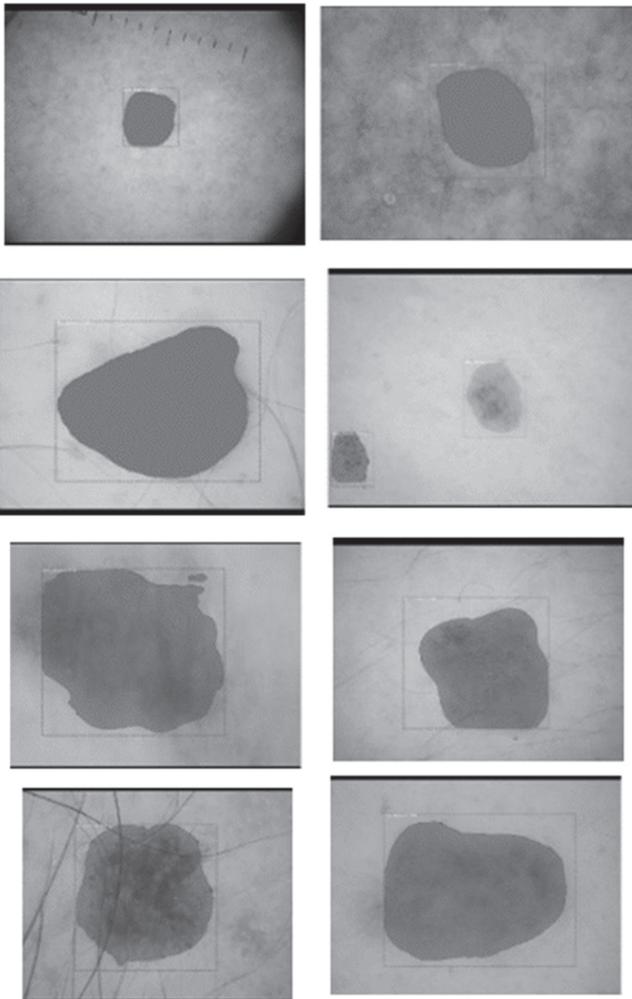


Рис. 6. Результаты классификации

Выводы

В работе рассмотрено применение сверточной нейронной сети Mask R-CNN для распознавания изменения размера и цвета изображения. Предложена модификация алгоритма обучения многослойного перцептрона, представляющая собой матричные варианты процедуры Качмажа (Уидроу-Хоффа), содержащие зону нечувствительности и приведены процедуры настройки этой зоны.

Полученные результаты проведенных экспериментов свидетельствуют об эффективности развиваемого подхода.

Список литературы:

- [1] Хайкин С. Нейронные сети: полный курс. М.: Вильямс, 2006. — 1104 с.
- [2] LeCun Y. A theoretical framework for backpropagation // Proc. of IEEE. — 1998. — P. 21-28.
- [3] LeCun Y., Bottou L., Bengio Y., Haffner P., Gradient Based Learning Applied to Document Recognition // Proc. of IEEE, 1998. V. 86, №11. — P. 2278-2324.
- [4] Hinton G.E., Osindero S., The Y.-W. A fast learning algorithm for deep belief nets. Neural Comput., 18(7):1527-1554, 2006.
- [5] Krizhevsky A., Sutskever I., Hinton G. E. ImageNet classification with deep convolutional neural networks. // In Advances in Neural Information Processing Systems, 2012. — P. 1097-1105.
- [6] He K, Gkioxari G, Dollár P, Girshick R. Mask R-CNN <https://arxiv.org/abs/1703.06870>
- [7] ISIC 2016: Skin Lesion Analysis Towards Melanoma Detection <https://challenge.kitware.com/#phase/566744dccad3a56fac786787>

Поступила в редколлегию 05.12.2018

RESUME

Ye. Bodyankiy, A. Shafronenko, K. Patlan

FUZZY CLUSTERING OF DATA ARRAYS BASED ON THE EVOLUTIONARY METHOD OF CAT SWARM OPTIMIZATION

Background: The problem of probabilistic fuzzy clustering of arrays of multidimensional observations in the general case reduces to a nonlinear programming problem that generates a set of local extrema of the goal function. In order to find the best solution in conditions of a lot of extrema, evolutionary algorithms based on “cat swarms” have recently been used. To speed up the search process, modification of the cat swarm optimization, which is characterized by an increased speed action, is introduced.

Materials and methods: The paper uses a data mining apparatus, methods for fuzzy cluster analysis, fuzzy C-means algorithm, evolutionary optimization algorithms based on particle swarms and, above all, a cat swarm optimization.

Results: An accelerated modification of the cat swarm optimization was introduced, which allows solving a lot of optimization problems arising in the framework of the problem of fuzzy probabilistic clustering of arrays of observations in conditions of overlapping classes.

Conclusion: The article proposes the method for solving the problem of fuzzy probabilistic clustering under the conditions of overlapping classes and multiextremal goal function using an evolutionary optimization algorithm of the “swarm of particles” type. To speed up the search process, a randomized modification of the cat swarm optimization, which is notable for its simple numerical implementation and improved speed, has been introduced.

Y.V. Bodyansky, K.E. Petrov, A.A. Deineko

EVOLVING RADIAL BASIS NEURAL NETWORK AND EXTREME LEARNING OF ITS PARAMETERS

Background: The active use of artificial neural networks (ANN) for solving various tasks associated with Data Mining such as classification, clustering, forecasting, etc., is due to their high efficiency in approximating non-linear dependencies, the ability to work with incomplete and “noisy” data, as well as their ability to easily adapt to changes in environmental conditions.

Particular interest is the development of systems that process incoming data sequences online. In this case, the question of learning speed comes to the fore, which significantly limits the class of ANNs suitable for work in this mode. Promising, from the point of view of optimality in terms of learning speed, is the use of radial basis neural networks (RBFN). Their significant advantage is that the output signal in such an ANN depends linearly on the tunable parameters.

The main disadvantage of RBFN is its susceptibility to the “curse of dimensionality” — an exponential increase in the number of neurons with increasing dimension of the input space.

Therefore, it is very relevant to develop an approach to learning the parameters and the formation of the RBFN architecture, in which the structure of a neural network can change in the process of learning as new information arrives at its input (in online mode).

Materials and methods: It is proposed to use a hybrid evolving ANN based on RBFN with a variable number of neurons and a T. Kohonen selforganizing map (SOM), which is responsible for managing the number of RBFN neurons and setting the location of the centers of kernel functions in the selflearning mode. During the operation of this network, the first observation is fed to the RBFN input, where the first neuron is formed according to the principle “neurons at data points”. Upon receipt of subsequent observations, they are first processed in SOM, where they are compared with existing centroids (kernels), and then, if no match is found, a new kernel is formed and accordingly a new neuron in RBFN. Thus, at the first stage, the RBFN structure is formed based on the use of SOM and at the second stage its extreme learning (ELM) is performed.

Results: The main problems arising in the development of intelligent systems that allow data to be processed online are considered.

Proposed an approach to learning the evolving RBFN, which allows in realtime mode not only setting up the output weights of the ANN and the parameters of the radial basis functions, but also gives the opportunity to change the network architecture in the process of receipt of new information at its input. This allowed to solve the problem of the “curse of dimensionality”.

Conclusion: A significant advantage of the described approach to the learning of evolving RBFN is the possibility of changing the network architecture in the process of entering new data at its input, which allowed to solve the problem of "curse of dimensionality".

To increase the learning rate of RBFN was used the learning algorithm called extreme learning machine (ELM).

This all provided a high rate of learning an artificial neural network with acceptable accuracy of the results, which were confirmed by a number of computational experiments.

Promising are studies related to establishing the dependence of the learning rate of RBFN and the accuracy of the results on the use of various types of activation kernel functions.

V.V. Moskalenko, A.S. Moskalenko, A.G. Korobov, M.O. Zaretskiy

**AERIAL COMPACT DRONE AUTONOMOUS NAVIGATION SYSTEM –
THE MODEL AND TRAINING ALGORITHMS**

Background: Autonomous navigation is crucial for search, rescue and remote inspection because teleoperation by video streams is extremely challenging when flying close to buildings or trees and in an indoor environment. Besides, Global Positioning System (GPS) may not be reliable in cases of low satellite coverage and signal multipath interference. Vision-based solution seems particularly suitable to the autonomous navigation in terms of the weight, cost and information it can provide. With an onboard camera, the drone can estimate the ego-motion and obtain the information on the flight environment simultaneously.

Materials and methods: The paper presents a novel model of convolutional neural network for visual feature extraction, extreme-learning machine for position displacement estimation and boosted information-extreme classifier for obstacle prediction with new training algorithms to build decision rules of autonomous navigation system for compact drones are presented in the paper. Growing sparse-coding neural gas algorithm for unsupervised training of the convolution filters, supervised learning algorithms for training decision rules and simulated annealing search algorithm for fine tuning are proposed. The complex criterion for choosing parameter of feature extractor model is considered.

Results: Our approach perform better than the geometric ORB-SLAM. In particular, for Sequence 7 from KITTI dataset, the translation error of length reduces nearly 60% under the same frame rate.

Conclusion: Model of autonomous navigation system of compact drone is proposed. Developed complex criterion for evaluating effectiveness of the model taking into account computation complexity of feature extractor and accuracy of decision rules also training algorithm.

S.G. Udovenko, D.V. Kelembet

CONTROL OF COMPRESSOR STATION UNITS USING A FUZZY INVERSE MODEL

Background: The tasks of control the technological processes of the compressor station are to regulate the flow, pressure and temperature of the gas. The gas pumping units are controlled by autonomous automatic control systems. For the organization of the effective functioning of the control systems of the compressor station, it is advisable to use systems with elements of artificial intelligence. Perspective is the study of the possibility and efficiency of the use of inverse fuzzy compensation regulators in closed digital control systems with an internal fuzzy model, reflecting the dynamic properties of technological processes of the compressor station of the gas pumping system.

Materials and methods: To simulate a fuzzy control system for compressor station units using an inverse model, the capabilities of the MATLAB software environment were used, in particular, the Simulink simulation package and the Fuzzy Logic Toolbox synthesis fuzzy control systems synthesis package. For different output pressure ranges, the dynamic properties of gas pumping units may vary. In this regard, the total range of output pressure was divided into 5 intervals, which allowed, after selecting the membership functions, to form 5 linear subsystems..

Results: The results of the simulation of the system with an abrupt change in the input signal confirmed the advantages of the proposed fuzzy control scheme compared to the PID control. Changing the mode of operation is carried out at the top level of the automated process control system of the compressor station.

Conclusion: The considered approach to building a system of fuzzy control of units of a compressor station confirms the effectiveness of using inverse fuzzy models to maintain a given gas pressure at the outlet of a compressor station.

O.G. Rudenko, O.O. Bezsonov, O.G. Lebediev, O.S. Romanyk

**CRITERIAS FOR THE SELECTION OF THE PERSEPTRONIC MODEL FOR FORECASTING:
ANALYSYS AND PRACTICAL RECOMENDATIONS REGARDING THEIR USAGE**

Background: An analysis of the existing and rather common statistical (RMSE, MARE, MAE, etc.) and some information criteria (AIC, KIC, HQ, etc.) of the choice of the perceptron model for forecasting time series is given. Their advantages and disadvantages are presented. In addition, the newly proposed criterias, which have not yet been widely disseminated, are considered. Among these criteria there are the accuracy criterion of the direction and its modification, directed symmetry and its modification, weighted information criterion and its adaptive modification.

Materials and methods: In this paper we consider the forecasting of time series by means of neural networks. It has been shown that recently proposed criteria that combine statistical and information criteria in a single weighted one, have a number of significant drawbacks. These criteria do not take into account the characteristics of the time series and include a subjective approach to determining weighting factors.

Results: On the basis of the results of the analysis, recommendations for choosing estimations of prediction in solving of practical problems were developed. Due to the impossibility to allocate one universal criterion it is recommended to use a certain set of error metrics. It is noted that the choice of qualitative criterion depends on the purpose of forecasting and the availability of information on statistical data properties and noise parameters.

Conclusion: The considered criterias for selecting the perceptron model are analyzed for forecasting the volume of passenger traffic at airports in Germany. The results indicate that the criteria for choosing a model based on the median of forecasts have less variability than others, while geometric average criteria have greater variability.

Mariya Shirokopetleva, Oleksandr Ponomarenko, Zoya Dudar

COMPARISON OF TIME SERIES FORECASTING METHODS

Background: Forecasting events based on formalized models plays one of the key roles in the current forecasting of a variety of processes: from estimating economic indicators to weather forecasting, etc. This is due to the desire to have a more stable understanding of the situation in their subject area. However, some of the methods are more effective than others.

Materials and methods: In this paper, the method of prediction of ARIMA and the method based on artificial neural networks is used. The results of the forecast of each of the methods and their prediction error are considered. Due to this error, we can estimate the relative deviation of each of the predictions from real data at a specific point and analyze the trends.

Results: This work contains analysis and classification of models of prediction of time series according to their specific categories and criteria. Among the methods, two of the most popular and effective today are outlined. These methods (ARIMA, ANN) were also ranked practically, in real time series.

Conclusion: The article summarizes the trends and performance of each of the comparable methods. In addition, consideration is given to the possibilities of combining these methods.

Yevgeniy Putyatin, Pavlo Orobinskyi, Vyacheslav Lyashenko

DETECTION OF LUNG NODULES ON DIGITAL MEDICAL IMAGES

Background: Analysis of digital images is one of the most promising research methods. At the same time, the choice of image analysis tools largely depends on the object of study. Among such objects of research an important place is given to digital medical images. This analysis allows to improve the quality of medical diagnostics, to identify a possible human disease at an early stage.

Materials and methods: For these studies, various digital image processing tools were used: binarization, breakdown into energy levels, probabilistic segmentation and neural networks.

Results: Analysis of the research problems contributed to the disclosure of the specifics of his decision. It is shown that the analysis of digital medical images is based on the use of various image processing methods. At the same time, such image processing methods should be used in a certain sequence. This improves the efficiency of medical image analysis. This is achieved by obtaining additional information. As a result of this analysis, the reliability of diagnosing the state of human health, the possibility of detecting foci of diseases in the early stages increases.

Conclusion: A procedure has been proposed for identifying lung nodules that are located in difficult areas of their location. A comparative analysis of the effectiveness of the proposed procedure for the identification of lung nodules in various images. The choice of the neural network activation function to identify lung nodules is substantiated. The results obtained indicate the practicality, reliability and expediency of the application of the developed procedure for the identification of lung lesions.

V.M. Levykin, O.V. Chala

**METHOD OF SUPPORT OF MANAGERIAL DECISIONS IN CONDITIONS
OF UNCERTAINTY ON THE BASIS OF TEMPORAL KNOWLEDGE**

Background: The problem of support for managerial decisions under conditions of uncertainty regarding the object and the management process is considered. Uncertainty about the control object is characterized by incomplete information about its status. Uncertainty about the management process is characterized by incomplete formal knowledge of this process as a result of the use of non-formalized knowledge "by default" when making managerial decisions. Support for management decisions under uncertainty conditions is realized using the knowledge base. Such a knowledge base needs to be kept up to date. For its replenishment, communicative and passive methods of knowledge engineering are used. The first set of methods requires a significant amount of time and does not allow the database to be kept up to date. The second group of methods can be automated, which provides the ability to incrementally replenish the knowledge base to support decision-making. In the conditions of uncertainty about the object and the control process, obtaining causation dependencies causes difficulties, however, temporal dependencies can be obtained. The latter specify the time communication between the states of the control object, as well as the contextual conditions for executing control actions.

Materials and methods: methods of decision support in conditions of uncertainty.

Results: Key features of the integrated management solution as one that includes all alternatives for choosing a sequence of actions to eliminate the current problem situation are identified. The temporal model of integrated management solution is proposed as well as method of support of managerial decisions in conditions of uncertainty based on temporal knowledge.

Conclusion: The temporal model of a complex management solution, which contains a multitude of states of the control object, including the initial and target states, the set of permissible transitions between these states, executed as a result of control actions, the set of temporal rules for these transitions, and also the evaluation of temporal rules are proposed. The model provides the opportunity for rational choice of management solution based on the integral evaluation of temporal rules.

The method of support of managerial decisions in the conditions of uncertainty based on temporal knowledge is proposed. The method includes the stages of analysis of the current situation, the formation of a comprehensive management solution, the construction of temporal dependencies and the addition of knowledge base by the obtained dependencies. The method makes it possible to increase the effectiveness of supporting management decision-making by constantly adding knowledge base to the temporal rules used to evaluate these decisions.

Kaminsky R., Shakhovska N., Ladanivskyy B., Savkiv L.

**CONSTRUCTION OF THE GEOELECTRIC CUTTING MODEL
BY FINAL INTERPRETATION OF THE DATA PROCESSING METHOD
FOR THE POSITATION OF THE FIELD IN A NEIGHBORHOOD ZONE**

Background: Various geophysical methods and techniques are used to study and study the physical processes and phenomena occurring in the surface layers and depths of the Earth, as well as on its surface and in the near-Earth space. Such studies are organized in different ways: it may be systematic scientific research in the form of continuous monitoring or regime geophysical observations, or periodic studies of individual regions or areas. Each of these directions provides important background information for further study of these processes.

Continuous geophysical observations are conducted continuously in stationary and temporary observation points. At present, the collection and registration of physical fields occurs mainly automatically, through the creation and replenishment of diverse databases. According to regular and longterm observations, it is possible to evaluate and analyze the field parameters and their characteristics, study the dynamics of changes over a long period of time, and control the critical indicators. Usually such studies are provided by international networks, world

and national centers, consortia, geological, ecological and other services, associations, agencies. The mentioned organizations, in addition to other types of data, work with geomagnetic, seismic, magnetotelluric observations, formulate appropriate archives, provide visual representation of such data in online mode and, in addition, provide special or direct access to them.

The purpose of the work is to elaborate a method for determining the characteristic indices of geoelectric layers and to construct these geocoding data based on these data.

Method: The essence of the method is that the characteristic parameters of the geological environment, that is, the parameters of the geoelectric section model, are calculated from the obtained at the stage of the formal interpretation of the electric resistance curve with the application for finding the necessary arrays of resistivity and corresponding depths of the first and second derivatives of the curve $\rho(h)$. The result of a formal interpretation is the array of values of resistance and depths. There are two ways to find derivatives from the data in a table representation: the first one — by searching for some approximate function that would replace the table values. And the second one is to find derivatives of neighboring points based on the physical content of the derivative itself, that is, the first derivative is the speed calculated as the ratio of the ordinates of the neighboring points to the difference between their abscissas.

Results: The method of determining the characteristic parameters of the geological environment according to the table data of the electric resistance curve is developed for express analysis of the results of field measurements using elementary mathematical operations. This will allow to quickly build approximate models of the geoelectric section using simple computer technology, which is especially relevant in the field of experimental field work.

Conclusions: The method of calculation of parameters of the approximate model of the geoelectric section is developed on the table data obtained at the stage of the formal interpretation of the electric resistance curve. On the basis of the proposed method, the algorithm is described and a calculation program is created, the input and output data are described, the form of presentation of the result.

O.D. Mikhnova

MULTIMEDIA DATA EXTRACTION WITH FUZZY MATCH

Background: The introductory section of this paper provides a number of examples to enrich the developing area of artificial intelligent processing of multimedia content, starting from translation and voice recognition applications and up to video search by a sample fragment. Well-known approaches are briefly touched here, among which are: minimum edit distance, dynamic programming, Markov models, semantic and neural networks, Bayes classifier, n-gram language predictive model, regular expressions, supervised and unsupervised machine learning, template matching, description logics.

Materials and methods: Mathematical formulations, based on semi-group set theory provided above and applied to text and video content fuzzy match, equally true represent all the types of currently available multimedia data. The value of research lies mostly in theoretical groundings, but may also be used in practical search applications for content match.

Results: The proposed technique can be implemented as part of text or video parsing procedure. The problem of finding correct translation also resembles the problems of misspelling / mistyping, noisy video input due to environmental conditions or low camera characteristics / invalid settings, when correct output tuple is to be found based on the wrong initial observation.

Conclusion: In this paper, an attempt of a unique mathematical model development is made for text and image recognition that could cope with long length data. A single variant of biomedical, textual, audio and visual data presentation enables development of highly accurate systems with more initial variables.

V.M. Levykin, I.O. Iuriev

DEVELOPMENT OF INFORMATION TECHNOLOGY MANAGEMENT SYSTEM PROVIDING IT SERVICES

Background: An analysis of existing approaches and management methodologies for providing IT services was conducted. Their main advantages and disadvantages are determined. According to the results of the analysis of the use of the library, ITIL is the most grounded means for constructing modern systems of management of IT services.

The analysis of service-oriented information systems and technologies based on the principles of ITIL is carried

out. The basic mathematical models which are the basis for construction of modern ITSM-systems are determined. The main advantage of ITSM systems is that there is practically no industry-specific application of services. Depending on the industry, different ways of organizing work and building services are used.

Materials and methods: methods of analysis of processes of the life cycle of systems, methods of algorithm theory, methods of developing web-based software.

Results: information technology management system providing IT services.

Conclusion: The proposed information technology, which is a sequence of work, is united in 4 main stages: the formation of a request for a change in functional requirements, the search for IT services in the directory, the selection of the best IT service to functional requirements, an assessment of the impact of the selected IT service on the infrastructure of the enterprise.

The software implementation of this information technology is developed. The results of the application of the software implementation of information technology in the implementation of the request to change the functional requirements are presented. The results of its application allowed to reduce the expenses of the working time of the personnel of the enterprise in carrying out work on the implementation of requests for the change of functional requirements.

Kyrychenko I.V.

**CONCEPT OF ADAPTIVE E-LEARNING SYSTEM BASED
ON CONTENT IDENTIFICATION**

Background: Currently, the fundamental changes that are taking place in modern society determine the evolution of educational technologies, promote the active implementation of distance education as one of the areas of reforming and developing the educational system of Ukraine. The efficiency of remote technology application depends to a large extent on the level of readiness of teachers to implement the remote training, scientific-methodical and logistical support.

Materials and methods: based on the need to develop and implement in the curriculum of distance learning programs that meet the best world standards and provide training for professionals at a high professional level.

Results: The work considered the existing requirements for distance learning introduced today. The teaching subject, which is taught remotely, has certain didactic features, it is proved that it is a clear structure, which includes methodical, informative, diagnostic-correction blocks.

Conclusion: A formed approach to the implementation of information and communication technologies in the learning process, which primarily involves the interaction and integrity of the three components: organizational forms, didactic process and qualifications of teachers.

I.A. Revenchuk, K.V. Pertseva, O.I. Marenich

SOFTWARE IMPLEMENTATION OF SEARCH QUERY CLUSTERING

Background: forecasting, research of market trends allows to assess the situation of trends and achieve significant market advantages. For owners of e-markets, it is important to know what customers are looking for in electronic stores and by which priorities. For solution of this problem need to analyze all search queries and also use the k-means algorithm for clustering by phrase composition. When collecting a semantic core for a site, the most important step is processing, cleaning and structuring the resulting list of queries. Clustering search queries is the process of forming on the basis of a common list of key phrases of individual groups (clusters) that are related to each other by the description of the object, common sense, and context of use.

Materials and methods: the k-means algorithm as an approach to the software implementation of clustering of search queries in the Python language is considered. Hierarchical clustering is presented by SERP.

Results: a software implementation of the clustering of search queries by Python language was performed, clustering by phrase composition based on k-means algorithm, hierarchical clustering is represented by SERP.

Conclusion: the article proposes the software implementation of clustering search queries, including the composition of phrases. The results of the analysis are relevant for online store owners, for whom it is important to know what customers are looking for in electronic stores and by what priorities.

Yu.S. Kurskoy

MEASUREMENT MODEL FOR SYSTEM WITH SELF-ORGANIZATION FUNCTION

Background: The most important scientific problems of the day is the problems of research and measurements in evolving nonlinear dynamic systems with a self-organizing function. A systematic approach to the issue of measurements in such systems was proposed in the framework of the nonlinear measurement theory. The task of this paper is to develop a measurement model for systems with a self-organizing function.

Materials and methods: As an object of study, an open, dissipative, evolving system with a self-organization function is considered in the problem. Measuring the state of a such system is represented as an experiment with a purpose to determine the intervals of the dynamic variables possible values; classification of the processes dynamics; determining the time of return to the equilibrium state after the normalized external influence.

Results: The paper presents a measurement model for such systems. The measurement model includes the measurements of dynamic variables of the system in the equilibrium and excited states. The results of the measurement are: the intervals of the dynamic variables values, the fractal dimension, the Shannon entropy and the dynamics prediction time. For assess of the evolutionary processes the model has the entropy and time scales. As a quantitative assessment of the system state the model offers the time of return to the equilibrium state after the normalized external influence. An entropy method for estimating normalized external influence is proposed too.

Conclusion: The measurement model can be used for research, measure, and control evolving systems with a self-organizing function by physical, biological, and other nature. It allows to evaluate evolutionary changes. The choice of a time to return to equilibrium as a quantitative assessment of the state of dynamic systems will allow the creation of appropriate metrological assurance of such studies and traceability of measurement results, based on the time standard.

Y.V. Zhmaeva, L.E. Chala

A HYBRID METHOD FOR ESTIMATING THE COMPLEXITY OF IT PROJECTS

Background: Determining the complexity of software development is one of the most critical aspects in the management of IT projects. Accurately determining the size of software applications helps project managers to effectively plan and execute assigned tasks. Compliance with the rules dictated by existing methods of assessing the complexity of IT projects is extremely important. Software developers have to produce significant volumes of the original product under conditions of constant expansion of user requirements, as a result of which the method of measuring the size of the program should be as unified as possible. A common method for solving this problem is functional point analysis, which allows functional decomposition of the software design process. Promising is the direction of research associated with the improvement of this method.

Materials and methods: The proposed method was tested on an example of an assessment of the complexity of designing the application "Hostel Issue Reporting", which allows you to send a request describing the problem that arose with equipment at the hotel. In the program and algorithmic implementation of the method using means of the method of functional points, PERT and ALARP.

Results: The analysis of existing methods for assessing the complexity of designing software applications. Ways of defining the boundaries and interaction of transactions of the analyzed programs are considered. An algorithm for correcting estimates of complexity obtained using the method of functional points is given. In the hybrid correction method, PERT and ALARP tools are used.

Conclusion: The article proposes a modified hybrid approach to assessing the complexity of IT projects based on the improvement of the functional point method through the use of risk management mechanisms and forecasting the costs associated with developing software applications.

Ivan Bozhko, Natalia Kozel, Grygoriy Chetverykov

QUANTUM COMPUTING TOOLS

Background: The research of quantum computing takes place by scientists and technological companies today. There are new tools appearing as well as new articles published on the quantum computing topic. Since the tools are new and have not been tested and used, the problem of understanding the specifics of each of them also takes place, hence the authors try to resolve in the given paper.

Materials and methods: The literature analysis was used for retrieving the information about the tools for investigation in this paper. Also test Q# algorithms and a computer with Microsoft Quantum Development Kit was used for modeling and holding the experiments.

Results: According to the retrieved data, it can be seen that Q# language today is a unique tool provided as a part of Microsoft Development Kit with the standard instruments for quantum programs creation included. It integrates with other .NET technologies, which provides the ways of creating quantum programs as parts of classic programs. Also, the way of writing programs via Q# differs from the classic programming languages, which provides the flexibility of creating quantum programs without additional code for making requests and so on by means of having special quantum types and standard functions.

Conclusion: A high-level overview of the currently existing tools including Q# language was provided in this work, as per the initial goals. However, the research requires continuation in order to provide more detailed results of practical usage of the tool (e.g. implementation of one of the quantum algorithms). Another direction of the investigation could be explicit comparison of the same algorithm implemented using different tools and, therefore, comparing the efficiency of them.

N.G. Axak, O.O. Bezsonov, I.V. Novoselcev, O.G. Rudenko

**RECOGNITION OF IMAGE RESIZING AND COLOR BASED
ON A CONVOLUTIONAL NEURAL NETWORK**

Background: In recent years, a number of methods and algorithms that were used to solve image recognition problems were presented, among which one of the most effective are artificial neural networks (ANN). The use of perceptron, radial-base networks, autoconfigurator, neocognitron, probabilistic networks proved to be quite effective in solving a wide range of problems. However, a new impetus to the interest in research in this area was the appearance in 1998 of a new type of networks — convolutional ANN (CANN). Convolutional ANN was proposed as a development of a neo-cognitive system designed for effective image recognition. On the basis of CNN was constructed R-CNN network for object recognition. R-CNN creates a bounding box for each object in the image or region suggestions, using the process of choosing a search method. The fast R-CNN, which increases the productivity of R-CNN, provides classifications for the objects of each region, along with more stringent constraints. Faster R-CNN improves the mechanism of generating candidate regions by analyzing the regions not by the original image, but by the definition obtained through CNN. To do this, a module called Region Proposal Network (RPN) was added. The Mask R-CNN network develops the architecture of the faster R-CNN and adds another branch that encapsulates the latest object, and thus replicates the segmentation of the object. Obtained images show objects (bbox), bounding frames, classes (class) and masks (masks).

Materials and methods: In this paper, the main attention is paid to the application of artificial neural networks in the problem of recognition and classification of images. We consider a number of neural networks used to solve a considered problem and make a conclusion on the effectiveness of the Mask R-CNN network, which is the main focus of the work.

Results: In this paper we consider the problem of image recognition by means of CANN. The paper considers the use of the convolutional neural network Mask R-CNN to recognize the change in the size and color of the image. The modification of the training algorithm for multi-layer perceptron, which is a matrix version of the Kachmag's procedure (Widrow-Hoff procedure) containing the zone of insensitivity, is presented and the procedure for setting up this zone is presented.

Conclusion: The problem of recognizing and classifying images by means of artificial neural networks is considered. The main advantage of the proposed method is that due to the use of newest architectures and training algorithms, ANN are able to provide the quality of object recognition with accuracy to the pixel.

The result of solving the practical task is the complete automation of the process of recognition and classification of images with various skin injuries. The results of the work of the developed application confirm the high efficiency and speed of artificial neural networks during solving of the considered task.

ПРАВИЛА оформлення рукописів для авторів науково-технічного журналу «БІОНІКА ІНТЕЛЕКТУ»

Науково-технічний журнал «Біоніка інтелекту» приймає до друку написані спеціально для нього оригінальні рукописи, які раніше ніде не друкувались. Структура рукопису повинна бути такою: індекс УДК, відомості про авторів, заголовок, анотації (на трьох мовах), ключові слова, вступ, основний текст статті, висновки, список використаної літератури, резюме.

Відповідно до Постанови ВАК України від 15.01.2003 №7-05/1 (Бюлетень ВАК, №1, 2003, с. 2), стаття повинна мати такі необхідні елементи: постановка проблеми у загальному вигляді та її зв'язок із важливими науковими чи практичними завданнями; аналіз останніх досліджень і публікацій і виділення не вирішених раніше частин загальної проблеми в даній області; формулювання цілей та завдань дослідження; виклад основного матеріалу досліджень з повним обґрунтуванням отриманих наукових результатів; висновки з даного дослідження та перспективи подальших досліджень у даному напрямку.

Статті мають бути виконані в редакторі Microsoft Word. Формат сторінки – А4 (210×297 мм), поля: верхнє – 25 мм, нижнє – 20 мм, ліве, праве – 17 мм. Кількість колонок – 2, з інтервалом між ними 5 мм, основний шрифт Times New Roman, кегль основного тексту – 10 пунктів, міжрядковий інтервал – множник (1,1), абзацний відступ – 6 мм. Обсяг рукопису – від 6 до 12 сторінок (мови: українська, англійська, російська та мовою оригінала).

УДК друкується з першого рядка, без відступів, вирівнювання по лівому краю.

ПІБ автора (-ів), назва статті, назва та адреса учбового закладу необхідно надати повністю російською, українською та англійською мовами.

Назва статті друкується прописними літерами; шрифт прямий, напівжирний, кегль 12.

Назви розділів нумерують арабськими цифрами, виділяють жирним шрифтом. Відступи для назви статті, ініціалів та прізвищ авторів, відомостей про авторів, назв розділів, вступу та висновків, списку літератури: зверху – 6 пт, знизу – 3 пт.

Анотації (мовою статті, абзац 6–12 рядків, кегль 9) розміщують на початку статті, в ній має бути розміщена інформація про очікувані результати описаних досліджень (на трьох мовах).

Ключові слова (4–10 слів з тексту статті, які з точки зору інформаційного пошуку несуть змістовне навантаження) наводять мовою рукопису, через кому в називному відмінку, кегль 9.

Рисунки та таблиці (чорно-білі, контрастні) розміщуються у тексті після першого посилання у вигляді окремих об'єктів і нумерують арабськими цифрами наскрізною нумерацією за наявності більше ніж одного об'єкта. Невеликі схеми, що складаються з 3–4 елементів виконують, використовуючи вставку об'єкта Рисунок Microsoft Word. Більш складні виконують у графічних редакторах у вигляді чорно-білих графічних файлів форматів .tif, .jpg, .wmf, .cdr із розділенням 300 dpi. Рисунки мають міститися у текстовому файлі й обов'язково подаватися

окремими файлами з відповідними назвами (наприклад, рис1.jpg).

Усі елементи рисунка, включаючи написи, повинні бути згруповані. Усі написи в рисунках і таблицях мають бути виконані шрифтом Times New Roman, кегль у рисунках – 10, у таблицях – 9.

Рисунок повинен мати центрований підпис (поза рисунком), шрифт 9, відступи зверху і знизу по 6 пт. Ширина рисунка має відповідати ширині колонки (або ширині сторінки).

Формули, символи, змінні повинні бути набрані в редакторі формул **MathType**. Формули розміщують посередині рядка й нумерують за наявності посилань на них у рукописі. Шрифт – Times New Roman. Висота змінної – 10 пунктів, великих і малих індексів – 8 пт, основний математичний символ – 12 (10) пт. Змінні, позначені латинськими літерами, набирають курсивом, грецькі літери, скорочення російських слів і цифри – прямим написанням. Змінні, які є в тексті, також набирають у редакторі формул.

Список літератури вміщує опубліковані джерела, на які є посилання в тексті, укладені у квадратні дужки, друкують без абзацного відступу, кегль 9 пт, відступ зверху – 6 пт.

Після списку літератури з відступом зверху 6 пт зазначають *дату подання статті до редколегії*. Число та місяць задають двозначними числами через крапку. Розмір шрифту – 9 пт, курсив, вирівнювання по правому краю.

Резюме (Times New Roman, кегль – 10 пунктів,) подають англійською мовою: обсяг резюме до 2000 знаків (бажаний переклад). *Структура резюме: Background, Materials and methods, Results, Conclusion.*

Разом із рукописом (на аркушах білого паперу формату А4 щільністю 80-90 г/м², надрукований на лазерному принтері) необхідно подати такі документи:

1. Заяву, яку повинні підписати всі автори.
2. Акт експертизи про можливість опублікування матеріалів у відкритому друці (якщо потрібно).
3. Рецензію, підписану доктором чи кандидатом наук.
4. Відомості про авторів.
5. Електронний варіант рукопису, резюме та відомостей про авторів.
6. Зробити оплату публікації.

Необхідно також зазначити один з наступних тематичних розділів, якому відповідає рукопис:

1. Теоретичні основи інформатики та кібернетики. Теорія інтелекту.
2. Математичне моделювання. Системний аналіз. Прийняття рішень.
3. Інтелектуальна обробка інформації. Розпізнавання образів.
4. Інформаційні технології та програмно-технічні комплекси.
5. Структурна, прикладна та математична лінгвістика.
6. Дискусійні повідомлення.

СОДЕРЖАНИЕ

МАШИННОЕ ОБУЧЕНИЕ. ЭВОЛЮЦИЯ. НЕЙРОННЫЕ СЕТИ. РАСПОЗНАВАНИЕ ОБРАЗОВ

<i>Бодянский Е.В., Шафроненко А.Ю., Патлань Е.В.</i> Нечеткая кластеризация массивов данных на основе эволюционного метода оптимизации кошачьих стай	3
<i>Бодянский Е.В., Петров К.Э., Дейнеко А.А.</i> Эволюционирующая радиально-базисная нейронная сеть и экстремальное обучение ее параметров	9
<i>Москаленко В.В., Москаленко А.С., Коробов А.Г., Зарецкий М.О.</i> Модель та алгоритм навчання системи автономної навігації для малогабаритного безпілотного апарату	16
<i>Удовенко С.Г., Келембет Д.В.</i> Керування агрегатами компресорної станції з використанням нечітких інверсних моделей.....	25
<i>Руденко О.Г., Безсонов О.О., Лебедев О. Г., Романюк О. С.</i> Критерії вибору персептронної моделі для прогнозування: аналіз і практичні рекомендації щодо їх використання	31
<i>Широкопетлева М.С., Пономаренко О.А., Дудар З.В.</i> Порівняння методів прогнозування часових рядів	41
<i>Путятін Євгеній, Оробінський Павло, Ляшенко Вячеслав.</i> Виявлення вузликів легкого на цифрових медичних зображеннях	48

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ. ПРИНЯТИЕ РЕШЕНИЙ. ПОИСК ИНФОРМАЦИИ

<i>Левикін В.М., Чала О.В.</i> Метод підтримки управлінських рішень в умовах невизначеності на основі темпоральних знань	54
<i>Камінський Р.М., Шаховська Н.Б., Ладанівський Б.Т., Савків Л.Г.</i> Побудова моделі геоелектричного розрізу за результатами формальної інтерпретації даних методу зондування становленням поля в ближній зоні	60
<i>Михнова Е.Д.</i> Извлечение мультимедийных данных с нечетким соответствием.....	66
<i>Левикін В.М., Юр'єв І.О.</i> Розробка інформаційної технології управління системою надання ІТ-сервісів	72

ОНТОЛОГИЧЕСКИЕ И ЭКСПЕРТНЫЕ СИСТЕМЫ

<i>Кириченко І.В.</i> Концепція системи адаптивного електронного навчання на основі ідентифікації навчального контенту.....	79
<i>Ревенчук І.А., Перцьова К.В., Маренич О.І.</i> Програмна реалізація кластеризації пошукових запитів	86
<i>Курской Ю.С.</i> Модель измерения для систем с функцией самоорганизации	94

ВЫЧИСЛИТЕЛЬНЫЕ МЕТОДЫ. ПАРАЛЛЕЛЬНАЯ ОБРАБОТКА ИНФОРМАЦИИ

<i>Жмаєва Ю.В., Чала Л.Е.</i> Гібридний метод оцінювання складності іт проєктів	99
<i>Божко І.К., Козел Н.Б., Четвериков Г.Г.</i> Інструменти квантових обчислень	108

РАСПОЗНАВАНИЕ ОБРАЗОВ. НЕЙРОННЫЕ СЕТИ И НЕЙРОННАЯ МАТЕМАТИКА

<i>Аксак Н.Г., Бессонов А.А., Новосельцев И.В., Руденко О.Г.</i> Распознавание изменения размера и цвета изображения на основе сверточной нейронной сети	114
--	-----

Resume	120
--------------	-----

ПРАВИЛА оформлення рукописів для авторів науково-технічного журналу «БІОНІКА ІНТЕЛЕКТУ»	128
---	-----

Наукове видання

БІОНІКА ІНТЕЛЕКТУ
інформація, мова, інтелект

Науково-технічний журнал

№ 2 (91)

2018

Головний редактор — *Г. Г. Четвериков*

Відповідальний редактор — *І. Д. Вечірська*

Комп'ютерна верстка — *О. Б. Ісаєва*

Рекомендовано Вченою Радою
Харківського національного університету радіоелектроніки
(протокол № 10/6 от 19.10.2018)

Адреса редакції:

Україна, 61166, Харків-166, просп. Науки, 14,
Харківський національний університет радіоелектроніки, к. 127
тел. 702-14-77, факс 702-10-13,
e-mail: bionics19@gmail.com

Підписано до друку 30.11.2018. Формат 60 × 84 ¹/₈. Друк ризографічний.
Папір офсетний. Гарнітура Newton. Умов. друк. арк. 15,4. Обл.-вид. арк. 15,0.
Тираж 100 прим.

Віддруковано в редакційно-видавничому відділі ХНУРЕ
61166, Харків, просп. Науки, 14.