НАЦІОНАЛЬНА АКАДЕМІЯ НАУК УКРАЇНИ
НАВЧАЛЬНО-НАУКОВИЙ КОМПЛЕКС
«ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ»
НАЦІОНАЛЬНОГО ТЕХНІЧНОГО УНІВЕРСИТЕТУ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

СИСТЕМНІ ДОСЛІДЖЕННЯ ТА ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

МІЖНАРОДНИЙ НАУКОВО–ТЕХНІЧНИЙ ЖУРНАЛ

№ 1 2025

ЗАСНОВАНО У ЛИПНІ 2001 р.

РЕДАКЦІЙНА КОЛЕГІЯ:

Головний редактор

М.З. ЗГУРОВСЬКИЙ, акад. НАН України

Заступник головного редактора

Н.Д. ПАНКРАТОВА, чл.-кор. НАН України

Члени редколегії:

П.І. АНДОН, акад. НАН України
А.В. АНІСІМОВ, чл.-кор. НАН України
Х. ВАЛЕРО проф., Іспанія
Г.-В. ВЕБЕР, проф., Турція
П.О. КАСЬЯНОВ, проф., д.ф.-м.н., Україна
Й. КОРБИЧ, проф. Польща
О.А. ПАВЛОВ, проф., д.т.н., Україна
Л. САКАЛАУСКАС, проф., Литва
А.М. САЛЕМ, проф., Єгипет
І.В. СЕРГІЄНКО, акад. НАН України
Х.-М. ТЕОДОРЕСКУ, акад. Румунської Академії
Е.О. ФАЙНБЕРГ, проф., США
Я.С. ЯЦКІВ, акад. НАН України

АДРЕСА РЕДАКЦІЇ:
03056, м. Київ,
просп. Перемоги, 37, корп. 35,
ННК «ІПСА» КПІ ім. Ігоря Сікорського
Тел.: **204-81-44**; факс: **204-81-44**
E-mail: journal.iasa@gmail.com
http://journal.iasa.kpi.ua

У номері:

© *ННК «ІПСА» КПІ ім. Ігоря Сікорського*

NATIONAL ACADEMY OF SCIENCES OF UKRAINE
EDUCATIONAL AND SCIENTIFIC COMPLEX
«INSTITUTE FOR APPLIED SYSTEM ANALYSIS»
OF THE NATIONAL TECHNICAL UNIVERSITY OF UKRAINE
«IGOR SIKORSKY KYIV POLYTECHNIC INSTITUTE»

SYSTEM RESEARCH AND INFORMATION TECHNOLOGIES

INTERNATIONAL SCIENTIFIC AND TECHNICAL JOURNAL

№ 1 2025

IT IS FOUNDED IN JULY 2001

EDITORIAL BOARD:

The editor – in – chief

M.Z. ZGUROVSKY, Academician of NASU

Deputy editor – in – chief

N.D. PANKRATOVA, Correspondent member of NASU

Associate editors:

F.I. ANDON, Academician of NASU
A.V. ANISIMOV, Correspondent member of NASU
E.A. FEINBERG, Prof., USA
P.O. KASYANOV, Prof., Ukraine
J. KORBICH, Prof., Poland
A.A. PAVLOV, Prof., Ukraine
L. SAKALAUSKAS, Prof., Lithuania
A.M. SALEM, Prof., Egypt
I.V. SERGIENKO, Academician of NASU
H.-N. TEODORESCU, Academician of Romanian Academy
J. VALERO Prof., Spain
G.-W. WEBER, Prof., Turkey
Ya.S. YATSKIV, Academician of NASU

THE EDITION ADDRESS:

03056, Kyiv,
av. Peremogy, 37, building 35,
Institute for Applied System Analysis
at the Igor Sikorsky Kyiv Polytechnic Institute
Phone: **204-81-44**; Fax: **204-81-44**
E-mail: journal.iasa@gmail.com
http://journal.iasa.kpi.ua

In the issue:

Шановні читачі!

Навчально-науковий комплекс «Інститут прикладного системного аналізу» Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського» видає міжнародний науково-технічний журнал

«СИСТЕМНІ ДОСЛІДЖЕННЯ ТА ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ».

Журнал публікує праці теоретичного та прикладного характеру в широкому спектрі проблем, що стосуються системних досліджень та інформаційних технологій.

Провідні тематичні розділи журналу:

Теоретичні та прикладні проблеми і методи системного аналізу; теоретичні та прикладні проблеми інформатики; автоматизовані системи управління; прогресивні інформаційні технології, високопродуктивні комп'ютерні системи; проблеми прийняття рішень і управління в економічних, технічних, екологічних і соціальних системах; теоретичні та прикладні проблеми інтелектуальних систем підтримання прийняття рішень; проблемно і функціонально орієнтовані комп'ютерні системи та мережі; методи оптимізації, оптимальне управління і теорія ігор; математичні методи, моделі, проблеми і технології дослідження складних систем; методи аналізу та управління системами в умовах ризику і невизначеності; евристичні методи та алгоритми в системному аналізі та управлінні; нові методи в системному аналізі, інформатиці та теорії прийняття рішень; науково-методичні проблеми в освіті.

Головний редактор журналу — науковий керівник ННК «ІПСА» КПІ ім. Ігоря Сікорського, академік НАН України Михайло Захарович Згуровський.

Журнал «Системні дослідження та інформаційні технології» включено до переліку наукових фахових видань України (категорія «А»).

Журнал «Системні дослідження та інформаційні технології» входить до таких наукометричних баз даних: Scopus, EBSCO, Google Scholar, DOAJ, Index Copernicus, реферативна база даних «Україніка наукова», український реферативний журнал «Джерело», наукова періодика України.

Статті публікуються українською та англійською мовами.

Журнал рекомендовано передплатити. **Наш індекс 23918.** Якщо ви не встигли передплатити журнал, його можна придбати безпосередньо в редакції за адресою: 03056, м. Київ, просп. Перемоги, 37, корп. 35.

Завідувачка редакції **С.М. Шевченко**
Редакторка **Р.М. Шульженко**
Молодша редакторка **Л.О. Тарин**
Комп'ютерна верстка, дизайн **А.А. Патіохи**
Рішення Національної ради України з питань телебачення і радіомовлення №1794 від 21.12.2023. Ідентифікатор медіа R30-02404

Підписано до друку 28.03.2025. Формат 70×108 1/16. Папір офс. Гарнітура Times.
Спосіб друку – цифровий. Ум. друк. арк. 14,411. Обл.-вид. арк. 28,56. Наклад 70 пр. Зам. № 11/04

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Свідоцтво про державну реєстрацію: ДК № 5354 від 25.05.2017 р.
просп. Перемоги, 37, м. Київ, 03056.
ФОП Пилипенко Н.М., вул. Мічуріна, б. 2/7, м. Київ, 01014. тел. (044) 361 78 68.
Виписка з Єдиного державного реєстру № 2 070 000 0000 0214697 від 17.05.2019 р.

Dear Readers!

Educational and Scientific Complex «Institute for Applied System Analysis» of the National Technical University of Ukraine «Igor Sikorsky Kyiv Polytechnic Institute» is published of the international scientific and technical journal

«SYSTEM RESEARCH AND INFORMATION TECHNOLOGIES».

The Journal is printing works of a theoretical and applied character on a wide spectrum of problems, connected with system researches and information technologies.

The main thematic sections of the Journal are the following:

Theoretical and applied problems and methods of system analysis; theoretical and applied problems of computer science; automated control systems; progressive information technologies, high-efficiency computer systems; decision making and control in economic, technical, ecological and social systems; theoretical and applied problems of intellectual systems for decision making support; problem- and function-oriented computer systems and networks; methods of optimization, optimum control and theory of games; mathematical methods, models, problems and technologies for complex systems research; methods of system analysis and control in conditions of risk and uncertainty; heuristic methods and algorithms in system analysis and control; new methods in system analysis, computer science and theory of decision making; scientific and methodical problems in education.

The editor-in-chief of the Journal is scientific director of the Institute for Applied System Analysis at the Igor Sikorsky Kyiv Polytechnic Institute, academician of the NASU Michael Zaharovich Zgurovsky.

The articles to be published in the Journal in Ukrainian and English languages are accepted. Information printed in the Journal is included in the Catalogue of periodicals of Ukraine.

СИСТЕМНІ ДОСЛІДЖЕННЯ ТА ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

1 • 2025

ЗМІСТ

SYSTEM RESEARCH AND INFORMATION TECHNOLOGIES

1 • 2025

CONTENT

ТЕОРЕТИЧНІ ТА ПРИКЛАДНІ ПРОБЛЕМИ ІНТЕЛЕКТУАЛЬНИХ СИСТЕМ ПІДТРИМАННЯ ПРИЙНЯТТЯ РІШЕНЬ

UDC 004.85
DOI: 10.20535/SRIT.2308-8893.2025.1.01

THE HYBRID SEQUENTIAL RECOMMENDER SYSTEM SYNTHESIS METHOD BASED ON ATTENTION MECHANISM WITH AUTOMATIC KNOWLEDGE GRAPH CONSTRUCTION

D.V. ANDROSOV, N.I. NEDASHKOVSKAYA

Abstract. Sequential personalized recommendations, such as next best offer prediction or modeling demand evolution for next basket prediction, remain a key challenge for businesses. In recent years, deep learning models have been applied to solve these problems and demonstrated high feasibility. With the introduction of graph-based deep learning, it has become easier to perform collaborative filtering and link prediction tasks. The current paper proposes a new method of building a recommender system using a graph representation learning framework in combination with deep neural networks for sequence-to-sequence modeling and statistical learning for sequence-to-graph mapping. Benchmarking model performance on an online retail store visits dataset provides evidence of the method's ranking capabilities.

Keywords: recommender system, graph neural network, graph embeddings.

INTRODUCTION

First, it is necessary to coin the term "recommender system" to reach a common understanding of this notion throughout the following work. A recommendation should reflect the strong relationship between user activities and item relations [1]. As an example, a user's preference for a historical documentary is highly correlated with dedicating more time to watch another documentary or other educational program, rather than an action film [1]. These types of relations can be set explicitly by the group of experts on a level of high granularity or determined in a data-driven way on an item level.

A recommender system (RS) is a set of statistical models that considers the certain user's interaction history, knowledge about this user and each item from the available interactions, and provides relevant content (recommendation) [2]. Relevancy means an ordering relation of the user's likelihood to interact with the set of available items. Hence there exists a broad range of recommender approaches, such as non-personalized, semi-personalized, and personalized [2]. In the scope of current work, we focus on the development of the personalized RS, thus terms "recommender system" and "personalized recommender system" along with their abbreviations are interchangeable.

© *Publisher IASA at the Igor Sikorsky Kyiv Polytechnic Institute*, 2025

In recent years, personalized RS have achieved significant success across various real-world applications, including e-commerce platforms, streaming media, and online retail industries. A particularly notable area of RS application is the next best offer (NBO) recommendation task, which involves predicting which item a user will view or purchase after interacting with the platform.

NBO, also known as next best action (NBA) [3], or generalized as next-basket recommendation (NBR) [4], is a prevalent use case for any enterprise engaged in business-to-consumer (B2C) operations. Marketing teams in such enterprises have been implementing NBO/NBA projects for many years. However, many of these projects fail to deliver the expected returns [3]. This lack of performance can be caused by several factors: the use of traditional methods, failure to retrain NBO models with new sets of features (resulting in underutilization of both breadth and depth of available data), lack of effective campaign validation methods, technological deficiencies, etc.

The advent of machine learning has provided a renewed perspective on NBO/NBR. There is now an opportunity to utilize these technologies, along with comprehensive data, to enhance and optimize basket recommendations more effectively than before.

As an example, by leveraging deep learning approaches, the delivery of personalized offers and recommendations was significantly enhanced, overall demonstrating improvements in customer engagement. These enhancements can lead to increased customer satisfaction and loyalty, which ultimately drive higher sales and revenue for the business [4; 5].

RELATED WORK

The sequential recommendation task and the next best action task aim to produce such a recommended item set that satisfies the condition of relevance given the most recent user interactions. More strictly, given user $u \in U$, user session vector $s \in S$, where S is the collection of subsets of item set S, candidate item set $I \in I_d$, and the content-dependent relevancy function $R: U \times S \times I_d \to [0,1]$, $d \in \mathrm{N}$, one can formulate the sequential recommendation task as:

$$I^{*d} = \arg\max R(u, s, I^d),$$

where R could be any real-valued function, but usually the likelihood function

$$R(u, s, I^d) = p(u, s \mid I^d(\theta))$$

is applied [6].

Since user sessions could be of arbitrary length, the objective of capturing long-term dependencies between session items and candidate ones is a key one. To accomplish this task, models based on high-order Markov chains were proposed, such as context tree models (CT) [6; 7] and Markov chain similarity models [8].

Context trees construct a partition tree for each user session and then define a high-order Markov chain by traversing this tree [7].

As an alternative, the combination of high-order Markov chains with similarity-based methods, such as sparse linear methods (SLIM) and factored item similarity models (FISM) allows capturing short-term and long-term user-item and item-item relations simultaneously [8].

Besides Markov chain models, deep learning models are widely used for solving sequential recommendation tasks. Among all deep learning architectures, recurrent neural networks (RNN) are most widely chosen for accomplishing the sequential recommendation problem, especially long short-term memory networks (LSTM), and their modifications, such as bi-directional LSTM [9; 10]. Similarly to Markov chain-based hybrid approaches, LSTM networks are often used to capture session-level patterns. Thus, to enrich the proposed recommendations with learned long-term behavioural patterns, rule-based recommender design is applied [9]. Alternatively, bidirectional LSTM model was leveraged to infer recommendation rules based on current and previous user sessions [10].

Recently, a relatively new family of neural networks was applied for this task, called attention networks [11]. Attention networks are ubiquitously applied in recommendation retrieval tasks, e.g. hierarchical attention networks, that considers inputs of user-item and item-item interactions to predict further user actions [12] or stochastic self-attention networks to produce next recommendation candidates [13].

MATERIALS AND METHODOLOGY

For solving the sequential recommendation retrieval task, it is proposed to rely on graph neural network (GNN) framework.

GNNs are a family of deep neural networks capable to perform inference on data structured as graphs [14; 15; 17]. During the training process each vertex $v \in V$ of graph $G(V, E)$ updates its representation by aggregating features from its neighbors. This process can be formalized as:

$$h_v^{(k+1)} = \sigma\left(\sum_{u \in N(v)} W^{(k)} h_u^{(k)} + b^{(k)} \right),$$

where $h_v^{(k)}$ is the feature vector of vertex v at layer k, $N(v)$ represents the neighbors of v, $W^{(k)}$ and $b^{(k)}$ are the layer-specific weights and biases, and σ is a non-linear activation function.

After choosing the framework, it is necessary to somehow interpret the input sequences for the network in form of a graph.

It is intuitively understood, that user-item, user-user and item-item relations could be naturally represented as a graph $G = G(V, E, w, f)$, where $V = \langle U, I \rangle$ is a set of users and items (i.e. vertices of a graph), $E = ((u,i) \mid u \in U, i \in I)$ is an edge set, and $f : E \to w$ is a mapping from edge set to edge weights $w \in \mathrm{R}$. Thus, geometrical deep learning frameworks, such as graph neural networks (GNN) are applied to solve recommendation retrieval tasks [16].

Embedding of such graph, i.e. dense vector collection of node relations could be obtained by processing G through graph convolution network (GCN) [15] and attention network [14]. As an example, [14] combines attention mechanism with graph convolutional networks [15] to capture embeddings of user-item graph and produce next item recommendation for a given user.

GCN updates node features by aggregating features from neighboring nodes, using the following formula [15]:

$$h^{(k+1)} = \sigma\left(\widetilde{D}^{-1/2} \widetilde{A} \widetilde{D}^{-1/2} h^{(k)} W^{(k)} \right),$$

where $\widetilde{A} = A + I$ is the adjacency matrix A with added self-loops (identity matrix I); $\widetilde{D}$ is the degree matrix corresponding to $\widetilde{A}$; $h^{(k)}$ represents the node features at layer *k*; $W^{(k)}$ is the weight matrix at layer *k*; σ is an activation function.

By further considering the assumption that user preferences are non-static in long-term perspective, it is proposed to add to graph *G* (continuous) time dimension $t \in \mathbb{R}$ and obtain in such way dynamic graph $G^{(t)}(V^{(t)}, E^{(t)}, w^{(t)}, f)$. In such case, every user session at time *t* is generated by traversing the graph $G^{(t)}$. However, synthesis of such graph at a step $t = 0$ is required based on previous interactions.

For solving the task above, it is proposed to use rule mining algorithms, such as computing pointwise mutual information (PMI) [18] between each pair of bought/seen items, FP-Growth trees [19] and TF-IDF metric [20] for selecting the most relevant item combinations in the context of each session.

Pointwise mutual information for knowledge graph synthesis

Consider a set of user sessions/transactions $D = \{s_1, s_2, ..., s_m\}$, where each transaction s_j is a set of items $\{i_1, i_2, ..., i_k\}$ from a larger set of items i.

Let $p(i)$ denote the probability of item *i* occurring in a transaction, and $p(i, j)$ denote the joint probability of items *i* and *j* co-occurring in the same transaction. These probabilities can be estimated as follows:

$$p(i) = \frac{|D_i|}{|D|}; \quad p(i, j) = \frac{|D_{i,j}|}{|D|},$$

where $|D_i|$ is the number of transactions containing item *i*, and $|D_{i,j}|$ is the number of transactions containing both *i* and *j*.

The PMI between two items *i* and *j* is calculated using the formula:

$$PMI(i, j) = \log \frac{p(i, j)}{p(i)p(j)}.$$

Then by introducing the threshold ε and indicator function $\mathbf{1}_{PMI(i,j) \geq \varepsilon}$, one can construct a knowledge graph $G^{(0)} = G(V, E, \mathbf{1}_{PMI(i,j) \geq \varepsilon}, f)$.

FP-Growth algorithm for rule mining and knowledge graph synthesis

The FP-Growth (Frequent Pattern Growth) algorithm is a widely used method for mining frequent itemsets in large datasets, particularly in the context of association rule learning. The FP-Growth algorithm is more efficient and scalable compared to other rule mining algorithms like Apriori, since it does not explicitly generate itemsets, but builds FP-Growth tree for this purpose [19].

Consider a set of user sessions $D = \{s_1, s_2, ..., s_m\}$, the FP-tree *G* is defined as:

$$G = \{g \mid g = (Name(g), Supp(g), Child(g))\},$$

where each node g is a structure storing the node's name (*Name*(g)), its support value (*Supp*(g)), and a set of references to its child nodes (*Child*(g)). Items $i \in I$ are vertices of the FP-tree. The path from the root g_0 to a vertex g represents a set of items $F \subseteq I$. Let $G(i) = \{g \in G \mid g = i\}$ be the set of vertices corresponding to item $i \in I$. The support of item i is calculated as follows:

$$Supp(i) = \sum_{g \in G(i)} Supp(g).$$

To construct the FP-tree, items $i \in I$ are ordered in descending order of their support *Supp*(i). Items with support below the minimum threshold $\text{Supp}_{\min}$ are excluded from further consideration. For each element in each sorted transaction in the initial session/transaction database (TDB), the FP-tree nodes are constructed as follows:

- If a descendant of the current node exists that contains the current element, no new node is created, and the support of this descendant is incremented by one.
- Otherwise, a new descendant node is created with support initialized to one.
- The newly found or created node becomes the current node.

Thus, the levels of the FP-tree correspond to items ordered by descending support values *Supp*(i), resulting in a specific order for the set of items.

For FP-Growth trees, it is proposed to clip the maximum height of tree at 2, thus limiting the maximum frequent pattern length to be 2. Then, the graph $G^{(0)}$ is directly obtained from the frequent patterns of lengths 1 and 2, and the weight mapping is defined as follows:

$$w : e \to Supp(e).$$

Leveraging TF-IDF for knowledge graph construction

Term Frequency-Inverse Document Frequency (TF-IDF) is a metric that is commonly used to evaluate the importance of a word in a document from a collection of documents, called corpus. It is widely used in text mining and information retrieval to identify significant words in documents [20]. The term frequency TF(*t,d*) is the number of times a term t appears in a document d. It is often normalized to prevent bias towards longer documents:

$$TF(t,d) = \frac{\left|\{t : t \in d\}\right|}{\left|\{\hat{t} : \hat{t} \in d\}\right|},$$

where $\left|\{t : t \in d\}\right|$ is the number of times term t appears in document d, and $\left|\{\hat{t} : \hat{t} \in d\}\right|$ is the total number of terms in document d. The inverse document frequency IDF(*t,D*) measures how important a term is across the corpus:

$$IDF(t,D) = \log \frac{\left|D\right|}{\left|\{d \in D : t \in d\}\right|},$$

where $\left|\{d \in D : t \in d\}\right|$ is the quantity of documents that contain the term t. Note that for the $\text{IDF}(t,D)$ it is encouraged to use logarithm of inverse-document frequency, hence rare items will not receive too big values of TF-IDF score.

After obtaining the latter measures, the TF-IDF score for a term t in a document d is retrieved by multiplying the latter quantities:

$$TF - IDF(t,d,D) = TF(t,d) \times IDF(t,D) .$$

It is proposed to apply the TF-IDF metric to each pair of user sessions items $(i,j) \in s$ and by interpreting the tuple (i,j) as a single term in document u. After the following operation, by introducing the threshold ε and indicator function $\mathbf{1}_{TF-IDF(i,j)\geq\varepsilon}$, one can construct a knowledge graph $G^{(0)} = G(V,E,\mathbf{1}_{PMI(i,j)\geq\varepsilon},f)$.

Graph Neural Network training algorithm

After obtaining such graph $G^{(0)}$, it is proposed to create a deep neural network, which algorithm of fitting is shown on Fig. 1.

Algorithm 1 Recommendation retrieval network fitting algorithm (single forward and update pass)

Require: u - user vector, S_u - set of sessions, s^* - next best offer item for session s, Adj $\left(G^{(0)}\right)$ - adjacency matrix of a knowledge graph $G^{(0)}$, T - discrete time, $\alpha \in [0,1]$ - neighbor decay factor, E_I - item embedding matrix, E_U - users embedding matrix;

Adj$_0$:= Adj $\left(G^{(0)}\right)$
$e_u := E_U(u)$
for $t = 0,\ldots,T$
  for all $s \in S_u$
    $e_{true} := E(s^*)$
    $A := \varnothing$
    for all $i \in s$
      calculate sum of embeddings of all neighbors of the item i from the graph $G^{(t)}$ and call it Σ_i;
      $\Sigma_i := (1-\alpha)E_I(i) + \alpha\Sigma_i$;
      calculate self-attention over Σ_i and call it a_i;
      $A \longleftarrow a_i$
    pass the obtained matrix A to the multilayer perceptron and obtain the vector $e_{predicted}$;
    Calculate the loss between $e_{predicted}$ and e_{true}
  Update Adj$_t$, E_I and E_U according to backpropagation algorithm

Fig. 1. GNN training algorithm

The loss function in the context of the proposed algorithm is the cross-entropy function:

$$H(e_{predicted}, e_{true}) = E_{e_{predicted}}(e_{true}) ,$$

where $E_p(q)$ is the expected value of random variable q with respect to the random variable p.

For the top-k retrieval phase for the given user u and its session s, it is proposed to calculate affinity scores between embedding each item $i \in I$ and retrieved embedding representation of the pair $\langle u,s \rangle$. In the scope of the current approach, it is proposed to use the cosine similarity function:

$$\cos(i,j) = \frac{i^{\mathrm{T}} j}{\|i\| \cdot \|j\|} .$$

Warm embedding initialization

Since the proposed GNN operates with the adjacency matrix of a knowledge graph $G^{(t)}$ and user and item properties' embeddings E_U an E_I, respectively, it

is crucial to initialize such embedding matrices in a way to preserve the relationship between embeddings, depicted in the original graph. Thus, it is recommended to leverage warm embedding initialization technique before starting model training.

Warm embedding initialization is the practice of initializing the embeddings with pre-trained values rather than random values. This technique leverages prior knowledge to potentially enhance the model's performance, particularly during the initial stages of training.

One can pre-compute embeddings by using such natural language processing (NLP) models as Word2Vec [21], GloVe [22], and FastText [23].

Since it is necessary to depict the relationships formulated in knowledge graph $G^{(0)}$, it is proposed to pre-compute embeddings for $G^{(0)}$ using Node2Vec algorithm [24].

The main idea is to generate random walks on the graph and treat these walks as sentences, where each node corresponds to a lexeme, e.g. a word or item identifier.

Node2Vec generates biased random walks starting from each node to explore the graph. The algorithm introduces two parameters, p and q:

- p is a return parameter, which controls the likelihood of revisiting a node in the walk. A higher value makes it less likely to revisit the previous node, promoting exploration.
- q is an in-out parameter, which controls whether the likelihood of visiting nodes is close or far from the starting node. A higher value biases the walk towards breadth-first search, while a lower value biases it towards depth-first search.

Once the random walks are generated, Node2Vec utilizes the same Skip-Gram approach of learning lexeme embedding, as in the original Word2Vec architecture [24]. The objective is to maximize the following probability of observing a node's neighborhood given its embedding:

$$\max \sum_{u \in V} \sum_{v \in N(u)} \log p(v \mid u),$$

where V is the set of nodes, $N(u)$ is the neighborhood of node u, and $p(v \mid u)$ is the probability of node v being a neighbor of node u, modeled using the embeddings.

EXPERIMENTS AND RESULTS

Dataset description

For the experiment purposes, the dataset was selected from an e-commerce consumer electronics retail platform. The schema of the data is presented in the Table 1.

Table 1. Dataset fields description

Column name	Column type	Column description
user_id	unsigned bigint	user identifier
product_id	unsigned bigint	product identifier
category_id	unsigned bigint	item category identifier
event_type	string	interaction type (view, purchase)
user_session	base64	user session identifier

The dataset for evaluation of proposed method is a two-month snapshot from an anonymous e-commerce store database and has approximately 69 million records and 6.5 million sessions (9 GB of raw data).

Models' configurations

For the sake of qualitative assessment of experiment results, i.e. understanding the potential benefits of leveraging the proposed approach, it was decided to use a LSTM-based deep neural network as a baseline model for sequential recommendation.

For the experiment purposes, the following models' configuration was selected after performing hyperparameters tuning:

- embedding dimension size – 64;
- number of multi-layer perceptron layers – 2;
- prediction per session – 1 item, i.e. next best offer prediction;
- neighbor decay parameter α – 0.5;
- training step – 0.01;
- number of epochs – 20.

For Node2Vec algorithm, the following configuration was set:

- embedding dimension size – 64;
- number of random walk stages – 3;
- window size for skip-grams generation – 10;
- number of negative samples per skip-gram sequence – 5;
- training step – 0.01;
- number of epochs – 5.

For automated knowledge graph generation for different approaches, the given thresholds for minimum values were selected:

- for FP-Growth algorithm, the minimum support rate was set at 0.05% (which roughly correspond to be 1 in 3450 sessions);
- for TF-IDF filtering, the threshold for TF-IDF score for each item was set at 0.3;
- for PMI-based algorithm, the minimum PMI was set at the value of 12.

In order to evaluate the quality of recommendation retrieval, the most popular ranking metrics Mean Average Precision@k and Negative Discounted Cumulative Gain@k were chosen along with average training time per epoch in seconds.

Results and discussions

The results of model evaluation over the dataset are shown in Table 2.

Table 2. Model evaluation

Model	Avg. time per epoch, s	MAP@1	MAP@10	NDCG@1	NDCG@10
baseline LSTM	0.2670	0.0389	0.0937	0.0388	0.1217
proposed GNN w. PMI	610.02	0.1442	0.2551	0.1444	0.2988
proposed GNN w. TF-IDF	115.08	0.0844	0.1622	0.0845	0.1960
proposed GNN w. FP-Growth	53.920	0.0399	0.0887	0.0378	0.1150

Overall, GNNs performs significantly better than baseline LSTM, despite their training time being slower more than 200 times. Such slowdown is caused by traversing each time the synthesized knowledge graph $G^{(t)}$. From the comparison above one can decide to stick either with TF-IDF option which provides the compromise in terms of both accuracy and learning time, or experimenting with PMI clipping for achieving the 4x increase in recommendation retrieval accuracy in the scope of next best action prediction. Using FP-Growth algorithm for automatic knowledge graph construction wasn't beneficial nor in accuracy of predictions, offering the same results as the baseline LSTM model, but yet being much slower.

The evolution of the MAP@1 metric for all models is depicted on Fig. 2. The evolution of the MAP@10 metric for all models is depicted on Fig. 3.

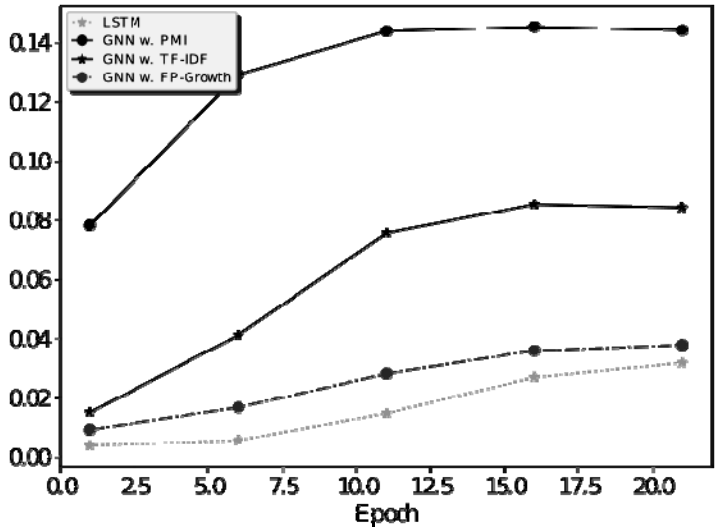


Fig. 2. MAP@1 per epoch

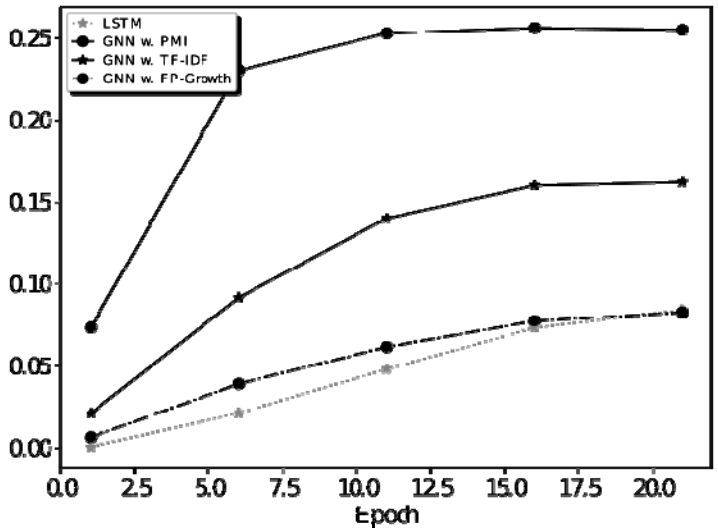


Fig. 3. MAP@10 per epoch

On the other hand, the evolution of the NDCG@1 and NDCG@10 metric for all models are depicted on Fig. 4 and Fig. 5, respectively.

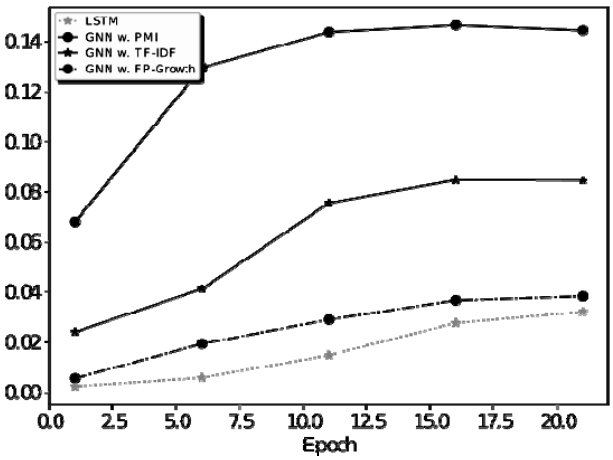


Fig. 4. NDCG@1 per epoch

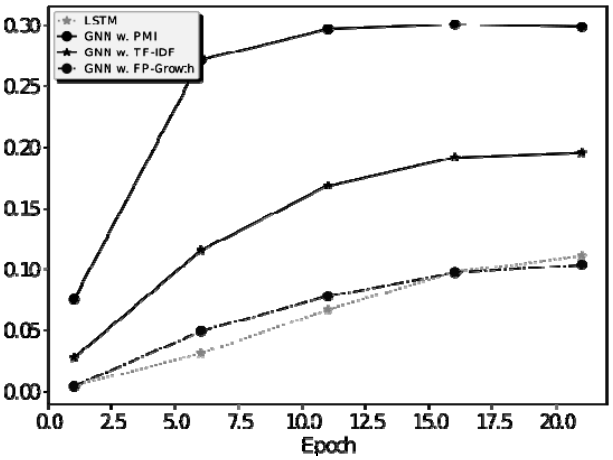


Fig. 5. NDCG@10 per epoch

By examining the results one can conclude that the proposed approach is feasible to use in the context of the next-best-offer recommendation.

CONCLUSIONS

In the current research the problem of personalized sequential recommendations was addressed. By conducting a literature review on a given topic, it was discovered that by leveraging hybrid approaches, including Context Trees and high-order Markov chains, the problem of sequential recommendation was solved, but with a severe limitation — order of items to consider depends on order of Markov chain.

By introduction of deep learning and leveraging the ability of (artificial) neural networks to be a universal approximators, to solve the given problem, deep recurrent neural networks, namely LSTMs were used, since of their ability of capturing temporal dependencies of arbitrary lengths.

However, RNNs are requiring the data to be only in the form of a sequence, thus discarding the ability to naturally represent user-user, item-item and user-item relations as a network. Such a way of representing these relations and capturing them as a knowledge graph, using data mining algorithms, is a key idea of the proposed research.

It was also proposed by the current paper to slightly change the mechanics of an Attention mechanism, to allow capture not only the studied sequence, but examine associations, presented in a knowledge graph, for each item in the sequence.

By examining different techniques to capture associative similarities between sets of items, namely TF-IDF, FP-Growth trees and PMI metric, constructing a deep graph neural network with proposed modification of Attention mechanism, and leveraging Node2Vec algorithm to retrieve the embedding matrix of a knowledge graph, new method for hybrid sequential recommender system design was introduced.

By benchmarking the models on an anonymous retailer dataset, the evidence of high precision of GNN models in terms of MAP@*k* and NDCG@*k* for next-best offer recommendation was obtained.

REFERENCES

1. C. Aggarwal, *Recommender Systems*: The Textbook. Springer, 2016.
2. K. Falk, *Practical Recommender Systems*. Manning, 2019.
3. A. Rasool, *Next Best Offer (NBO) / Next Best Action (NBA)- why it requires a fresh perspective?* 2019. [Online]. Available: https://www.linkedin.com/pulse/next-best-offer-nbo-action-nba-why-requires-fresh-azaz-rasool/
4. S. Wang et al., "Modeling User Demand Evolution for Next-Basket Prediction," IEEE Transactions on Knowledge and Data Engineering, vol. 35(11), pp. 11585–11598, 2023. doi: https://doi.org/10.1109/TKDE.2022.3231018
5. K.A. Eliyahu, *Achieving Commercial Excellence through Next Best Offer models*. [Online]. Available: https://www.linkedin.com/pulse/achieving-commercial-excellence-through-next-best-offer-kisliuk/
6. S. Wang et al., "Sequential Recommender Systems: Challenges, Progress and Prospects," in *IJCAI 2019*, 2019. doi: https://doi.org/10.24963/ijcai.2019/883
7. F. Garcin, C. Dimitrakakis, and B. Faltings, "Personalized News Recommendation with Context Trees," in *Proceedings of the 7th ACM conference on Recommender systems, ACM, 2013*. doi: https://doi.org/10.1145/2507157.2507166
8. R. He, J. McAuley, "Fusing Similarity Models with Markov Chains for Sparse Sequential Recommendation," in 2016 *IEEE 16th International Conference on Data Mining (ICDM), Barcelona, 2016*, pp. 191–200. doi: 10.1109/ICDM.2016.0030
9. Q. Xia et al., "Modeling Consumer Buying Decision for Recommendation Based on Multi-Task Deep Learning," in *CIKM '18*, 2018. doi: https://doi.org/10.1145/3269206.3269285
10. C. Zhao, J. You, X. Wen, and X. Li, "Deep Bi-LSTM Networks for Sequential Recommendation," *Entropy* (*Basel*), 22(8), 870, 2020. doi: https://doi.org/10.3390/e22080870
11. A. Vaswani et al., "Attention Is All You Need," in *Advances in Neural Information Processing Systems (NIPS 2017)*, pp. 5998–6008, 2017. doi: https://doi.org/10.48550/arXiv.1706.03762
12. H. Ying et al., "Sequential Recommender System based on Hierarchical Attention Network," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, Stockholm, 13–19 July 2018*, pp. 3926–3932. doi: https://doi.org/10.24963/ijcai.2018/546
13. Z. Fan et al., "Sequential Recommendation via Stochastic Self-Attention," in *Proceedings of the ACM Web Conference 2022 (WWW '22), 2022*. doi: https://doi.org/10.1145/3485447.3512077
14. T. Hekmatfar, S. Haratizadeh, P. Razban, and S. Goliaei, "Attention-Based Recommendation On Graphs," *arXiv*. 2022. [Online]. Available: https://arxiv.org/abs/2201.05499
15. T.N. Kipf, M. Welling, "Semi-Supervised Classification with Graph Convolutional Networks," *arXiv*, 2016. [Online]. Available: https://arxiv.org/abs/1609.02907
16. N.I. Nedashkovskaya, D.V Androsov, "Multicriteria evaluation of recommender systems using fuzzy analytic hierarchy process," PREPRINT (Version 1), available at *Research Square*. doi: https://doi.org/10.21203/rs.3.rs-3228086/v1

17. Z. Wu et al., "A Comprehensive Survey on Graph Neural Networks," *IEEE Transactions on Neural Networks and Learning Systems*, 32(1), pp. 4–24, 2020. doi: https://doi.org/10.48550/arXiv.1901.00596
18. K.W. Church, P. Hanks, "Word association norms, mutual information, and lexicography," *Comput. Linguist*, vol. 16(1), pp. 22–29, 1990.
19. J. Han, J. Pei, Y. Yiven, "Mining Frequent Patterns without Candidate Generation," *ACM SIGMOD Record*, vol. 29(2), pp. 1–12, 2000. doi: 10.1145/342009.335372
20. S. Robertson, "Understanding inverse document frequency: on theoretical arguments for IDF," *Journal of Documentation*, vol. 60(5), pp. 503–520, 2004. doi: 10.1108/00220410410560582
21. T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," *International Conference on Learning Representations (ICLR), May 2-4, 2013, Scottsdale, Arizona, USA, 2013*. doi: https://doi.org/10.48550/arXiv.1301.3781
22. J. Pennington, R. Socher, and C.D. Manning, "GloVe: Global Vectors for Word Representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 25-29 October 2014, Doha, Qatar*, pp. 1532–1543, 2014. doi: 10.3115/v1/D14-1162
23. P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching Word Vectors with Subword Information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017. doi: 10.1162/tacl_a_00051
24. A. Grover, J. Lescovec, "node2vec: Scalable Feature Learning for Networks," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, August 13-17, 2016, San Francisco, CA, USA*, pp. 855–864, 2016. doi: https://doi.org/10.1145/2939672.2939754

Received 06.06.2024

INFORMATION ON THE ARTICLE

Nadezhda I. Nedashkovskaya, ORCID: 0000-0002-8277-3095, Educational and Research Institute for Applied System Analysis of the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Ukraine, e-mail: nedashkovskaya.nadezhda@lll.kpi.ua

Dmytro V. Androsov, ORCID: 0009-0001-1330-1473, Educational and Research Institute for Applied System Analysis of the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Ukraine, e-mail: androsovdmitry80@gmail.com

ГІБРИДНИЙ МЕТОД ПОБУДОВИ СЕКВЕНЦІАЛЬНИХ РЕКОМЕНДАЦІЙНИХ СИСТЕМ, ЗАСНОВАНИЙ НА АВТОМАТИЧНОМУ СИНТЕЗІ ГРАФІВ ЗНАНЬ ТА МЕХАНІЗМІ УВАГИ / Д.В. Андросов, Н.І. Недашківська

Анотація. Послідовні персоналізовані рекомендації, такі як прогнозування наступної найкращої пропозиції або моделювання еволюції попиту для прогнозування наповнення кошика покупок, залишаються ключовим завданням для бізнесу. Останнім часом, задля вирішення цих проблем застосовувалися ланцюги Маркова та моделі глибокого навчання, що прогнозували послідовності взаємодії користувачів із товарами, демонстрували високу ефективність. Проте ключовим недоліком таких моделей було неуніфіковане подання наборів даних для довгострокового та короткострокового прогнозуваннння вподобань. З появою архітектур глибокого навчання на графах та можливості їх застосування одночасно в задачах колаборативної фільтрації та прогнозування зв'язків між об'єктами, розвиток рекомендаціних систем отримав новий поштовх. Пропоновано новий метод розроблення гібридних рекомендаційних систем, який поєднує навчання подань графів з глибокими нейронними мережами для моделювання та прогнозування послідовностей, з метою розв'язання задачі видачі послідовних персоналізованих рекомендацій. Отримані результати оцінювання продуктивності моделі на основі набору даних відвідувань та купівель в інтернет-магазині доводять можливість ранжування та потенціал для впровадження бізнесами у сфері роздрібної торгівлі.

Ключові слова: рекомендаційна система, графова нейронна мережа, подання графів.

UDC 004.02, 004.67, 004.891.3, 616.24-002.5-02:316.342.6:316.62:314(477)
DOI: 10.20535/SRIT.2308-8893.2025.1.02

STUDYING THE RELATIONSHIP BETWEEN TUBERCULOSIS AND SOCIOECONOMIC, MEDICAL, AND DEMOGRAPHIC FACTORS IN UKRAINE

D.V. NEVINSKYI, D.I. MARTJANOV, I.O. SEMIANIV, Y.I. VYKLYUK

Abstract. Ukraine is currently experiencing a new, ongoing tuberculosis offensive. Our study analyzes the impact of various socioeconomic and medical factors, including the number of specialized hospitals, fluoroscopic examinations of the population, the number of healthcare workers, the level of alcohol and drug abuse, and others, on the prevalence of tuberculosis among different demographic groups in Ukraine. Artificial intelligence methods made it possible to identify key factors contributing to the growth or decline in tuberculosis incidence. The results of the SHAP (SHapley Additive exPlanations) analysis, which offers a methodology for interpreting complex machine learning models, shows the most important factors that influence the incidence of tuberculosis in Ukraine. The sensitivity analysis provided more important and detailed information, which confirmed the results of the SHAP analysis.

Keywords: artificial intelligence, tuberculosis, incidence, socio-demographic factors, medical factors, demographic factors.

RELEVANCE OF THE WORK

Currently, Ukraine is experiencing a new, regular offensive of tuberculosis. In the current conditions of development of Ukrainian society, one of the important problems that needs to be addressed is the spread of tuberculosis, a disease that is closely related to socioeconomic, medical and demographic factors [1]. The fact is that tuberculosis, as a social disease, is a mirror of socioeconomic well-being in the country [2].

The analysis of the ways of spreading, negative consequences for public health and other aspects of the spread of tuberculosis has long been the focus of research [3]. At the same time, the study of socioeconomic, medical and demographic reasons that influence the spread of tuberculosis in Ukrainian society remains an unexplored area of research.

Only a medical approach to the analysis of socio-economic, medical and demographic factors that affect the incidence of tuberculosis in Ukraine is insufficient in timely forecasting the prospects for the development of the tuberculosis epidemic and developing an appropriate plan to counter its challenges, as a result of which the incidence of tuberculosis remains extremely threatening not only to the life and health of our citizens, but also gives reason to consider this situation as a threat to the WHO European region [4].

Therefore, we used mathematical analysis with the use of artificial intelligence to establish the relationship between tuberculosis and socioeconomic, medical, and demographic factors in Ukraine.

© *Publisher IASA at the Igor Sikorsky Kyiv Polytechnic Institute*, 2025

ANALYSIS OF RESEARCH

Today, scientists are conducting research and modeling of the spread of tuberculosis [5]. Another study highlights how socioeconomic conditions contribute to the spread of tuberculosis [6]. In [7], the authors analyze how access to health care affects the effectiveness of tuberculosis control. They also consider how demographic changes affect the incidence of tuberculosis [8]. An overview of progress in the use of artificial intelligence (AI) in medicine [9]

The use of artificial intelligence in tuberculosis research is becoming increasingly popular due to its ability to analyze large data sets, identify complex relationships, and predict epidemiological trends. In particular, [10] uses various machine learning algorithms to predict the incidence of tuberculosis, which allows for high accuracy predictions and identification of regions at high risk of disease spread [11]. The authors have developed a deep learning-based system to automatically detect major chest diseases, including tuberculosis, in X-rays [12]. Although this study focuses on COVID-19, the methodologies and technologies they use can be adapted to monitor and predict the spread of tuberculosis, demonstrating the potential of AI in global epidemic management [13]. In this review, the authors discuss the possibilities of machine learning in the medical field, including its ability to integrate and analyze large amounts of data on socioeconomic factors to better understand their impact on the spread of tuberculosis.

However, there are currently no studies that examine the complex impact of various factors on the spread of tuberculosis based on artificial intelligence technology.

Therefore, **the purpose of our work** is to analyze the impact of various socioeconomic, medical, and demographic factors on the incidence of tuberculosis among the urban and rural population of Ukraine, in order to identify key factors that can contribute to the development of more effective strategies for controlling and preventing the disease.

MATERIALS AND METHODS

Description of the dataset. The dataset for analyzing the impact of various socioeconomic, medical, and demographic factors on tuberculosis incidence consists of the above fields and contains 400 records. The data was collected over the last 16 years and covers all regions of Ukraine. This dataset includes information on the number of specialized hospitals, the number of fluoroscopic examinations per 100.000 population, vaccination data, the number of bacterial isolators, the incidence among urban and rural residents, and the percentage of different demographic groups (workers, employees, healthcare workers, students, pupils, pensioners, unemployed, persons returned from prison, persons without permanent residence, private workers).

The dataset also includes indicators reflecting the level of alcohol abuse and drug use, the incidence of doctors in specialized hospitals per 10 thousand health care workers, HIV/TB rates per 100 thousand people, cases of resistant TB, treatment failure, interrupted treatment, patients dropped out of follow-up, treat-

ment outcomes for relapses and multidrug-resistant tuberculosis (MDR-TB), and the number of surgical interventions (lung and extrapulmonary TB surgeries).

Research methodology. The research consists of the following steps:

1. **Correlation analysis.** At the first stage of the study, correlation analysis is used to identify statistical relationships between various factors (e.g., number of hospitals, healthcare workers, vaccination rates) and TB incidence. This allows us to determine which variables have a potential impact on the prevalence of the disease. The use of Pearson correlation coefficient helps to assess the strength and direction of the interaction between variables.

2. **Testing different models with cross-validation.** The next step is to test different machine learning models, such as linear regression, decision trees, random forest, kNN, support vector machine (SVM), adaptive boosting (AdaBoost), stochastic gradient descent, back propagation neural networks. Cross-validation is used to check the stability of models, in our case through 5-fold cross-validation, where the data is divided into 5 subsets and the model is tested 5 times, each time using one subset as a test set and the others as training data. The consistency of the cross-validation results served as an indicator of the presence of overfitting in these machine learning models and the selection of their hyperparameters. The following hyperparameters were selected: Linear Regression — Elastic Net regularization $L1/L2 = 50/50$, Decision Trees and Random Forest — maximum depth $d = 5$, Nearest Neighbors Method — selecting the optimal value of the nearest neighbors — $k = 5$, Support Vector Machine (SVM) — selecting the parameters $C = 0.8$ and $\gamma = 0.1$, Adaptive Boosting — Limiting the number of base models: $n_$estimators = 50, Stochastic Gradient Descent (SGD) — Elastic Net reegularozation $L1/L2 = 50/50$, Backpropagation — regularization $= 0.001$ and Dropout $d = 0.2$.

3. **Building an ensemble of models.** Based on the obtained models, an ensemble is built that combines the forecasts of the best models to improve the accuracy and reliability of the results. The study used a stacking-based ensemble, which allowed us to consider various aspects of the data and reduce the variability of the forecast.

4. **Analysis of an ensemble of models.** This analysis evaluates the overall performance of the model ensemble. It evaluates how the combination of models performs compared to individual models, including an assessment of accuracy, specificity, and other fit metrics.

5. **Determining the importance of factors.** Factor importance analysis is conducted to identify the key variables that have the greatest impact on morbidity. This may include the use of importance metrics provided by the algorithms that are included in the ensemble model.

6. **Sensitivity analysis.** The final step of sensitivity analysis tests the robustness of the model ensemble to changes in the data or in the model parameters. This involves varying key parameters and assessing the impact of these changes on the model results.

The study was conducted in the Orange environment. The data flow diagram is shown in Fig. 1.

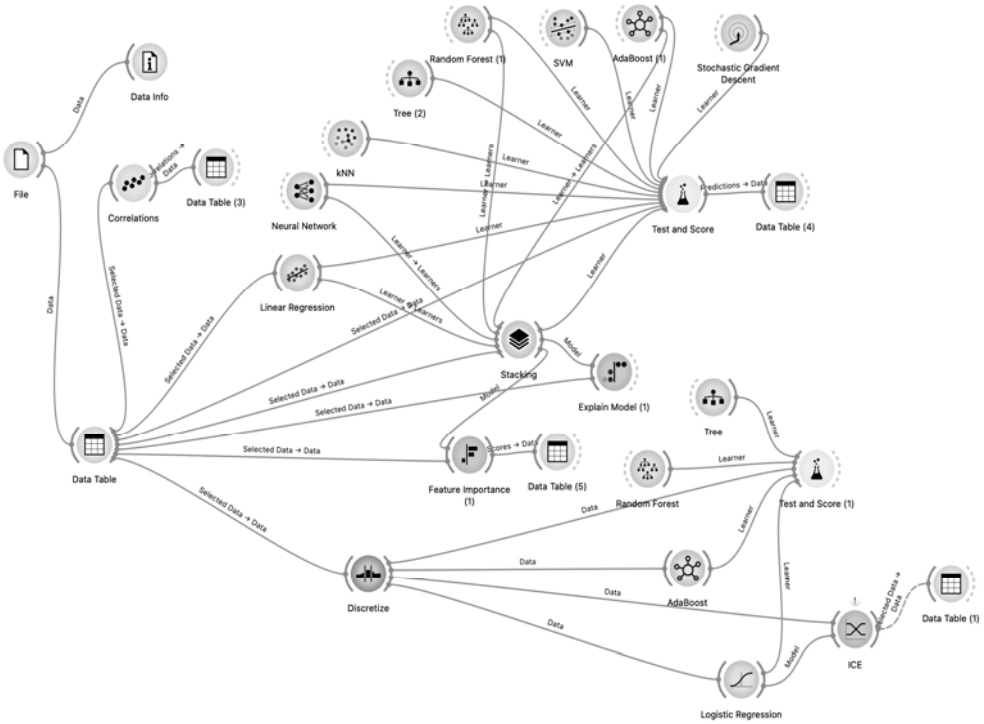


Fig. 1. The scheme of information flows of the study

RESULTS OF THE STUDY

Correlation analysis. Table 1 shows the results of the correlation analysis, which presents the values of the coefficients of determination R^2 for various factors that may affect the incidence of tuberculosis. The coefficient of determination R^2 measures the proportion of variation in a given variable that can be explained by the independent variables in the model. The key conclusions from the table include:

- **Bacterial excretion** has the highest coefficient $R^2 = 0.641$, indicating a strong relationship between the frequency of bacterial excretion in the population and the incidence of tuberculosis.
- **HIV/TB** (the ratio of HIV and TB incidence per 100.000 population) also has a significant coefficient of $R^2 = 0.542$, which emphasizes the link between these two diseases.
- **Fluoroscopic examinations** have a coefficient of $R^2 = 0.501$, which indicates the importance of regular medical examinations in detecting and controlling tuberculosis, especially in risk groups.
- **Physician morbidity** and **surgical treatment** also show relatively high R^2 values, which may reflect the impact of non-compliance with infection control conditions and the importance of surgery in some cases as an additional treatment method.
- The low R^2 coefficients for variables such as **alcohol** and **drug abuse and demographic groups** (e.g., **pensioners**, **students**, **workers**) indicate a less pronounced direct impact of these factors on morbidity compared to medical and epidemiological factors.

Table 1. Results of the correlation analysis

Factor	R^2
Bacterial excretion	0.641
HIV/TB (per 100 thousand)	0.542
Fluoroscopic examinations of the population (per 100 thousand)	0.501
Morbidity rate of doctors (per 10 thousand medical staff)	0.48
Surgical treatment (easy number of operations)	0.468
Resistant TB	0.466
Interrupted treatment	0.433
Unsuccessful treatment	0.387
Relapse rate (interrupted treatment)	0.379
Relapse rate (cured)	0.378
Expelled.	0.369
Non-operational (% of total)	0.364
MLS-TV (withdrawn)	0.335
Surgical treatment (total number of operations)	0.317
Relapse rate (unsuccessful treatment)	0.311
Recidivism rate (discharged)	0.308
Pensioners (% of total)	-0.294
Number of hospitals	0.216
Vaccinations carried out	0.2
R-treatment of MDR-TB (interrupted treatment)	0.146
Drug use (% of total)	0.118
Without a permanent place of residence (% of total)	-0.111
Alcohol abuse (% of total)	-0.107
Employees (% of total)	-0.091
MDR-TB treatment (failed treatment)	0.076
Private employees (% of total)	-0.056
Students (% of total)	0.052
Employees (% of total)	-0.047
People who returned from places of deprivation of liberty (% of the total)	-0.019
Students (% of total)	-0.01
Medical workers (% of total)	0.002

Testing different models by cross-validation. The next step was to analyze the performance of the above machine learning models in the context of tuberculosis incidence prediction using the 5-fold cross-validation method. The main parameters evaluated include the mean square error (MSE), root mean square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), and coefficient of determination (R^2). The results of the study are presented in Table 2.

Table 2. Results of testing different machine learning model

Machine Learning Model	MSE	RMSE	MAE	MAPE	R^2
Linear Regression	108.04	10.39	7.87	0.14	0.71
Neural Network	111.52	10.56	7.54	0.14	0.70
kNN	265.11	16.28	11.93	0.26	0.29
Decision Tree	191.64	13.84	9.42	0.18	0.49
Random Forest	80.92	9.00	6.64	0.13	0.78
SVM	255.39	15.98	11.69	0.25	0.32
AdaBoost	72.49	8.51	6.22	0.12	0.81
Stochastic Gradient Descent	132.32	11.50	8.52	0.16	0.65
Stacking	62.99	7.94	5.78	0.11	0.83

As can be seen from the table, the linear regression performed satisfactorily with a coefficient of determination of $R^2 = 0.71$, indicating that the model is moderately adequate for this data set. Although the RMSE and MSE are relatively high, this indicates potential deviations in predictions, especially when considering large and complex data. Neural networks are almost equal to linear regression in terms of R^2, but require more careful tuning and computational resources. This model can be particularly sensitive to overfitting due to the complexity of the model structure.

The KNN model showed the worst results with $R^2 = 0.29$, which indicates low prediction accuracy. The high MSE and RMSE values emphasize that the model does not work efficiently with the data, possibly due to insufficient data for training or mismatched model parameters. Decision trees showed average results $(R^2 = 0.49)$. This model is sensitive to changes in the data, and can create complex structures that lead to overfitting, especially in cases where tree pruning techniques are not applied.

Random Forest showed one of the best results $(R^2 = 0.78)$, demonstrating high accuracy and reliability of predictions. It efficiently manages overfitting and has a high classification and regression capability, thanks to the ensemble approach.

The support vector machine (SVM) method showed low efficiency $(R^2 = 0.32)$ with high MSE and RMSE, which may indicate the need to refine and optimize the kernel parameters to improve prediction.

Adaptive boosting (AdaBoost) showed the highest performance $(R^2 = 0.81)$ among all the models considered, with the lowest MSE and RMSE, indicating high accuracy and reliability. This model adapts well to different datasets, improving accuracy by consistently reducing the weight of errors in the training data

Stochastic Gradient Descent performed moderately well $(R^2 = 0.65)$, showing potential in situations where large datasets need to be optimized quickly. However, the method can be sensitive to noise in the data and requires careful tuning of the learning rate.

Building an ensemble of models. Based on the analysis of the performance of various machine learning models, it is proposed to create an ensemble of mod-

els using Stacking method, including the following estimators: linear regression, neural network, adaptive boosting (AdaBoost), and random forest. These models were chosen because of their high performance and complementarity in solving forecasting problems.

Stacking technology has the following advantages:

1. Complementarity of models: Random Forest and AdaBoost have demonstrated high accuracy in prediction, but they may tend to overlearn or bias in certain scenarios. Linear regression, while less accurate, offers stability and good generalization. Neural networks work effectively with non-linear relationships in data. Stacking allows you to combine their predictions, which can improve the overall accuracy and reliability of forecasting.

2. Reduce variability and errors: Stacking uses a linear model to stack predictions from the underlying models. This not only preserves the strengths of each model, but also effectively reduces the errors that can occur when using any single model.

3. Improved generalization: Using the predictions of different models as input to a "metamodel" in stacking allows the ensemble to generalize more effectively on unseen data, which is critical for real-world forecasting tasks.

Analysis of an ensemble of models. As can be seen in Table 2, the Stacking model shows the best performance among all the methods considered:

- R^2: The highest among all models, 0.83, indicating that the Stacking model explains approximately 83% of the variation in response across the dataset, outperforming its closest competitor (AdaBoost) by 0.02 points.
- MSE and RMSE: Stacking has the lowest MSE (6299) and RMSE (794), which indicates lower overall prediction errors compared to other models.
- MAE and MAPE: Also the lowest among all the models considered (MAE $= 578$ and MAPE $= 0.011$), which emphasizes the high accuracy of the forecasts created by the Stacking model.

Compared to individual models such as AdaBoost and Random Forest, which also showed high accuracy rates, Stacking provides an additional improvement in accuracy and stability. This demonstrates the power of a combined approach that considers different aspects of the data and the problem, while reducing the likelihood of overfitting that can occur with individual models.

Thus, stacking turned out to be the most efficient method among the analyzed ones, showing the highest performance across all evaluation criteria. This makes it an ideal candidate for use in real-world environments where high accuracy and reliability of forecasts are important.

Determining the importance of factors. The analysis of the importance of the factors, performed using a stacked model, allows us to identify the key variables that have the greatest impact on the incidence of tuberculosis. Assessment of the importance of each factor in the model allows us to better understand the dynamics of morbidity and optimize intervention strategies. Table 3 show the results of the importance of factors based on the stacked model.

As we can see from the data, the rate of bacterial shedding differs significantly from the others, which is fully supported by the literature [14]. It seems somewhat unexpected that the surgical treatment rate was among the factors with a significant impact. According to the current global TB treatment protocols, surgical treatment is indicated only in certain cases and is no longer

used as often as it used to be. All other factors undoubtedly have an impact on the incidence of tuberculosis, which is confirmed by medical research data [1].

Table 3. Importance of factors in the stacking model

Feature	Importance
Bacterial excretion	0.405
Fluoroscopic examinations of the population (per 100 thousand)	0.059
Surgical treatment (easy number of operations)	0.026
MDR-TB treatment (failed treatment)	0.020
Expelled	0.016
Morbidity rate of doctors (per 10 thousand medical staff)	0.015
Resistant TB	0.015
MLS-TV (withdrawn)	0.014
HIV/TB (per 100 thousand)	0.011
People who returned from places of deprivation of liberty (% of the total)	0.009
Non-operational (% of total)	0.008
Alcohol abuse (% of total)	0.007
Pensioners (% of total)	0.007
R-treatment of MDR-TB (interrupted treatment)	0.006
Unsuccessful treatment	0.006
Vaccinations carried out	0.006
Number of hospitals	0.005
Relapse rate (unsuccessful treatment)	0.005
Without a permanent place of residence (% of total)	0.005
Relapse rate (cured)	0.004
Surgical treatment (total number of operations)	0.004
Students (% of total)	0.004
Recidivism rate (discharged)	0.004
Relapse rate (interrupted treatment)	0.004
Employees (% of total)	0.004
Medical workers (% of total)	0.003
Private employees (% of total)	0.003
Interrupted treatment	0.003
Drug use (% of total)	0.003
Employees (% of total)	0.002
Students (% of total)	0.002

As one can see from the results:

- **Bacterial shedding** is the most important factor (0.405), indicating a high level of influence on TB incidence. This emphasizes the need to focus on controlling the spread of bacterial shedding, as this indicator correlates with high incidence rates.
- **Fluoroscopic examinations** have the second most important indicator (0.059). This confirms the importance of regular medical examinations, especially

for risk groups, in detecting and preventing the disease, which allows for early identification of new cases of tuberculosis.

- **Surgical treatment** and **outcomes** for **MDR-TB** are also important variables. This reflects the importance of additional surgical interventions, in addition to chemotherapy, and the importance of successful treatment in the context of fighting resistant forms of TB and the need to improve and optimize treatment strategies.
- **The incidence of** physician-associated and **resistant TB** is also relatively high, which may indicate the risk of non-compliance with infection control measures in healthcare facilities and challenges associated with the spread of resistant forms of TB.
- Less important, but still significant, variables include **HIV/TB comorbidity**, **reentry from prison**, and **socioeconomic indicators** such as **alcohol abuse**. These variables indicate the complexity of the links between social conditions and disease, which requires a comprehensive approach to community health.

Sensitivity analysis. Sensitivity analysis and SHAP analysis are important tools for analyzing the spread of tuberculosis, which help to better understand the mechanisms of the model and its response to changes in input data.

SHAP (SHapley Additive exPlanations) analysis offers a methodology for interpreting complex machine learning models. It allows one to identify the contribution of each factor to the model's prediction, which is crucial for transparency and clarity in medical and policy decision-making. In the context of tuberculosis, SHAP analysis helps to identify which factors are most important for disease incidence, which can help to develop targeted interventions.

Sensitivity analysis is used to assess the stability and reliability of predictive models by determining how they respond to changes in input parameters. In the context of this study, this analysis allows us to test how small changes in factors, such as the number of medical examinations or demographic composition, can affect the model's conclusions. This is critical to ensure the accuracy and reproducibility of the results, especially in settings where models may be used to support public health decisions.

Fig. 2 shows the SHAP analysis of the stacking model. The graph shows the most important factors of the model. Each point on the graph corresponds to a SHAP value for each factor. The SHAP value is a measure of how much each

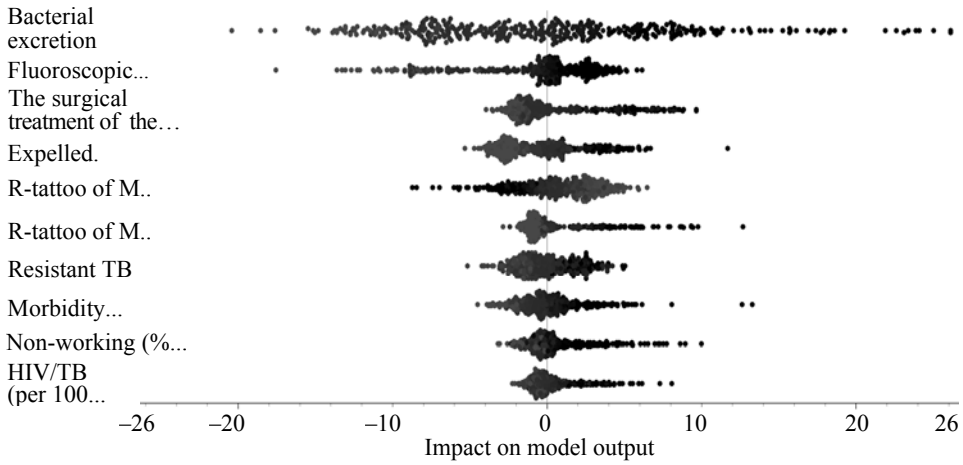


Fig. 2. SHAP analysis of the stacking model

factor influences the model outcome. A higher SHAP value (greater deviation from the center of the graph) means that the factor value has a greater impact on the prediction for the selected class. Positive SHAP values (points to the right of the center) are the values of features that influence prediction. The SHAP value shows how much the feature value affects the predicted value from the average prediction. The colors represent the value of each factor. Red represents a higher texture value and blue represents a lower value. The color range is determined based on all the values in the dataset for the object. As you can see from Fig. 2, the results of the SHAP analysis fully confirm the importance of the factors.

Sensitivity analysis provides more important information. Figs. 3–5 show the dependence of changes in tuberculosis incidence on changes in the most important factors.

All graphs are individual sensitivity plots for each individual row in the dataset. The yellow graph shows the average value of all records.

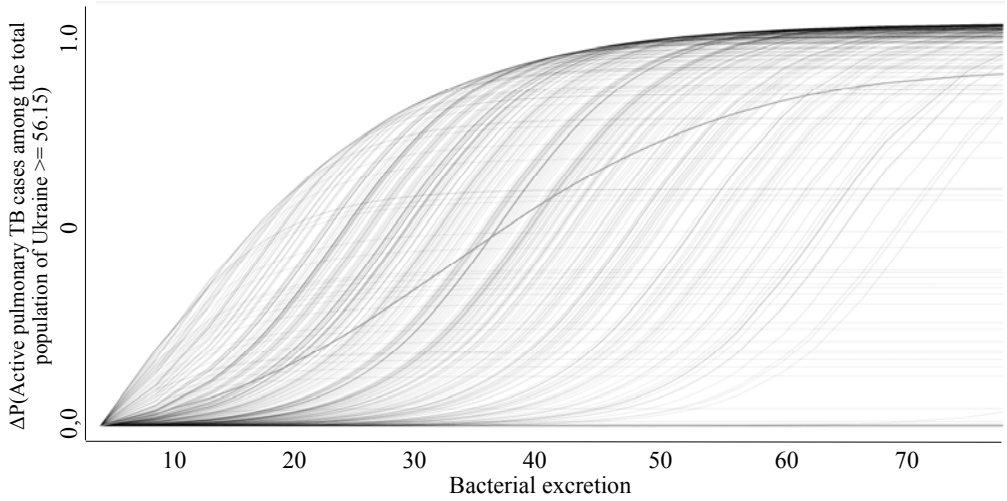


Fig. 3. The dependence of changes in tuberculosis incidence on bacterial shedding per 100 thousand people

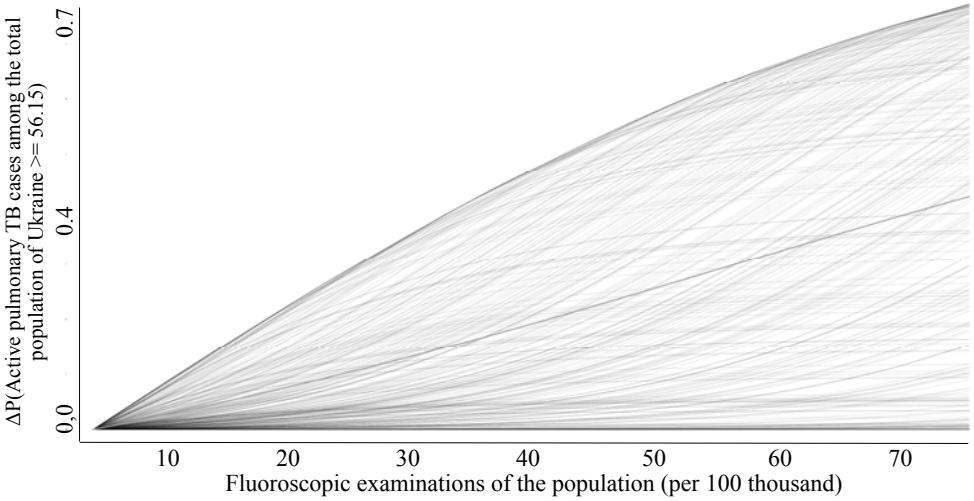


Fig. 4. The dependence of changes in tuberculosis incidence on bacterial shedding per 100 thousand people

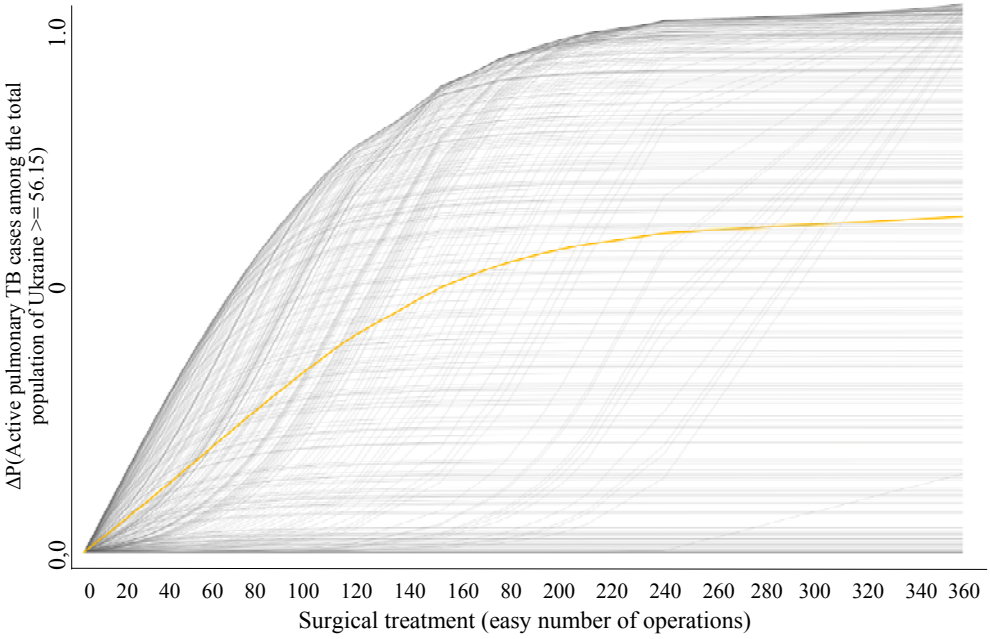


Fig. 5. Analysis of the sensitivity of tuberculosis incidence to the rate of surgical treatment of pulmonary tuberculosis (number of surgeries)

The logarithmic growth of the incidence shows a rapid increase against the background of an increase in bacterial shedding, but then a stable saturation level is determined. From a medical point of view, this is explained by the fact that active bacterial shedders quickly infect their contacts, and then the process of infection spread is suspended until new active patients start infecting others. This points to the importance of the efforts of health care systems in developed countries, which are primarily aimed at identifying and starting treatment of patients with bacterial excretion as soon as possible. Such patients pose a danger to others, often without realizing it. One undetected patient can infect 10 to 15 people who are in close daily contact with him or her. Thus, the result fully confirms the WHO epidemiological studies.

The linear increase in morbidity against the background of the fluoroscopic examination rate demonstrates a gradual, steady increase in the number of active TB patients. The importance of fluoroscopic examinations is confirmed by the latest WHO recommendations, especially the statement that fluoroscopic examinations of the population should focus on high-quality screening of risk groups rather than on random screening of everyone. Since Ukraine still has a quite high incidence of tuberculosis, and the number of internally displaced persons reached 4.9 million during the war period, all these people can be considered a risk group. The importance of regular fluoroscopic preventive examinations has been confirmed by numerous studies [15], and the fact that the sensitivity analysis ranked this indicator second in terms of its impact on morbidity is logical and understandable for the medical community.

An analysis of the sensitivity of the active TB incidence rate to the pulmonary tuberculosis surgical treatment rate shows a logarithmic increase at the beginning and a rapid transition to a stable level. This is due to the achievement of drug-free treatment of tuberculosis over a certain period. The number of surgical treatments for pulmonary tuberculosis is decreasing every year, but there are no large studies on the correlation of this indicator with the incidence of pulmonary

tuberculosis [16]. The sensitivity analysis demonstrated exactly these results the surgical interventions rate, as we performed statistical processing of the data from 2007, and for most of this sixteen-year period, surgical treatment was performed along with chemotherapy for tuberculosis.

CONCLUSIONS

The use of artificial intelligence to analyze socioeconomic, medical, and demographic data has helped to identify the main factors contributing to the incidence of tuberculosis in Ukraine. In particular, the analysis confirmed the significant impact of the number of specialized hospitals, fluoroscopic examinations of the population, and the frequency of bacterial excretion on the incidence rate.

The development and validation of machine learning models, including linear regression, random forests, and adaptive boosting, allowed for accurate forecasting of tuberculosis incidence. The use of 5-fold cross-validation increased the reliability of the predictions, ensuring stability and accuracy across different demographic groups.

The results of the SHAP analysis, which offers a methodology for interpreting complex machine learning models, show the most important factors that influence the incidence of tuberculosis in Ukraine, with the greatest impact shown in bacterial excretion rates and fluoroscopic examinations of the population.

Interpretation of complex models through SHAP analysis and sensitivity analysis provided a deep understanding of the impact of individual factors, allowing for the formulation of targeted strategies for TB control and prevention. This creates the basis for informed decision-making in the field of public health and optimization of health care resources.

REFERENCES

1. S.S. Chiang et al., “Clinical manifestations and epidemiology of adolescent tuberculosis in Ukraine,” *ERJ Open Res*, 6(3):00308-2020, 2020. doi: https://doi.org/10.1183/23120541.00308-2020
2. I. Margineanu et al., “TB therapeutic drug monitoring - analysis of opportunities in Romania and Ukraine,” *Int. J. Tuberc. Lung Dis.*, 27(11), pp. 816–821, 2023. doi: 10.5588/ijtld.22.0667
3. O.S. Shevchenko, L.D. Todoriko, I.A. Ovcharenko, O.O. Pogorelova, and I.O. Semianiv, “A mathematical model for predicting the outcome of treatment of multidrug-resistant tuberculosis,” *Wiad. Lek.*, 74(7), pp. 1649–1654, 2021. doi: 10.36740 WLek202107117
4. D. Butov et al., “National survey on the impact of the war in Ukraine on TB diagnostics and treatment services in 2022,” *Int. J. Tuberc. Lung. Dis.*, 27(1), pp. 86–88, 2023. doi: 10.5588/ijtld.22.0563
5. K. Lönnroth, E. Jaramillo, B.G. Williams, C. Dye, and M. Raviglione, “Drivers of tuberculosis epidemics: the role of risk factors and social determinants,” *Soc. Sci. Med.*, 68(12), pp. 2240–2246, 2009. doi: 10.1016/j.socscimed.2009.03.041
6. Rifat Atun, Diana E.C. Weil, Mao Tan Eang, and David Mwakyusa, “Health-system strengthening and tuberculosis control,” *The Lancet*, 375(9732), pp. 2169–2178, 2010. doi: 10.1016/S0140-6736(10)60493-X
7. M.A. Mujtaba et al., “Demographic and Clinical Determinants of Tuberculosis and TB Recurrence: A Double-Edged Retrospective Study from Pakistan,” *J. Trop. Med.*, vol. 2022, article ID 4408306, 2022. doi: 10.1155/2022/4408306
8. E.J. Topol, “High-performance medicine: the convergence of human and artificial intelligence,” *Nat. Med.*, vol. 25, pp. 44–56, 2019. doi: https://doi.org/10.1038/s41591-018-0300-7

9. P. Farmer, "The major infectious diseases in the world--to treat or not to treat?" *N. Engl. J. Med.*, 345(3), pp. 208–210, 2001. doi: 10.1056/NEJM200107193450310
10. N. Tang et al., "Machine Learning Prediction Model of Tuberculosis Incidence Based on Meteorological Factors and Air Pollutants," *Int. J. Environ. Res. Public Health*, 20(5), 3910, 2023. doi: 10.3390/ijerph20053910
11. E.J. Hwang et al., "Development and Validation of a Deep Learning-Based Automated Detection Algorithm for Major Thoracic Diseases on Chest Radiographs [published correction appears in JAMA Netw Open. 2019 Apr 5;2(4):e193260]," *JAMA Netw Open*, 2(3):e191095, 2019. doi: 10.1001/jamanetworkopen.2019.1095
12. S. Tuli, S. Tuli, R. Tuli, and S.S. Gill, "Predicting the growth and trend of COVID-19 pandemic using machine learning and cloud computing," *Internet of Things*, 11:100222, 2020. doi: 10.1016/j.iot.2020.100222
13. A. Rajkomar, J. Dean, and I. Kohane, "Machine learning in medicine," *N. Engl. J. Med.*, 380(14), pp. 1347–1358, 2019. doi: 10.1056/NEJMra1814259
14. K.E. Wiens et al., "Global variation in bacterial strains that cause tuberculosis disease: a systematic review and meta-analysis," *BMC Med.*, 16(1), article no. 196, 2018. doi: 10.1186/s12916-018-1180-x
15. V. Smelov et al., "Rationale and Purpose: The FLUTE Study to Evaluate Fluorography Mass Screening for Tuberculosis and Other Diseases, as Conducted in Eastern Europe and Central Asia Countries," *Int. J. Environ. Res. Public. Health*, 19(14), 8706, 2022. doi: 10.3390/ijerph19148706
16. R. Zaleskis, A.W. Mariani, F. Inzirillo, and I. Vasilyeva, "The Role of Surgery in Tuberculosis Management: Indications and Contraindications," in *G.B. Migliori, M.C. Raviglione (eds) Essential Tuberculosis*. Springer, Cham, 2021. doi: https://doi.org/10.1007/978-3-030-66703-0_15

Received 10.05.2024

INFORMATION ON THE ARTICLE

Denys V. Nevinskyi, ORCID: 0000-0002-0962-072X, Lviv Polytechnic National University, Ukraine, e-mail: nevinskiy90@gmail.com

Dmytro I. Martjanov, ORCID: 0009-0003-3919-4412, Lviv Polytechnic National University, Ukraine, e-mail: d.martjnoff@gmail.com

Ihor O. Semianiv, ORCID: 0000-0003-0340-0766, Bukovinian State Medical University, Ukraine, e-mail: igor_semianiv@bsmu.edu.ua

Yaroslav I. Vyklyuk, ORCID: 0000-0003-4766-4659, Lviv Polytechnic National University, Ukraine, e-mail: vyklyuk@ukr.net

ВИВЧЕННЯ ЗВ'ЯЗКУ МІЖ ТУБЕРКУЛЬОЗОМ ТА СОЦІАЛЬНО-ЕКОНОМІЧНИМИ, МЕДИЧНИМИ, ДЕМОГРАФІЧНИМИ ЧИННИКАМИ В УКРАЇНІ / Д.В. Невінський, Д.І. Мартьянов, І.О. Сем'янів, Я.І. Виклюк

Анотація. Натепер Україна переживає новий, черговий наступ туберкульозу. Це дослідження аналізує вплив різних соціально-економічних та медичних факторів, включаючи: кількість спеціалізованих лікарень, флюорографічні огляди населення, кількість медичних працівників, рівень зловживання алкоголем та наркотиками та інші на поширеність туберкульозу серед різних демографічних груп населення в Україні. Використання методів штучного інтелекту дало змогу визначити ключові чинники, що сприяють зростанню або зниженню захворюваності на туберкульоз. Результати SHAP (SHapley Additive exPlanations) аналізу, який пропонує методологію для інтерпретації складних моделей машинного навчання, показує найважливіші фактори, які впливають на захворюваність туберкульозом в Україні. Більш важливу інформацію несе аналіз чутливості, який підтвердив отримані показники в SHAP аналізі.

Ключові слова: штучний інтелект, туберкульоз, захворюваність, соціально-демографічні чинники, медичні чинники, демографічні чинники.

МАТЕМАТИЧНІ МЕТОДИ, МОДЕЛІ,
ПРОБЛЕМИ І ТЕХНОЛОГІЇ ДОСЛІДЖЕННЯ
СКЛАДНИХ СИСТЕМ

UDC 519.245+004.896
DOI: 10.20535/SRIT.2308-8893.2025.1.03

EFFICIENCY COMPARISON OF MISSING DATA IMPUTATION METHODS IN PREDICTIVE MODEL CREATION

A. POPOV

Abstract. Missing data is a common issue in data analysis and machine learning. This article analyzes the impact of missing data imputation methods during the data preprocessing stage on the quality of forecasting models. Selected methods are listwise deletion, mean imputation, and two implementations of the multiple imputation method in Python and R languages. Selected classifiers are Logistic Regression, Random Forest, Support Vector Machine, and Light Gradient Boosting Machine. The performance quality of forecasting models is estimated using accuracy, precision, and recall metrics. Two datasets were used as binary classification problems with different target metrics. The highest performance was achieved when the R implementation of the multiple imputation method was combined with RF and LGBM classifiers.

Keywords: missing data, imputation methods, forecasting models, machine learning.

INTRODUCTION

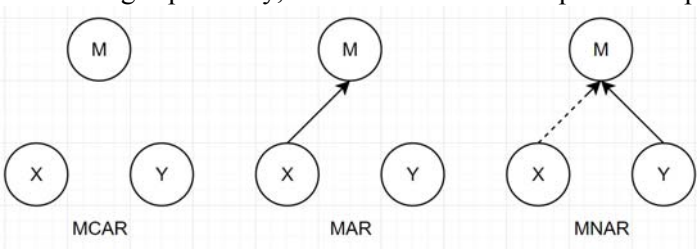
Today, every forecasting task involves processing large amounts of information. One of the key aspects of preparing data for creating predictive models is handling missing values, as machines learning algorithms mostly require complete data. In real-world datasets, it is common to find gaps that can occur for a variety of reasons, such as technical issues, human errors, the specifics of the research in which the data was collected, and other factors. Missing information in a dataset can distort statistical parameters, which can have a serious impact on the quality and reliability of the model and lead to incorrect conclusions. With proper handling of missing data prior to model training, the probability of successful training of a predictive model can be increased, which will positively affect its quality.

MISSING DATA MECHANISMS

To describe the logic behind the occurrence of missing data, the concept of a missing data mechanism was created. A mechanism is a term that is meant to describe in a general way the relationship between missing and observed data. According to the most common classification, there are three types of mechanisms based on what determines the probability of missing a particular variable in the observation: missing completely at random (MCAR), missing at random (MAR), and missing not at random (MNAR; Rubin 1976 [1]). MCAR is the case when the

© *Publisher IASA at the Igor Sikorsky Kyiv Polytechnic Institute*, 2025

missingness is completely random, i.e., independent of the values of the variables in the set. This mechanism usually poses the least problems for imputation, since the statistical parameters of the data are by definition not biased. MAR — the absence of data on a particular variable depends on the values of other variables in the set, but does not depend on the value of the variable itself. This mechanism inherently contains bias and requires more careful handling than MCAR. MNAR — the absence of a variable depends on the value of the variable itself. This is the most complicated mechanism that does not have a clearly described solution, and in this case, the processing of missing data requires a specialised approach. Figure shows a simplified diagram of the mechanisms, where Y is the variable in question, M is an indicator of missing data for Y, X is other variables, a solid arrow is an existing dependency, and a dashed arrow is a possible dependency.



Simplified diagram of missing data mechanisms

KEY STAGES OF IMPUTATION

Data analysis. Determining statistical parameters of the available data, analysing relations between variables, correlations, identifying missing data, analysing them if possible, and determining the mechanism of their formation. The purpose of this stage is to gain an understanding of the available data and, ideally, the missing data, which will greatly facilitate the process of filling in the data.

Method selection for processing missing data. The large variety of available methods allows choosing the most suitable option for a particular task. The choice may depend on the amount of missing data, the mechanism behind it and complexity. Simple single-imputation methods (such as mean, mode, interpolation) are very popular and generally accepted, they are easier to understand and implement, but have disadvantages that limit their use. Sophisticated methods usually provide better imputation because they are able to take into account the relations between the data and do not skew the statistical parameters as a result, thus there are fewer limitations to their use. [2] In many cases, it makes sense to choose several methods and compare the results to choose the most appropriate one for the task at hand.

Performing the imputation. Application of the selected methods to fill in the missing values based on the observed data. This stage results in a complete dataset. The statistical quality of the imputation may depend on the nature of the gaps, the number of gaps, and the selected method. The wrong choice of method can lead to significant distortion of the results.

RELATED WORK

The topic of missing data processing is addressed in a large number of different studies, since it appears in any field and can be solved in a variety of ways with different levels of efficiency. With the accelerating development of artificial intel-

ligence and machine learning technologies, the topic of missing data processing has become even more discussed — high-quality process modelling in any field requires high-quality data, which creates the necessity of efficient processing of missing values. Research on methods is diverse, and depends on the goal of the researchers: some papers address general issues, review methods, and propose solutions [3–7]. In other works, there is a specific problem and methods for solving it are considered.

In particular, in the paper "The impact of imputation quality on machine learning classifiers for datasets with missing values" [8], the authors study the impact of imputation methods on the predictive ability of models. The methods studied are mean imputation, multiple imputation by chained equations (MICE), MissForest, generative adversarial imputation networks (GAIN), and missing data importance-weighted autoencoder (MIWAE); the selected datasets include both complete datasets with MCAR gaps of 25–50% and datasets with intrinsic MNAR gaps. The models under study are logistic regression, random forest, XGBoost, and artificial neural network. The selected datasets are used to compare the results between the trained models. The paper uses a multivariate ANOVA model to evaluate the impact of the imputation on the quality of the models. The results show that the quality of predictive models depends on the amount of missing data and training on imputed datasets usually produces lower quality results compared to training on complete datasets. At the same time, for the same dataset, the quality ranking of the models usually does not change for different amounts of missing data, i.e. a model that performs better on 25% of missing data will also perform better on 50% of the missing data. Different methods perform better depending on the dataset, but some imputation methods have less variation in quality across datasets, with MIWAE consistently performing well across the study. In some cases, logistic regression, which typically has the worst quality metrics, was also able to achieve high quality metrics.

Another paper by Jale Bektas, Turgay Ibrikci, and Ismail Turkay Ozcan [9] investigates the impact of imputation methods on the quality of classifiers in the task of diagnosing coronary artery disease. In this paper, three imputation methods based on machine learning techniques (K-means, multilayer perceptron, and self-organising maps) are presented and their performance is compared with the conventional mean imputation method and listwise deletion. The selected classification methods were Logistic Model Trees (LMT), multilayer perceptron, random forest method, and support vector machine. The developed imputation methods showed significantly better results than the mean imputation method, which was ranked fourth in terms of model quality, surpassed only by the listwise deletion. The best results were achieved when using self-organising maps (88.23% accuracy), and the most stable results were obtained when using a multilayer perceptron.

The papers "Do we really need imputation in AutoML predictive modelling?" [10] and "Does imputation matter? Benchmark for predictive models" [11] investigate the necessity of using complex imputation methods in machine learning processes. In the first study, 6 imputation methods were used to process data in 25 datasets with natural missing data and 10 datasets with artificial missing data. In the second one, 7 imputation methods were used on 13 classification tasks. The conclusions of both papers are that simple methods usually perform slightly worse than more complex methods, while gaining considerably in computational power. The first paper found that using a binary indicator with simple mean/mode imputation (for continuous and categorical data, respectively) per-

formed well and was significantly more efficient than more complex methods. In the second paper, simple methods also achieved good results, although even with similar predictive quality of the models, more complex methods produced more statistically accurate imputations.

In summary, the use of imputation methods at the stage of data preprocessing is a common subject in machine learning, with wide application regardless of the specific field of study.

STATEMENT OF THE RESEARCH PROBLEM

The purpose of this paper is to investigate the impact of missing data processing method on the quality of predictive machine learning models. In the process, we take complete datasets and using them as basis we artificially create datasets with different missing data configurations to study the effect of imputations on the predictive ability of models. All datasets are taken from the public domain.

The research algorithm consists of the following general steps: selection of imputation methods for the study, selection of prediction methods, search and research of datasets, creation of datasets with missing data, processing missing data, training models on the obtained datasets, and analysis of the results.

Four imputation methods were selected for the study:

1. Listwise deletion.
2. Mean imputation.
3. Multiple imputation using Python library scikit-learn (Iterative Imputer).
4. Multiple imputation using R library MICE.

The following 4 algorithms were chosen as forecasting algorithms:

1. Logistic regression.
2. Support vector machine.
3. Random Forest.
4. LGBM (Light Gradient Boosting Machine).

The first selected dataset is the Churn dataset of bank customers, the task of classification is to determine customer churn, i.e. to identify customers who are likely to cancel their bank services based on the available data. The selected dataset consists of 10.000 records and 10 variables, including 4 continuous and 6 categorical variables. The continuous variables are: *CreditScore* — customer's credit score, *EstimatedSalary* — customer's estimated salary, *Age* — customer's age, and *Balance* — customer's balance. Categorical variables include: *Geography* — country of origin of the customer, *Gender* — gender of the customer, *Tenure* — number of years the customer has been with the bank, *NumOfProducts* — number of bank products used by the customer, *HasCrCard* — indicator of whether the customer has a bank credit card, *isActiveMember* — indicator of customer activity, and *Exited* — target variable reflecting the churn/retention status of the customer.

To handle missing data, we use only continuous variables. In total, 12 datasets with different types and numbers of gaps were created, including 4 datasets with only MCAR gaps, 2 datasets with only MAR gaps, 1 dataset with only MNAR gaps, 2 datasets with mixed MCAR and MAR gaps (such datasets are considered in the MAR category), and 2 datasets with mixed gaps using MNAR gaps. As a result, 48 datasets were obtained after completion of the imputation. [12] Datasets with mixed gaps are considered in the category of a less strong assumption — for example, for mixed MAR and MCAR gaps, the dataset is considered in the MAR category.

The second selected dataset is a set of characteristic parameters of wine for the purpose of wine quality classification. The selected dataset consists of 1599 records and 12 variables, of which 11 are continuous and 1 is a categorical variable. The categorical variable is the target variable quality. The continuous variables are: *fixed acidity* — fixed (nonvolatile) acids, *volatile acidity* — the amount of volatile acids, *citric acid* — the amount of citric acid, *residual sugar* — the amount of residual sugar after the fermentation process is stopped, *chlorides* — the amount of salt, *free sulfur dioxide* — the free form of SO_2 that exists in equilibrium between molecular SO_2 (as a dissolved gas) and bisulfite ion, *total sulfur dioxide* — the amount of free and bound SO_2, *density* — the density (the density of wine is almost the same as that of water, depending on the alcohol and sugar content), *pH* — an indicator of the acidity/alkalinity of wine from 0 to 14 (most wines are between 3–4 on this scale) and *sulphates* — additives to wine that can contribute to the level of SO_2.

In total, 9 datasets with different types and numbers of gaps were created, including 3 datasets with exclusively MCAR gaps, 2 datasets with exclusively MAR gaps, 1 dataset with exclusively MNAR gaps, 1 dataset with mixed MCAR and MAR gaps, and 2 datasets with mixed gaps using MNAR gaps. As a result, 36 datasets were obtained after the completion of the imputation.

PERFORMANCE METRICS

To evaluate the quality of the obtained predictive models, we used the accuracy, precision and recall metrics based on the confusion matrix.

- Accuracy is a metric of the overall classification accuracy of the model, calculated as the ratio of correct predictions to all predictions.
- Precision is a metric that shows how many positive predictions were correct.
- Recall is a metric that shows how many elements of a positive class were detected by the model.

For each dataset, one of the metrics is the target metric, i.e. the main quality criterion in the context of a particular task. The quality comparison was performed for the values of the target metrics for the models trained on the imputed datasets.

In the Churn dataset, the target metric is recall, since the most important ability of the model should be the ability to correctly identify customers who will leave. In the Wine dataset, the target metric is accuracy, since the accuracy of classification is equally important for both classes.

CREATION OF ARTIFICIAL MISSING DATA

Missing data was created with different combinations of mechanisms and quantities. For the purpose of more accurate comparison, the gaps were created exclusively in the training dataset — this was done in order to compare the classification quality of different models on the same test set. Before creating missing data, the full datasets were split into training and test samples in the ratio of 80 to 20.

The number of MCAR missing data for each selected variable ranges from 5% to 20%, and the total number of observations with gaps in the MCAR datasets ranges from 9.72% to 47.54%. To create MAR missing data two variables were selected and the values of the first variable were removed for records that had values for the second variable below or above the selected percentile. The selected percentiles ranged from 5% to 20% for values below them and 90–95% for values above them. The total number of observations with missing data ranged

from 21.59% to 48.83%. To create the MNAR type of missing data, a variable was selected and those values below or above the selected percentile were removed, which ranged from 7 to 13% and from 90% to 93%, respectively. The total number of observations with missing data ranged from 23.77% to 45.97%.

The total number of records with missing data for the datasets derived from the first dataset ranged from 9.72% to 48.82%. The average number was 30.7%. The size of the full training dataset was 8000 records and the test dataset was 2000 records.

The total number of records with missing data for the datasets derived from the second dataset ranged from 18.14% to 45.97%, with an average of 33%. The size of the full training set was 1279 records, and the test set was 320 records.

MODEL TRAINING RESULTS

The performance metrics of models trained on complete datasets are presented in Table 1. The highest predictive quality was achieved with LGBM model for the Churn dataset and RF model for Wine dataset.

Table 1. Results of training on complete datasets

Churn				Wine			
Model	Accuracy	Precision	Recall	**Model**	Accuracy	Precision	Recall
LR	0.725000	0.386397	0.679389	**LR**	0.784375	0.392157	0.851064
SVC	0.799500	0.492982	0.715013	**SVC**	0.859375	0.513514	0.808511
RF	0.808500	0.508711	0.743003	**RF**	0.8625	0.518072	0.914894
LGBM	0.820500	0.530357	0.755725	**LGBM**	0.840625	0.476744	0.87234

Tables 2, 3, 4 present the performance metrics of models trained on imputed Churn datasets with MCAR, MAR and MNAR missing data respectively.

Table 2. Results of training on imputed Churn datasets with MCAR missing data

MCAR Churn													
	Model	9.72%			18.46%			22.03%			47.54%		
		Accuracy	Precision	Recall	Accuracy	Precision	Recall	Accuracy	Precision	Recall	Accuracy	Precision	Recall
Listwise	**LR**	0.724	0.383602	0.666667	0.678	0.348247	0.732824	0.678	0.348247	0.732824	0.7285	0.387048	0.653944
	SVC	0.7945	0.484211	0.70229	0.805	0.502712	0.707379	0.79	0.477686	0.735369	0.805	0.502879	0.666667
	RF	0.8085	0.508865	0.73028	0.8015	0.496587	0.740458	0.8005	0.494845	0.732824	0.8135	0.518587	0.709924
	LGBM	0.812	0.514834	0.750636	0.804	0.500846	0.753181	0.8095	0.510345	0.753181	0.812	0.515371	0.725191
	Avg	0.78475	0.472878	0.712468	0.772125	0.462098	0.733461	0.7695	0.457781	0.73855	0.78975	0.480971	0.688932
Mean	**LR**	0.7285	0.390988	0.684478	0.7245	0.386494	0.684478	0.7275	0.389535	0.681934	0.721	0.382646	0.684478
	SVC	0.795	0.48532	0.715013	0.799	0.492091	0.712468	0.8025	0.498246	0.722646	0.812	0.515315	0.727735
	RF	0.802	0.497453	0.745547	0.7925	0.481544	0.73028	0.803	0.499145	0.743003	0.7995	0.493151	0.732824
	LGBM	0.817	0.52356	0.763359	0.8145	0.518966	0.765903	0.812	0.51463	0.760814	0.805	0.502555	0.750636
	Avg	0.785625	0.47433	0.727099	0.782625	0.469774	0.723282	0.78625	0.475389	0.727099	0.784375	0.473417	0.723918
Iterative	**LR**	0.6745	0.345694	0.735369	0.6745	0.345694	0.735369	0.678	0.348247	0.732824	0.68	0.349206	0.727735
	SVC	0.809	0.509874	0.722646	0.8115	0.514235	0.735369	0.8115	0.514337	0.73028	0.818	0.527938	0.697201
	RF	0.8385	0.573529	0.694656	0.8355	0.568085	0.679389	0.8415	0.581197	0.692112	0.8245	0.542339	0.684478
	LGBM	0.841	0.585421	0.653944	0.8415	0.584071	0.671756	0.8365	0.571121	0.6743	0.818	0.527619	0.704835
	Avg	0.79075	0.50363	0.701654	0.79075	0.503021	0.705471	0.791875	0.503726	0.707379	0.785125	0.486776	0.703562
MICE	**LR**	0.728	0.390421	0.684478	0.726	0.387844	0.681934	0.7265	0.387755	0.676845	0.7005	0.36658	0.720102
	SVC	0.795	0.485062	0.70229	0.798	0.490435	0.717557	0.8005	0.4947	0.712468	0.813	0.517625	0.709924
	RF	0.801	0.495798	0.750636	0.8085	0.508961	0.722646	0.8005	0.495	0.755725	0.814	0.519626	0.707379
	LGBM	0.8175	0.524735	0.755725	0.8125	0.515789	0.748092	0.8155	0.521053	0.755725	0.824	0.538752	0.725191
	Avg	0.785375	0.474004	0.723282	0.78625	0.475757	0.717557	0.78575	0.474627	0.725191	0.787875	0.485646	0.715649

Table 3. Results of training on imputed Churn datasets with MAR missing data

MAR Churn													
	Model	21.59%			26.16% (+ MCAR)			39.99%			48.73% (+ MCAR)		
		Accuracy	Precision	Recall	Accuracy	Precision	Recall	Accuracy	Precision	Recall	Accuracy	Precision	Recall
Listwise	**LR**	0.7235	0.385714	0.687023	0.6605	0.336758	0.750636	0.7315	0.395349	0.692112	0.737	0.399698	0.6743
	SVC	0.791	0.479132	0.73028	0.7915	0.480198	0.740458	0.7755	0.454984	0.720102	0.7725	0.449838	0.707379
	RF	0.787	0.473083	0.737913	0.798	0.490566	0.727735	0.8075	0.507143	0.722646	0.7835	0.466667	0.712468
	LGBM	0.803	0.499139	0.737913	0.8085	0.509158	0.707379	0.8105	0.512411	0.735369	0.805	0.502636	0.727735
	Avg	0.776125	0.459267	0.723282	0.764625	0.45417	0.731552	0.78125	0.467472	0.717557	0.7745	0.45471	0.705471
Mean	**LR**	0.7235	0.384058	0.6743	0.716	0.376934	0.681934	0.7215	0.381159	0.669211	0.7225	0.383285	0.676845
	SVC	0.7965	0.487931	0.720102	0.792	0.480475	0.720102	0.797	0.488927	0.73028	0.795	0.485114	0.704835
	RF	0.802	0.497427	0.737913	0.814	0.519409	0.715013	0.7855	0.471061	0.745547	0.795	0.485904	0.745547
	LGBM	0.8135	0.517241	0.763359	0.8175	0.525926	0.722646	0.81	0.511149	0.75827	0.801	0.495881	0.765903
	Avg	0.783875	0.471664	0.723919	0.784875	0.475686	0.709924	0.7785	0.463074	0.725827	0.778375	0.462546	0.723283
Iterative	**LR**	0.68	0.349206	0.727735	0.6765	0.346618	0.73028	0.68	0.349206	0.727735	0.689	0.356336	0.722646
	SVC	0.811	0.513711	0.715013	0.814	0.518717	0.740458	0.8125	0.515845	0.745547	0.8095	0.511111	0.70229
	RF	0.8435	0.58658	0.689567	0.837	0.572043	0.676845	0.835	0.567452	0.6743	0.834	0.564482	0.679389
	LGBM	0.845	0.595402	0.659033	0.8355	0.568966	0.671756	0.837	0.574944	0.653944	0.818	0.527619	0.704835
	Avg	0.794875	0.511225	0.697837	0.79075	0.501586	0.704835	0.791125	0.501862	0.700382	0.787625	0.489887	0.70229
MICE	**LR**	0.7235	0.384058	0.6743	0.7265	0.388081	0.679389	0.722	0.382055	0.671756	0.7265	0.387755	0.676845
	SVC	0.791	0.478336	0.70229	0.8025	0.498258	0.727735	0.7895	0.476271	0.715013	0.8075	0.50738	0.699746
	RF	0.8075	0.506849	0.753181	0.813	0.516522	0.755725	0.7965	0.488294	0.743003	0.8155	0.522556	0.707379
	LGBM	0.8125	0.515625	0.755725	0.813	0.516579	0.753181	0.812	0.51468	0.75827	0.822	0.533821	0.743003
	Avg	0.783625	0.471217	0.721374	0.78875	0.47986	0.729008	0.78	0.465325	0.722011	0.792875	0.487878	0.706743

Table 4. Results of training on imputed Churn datasets with MNAR missing data

MNAR Churn													
	Model	24.01%			31.74% (+ MAR)			35.81% (+ MCAR)			42.43% (+ MCAR/MAR)		
		Accuracy	Precision	Recall	Accuracy	Precision	Recall	Accuracy	Precision	Recall	Accuracy	Precision	Recall
Listwise	**LR**	0.7205	0.382436	0.687023	0.6605	0.336758	0.750636	0.65	0.331873	0.770992	0.65	0.331873	0.770992
	SVC	0.779	0.460292	0.722646	0.79	0.47644	0.694656	0.7925	0.480903	0.704835	0.779	0.459098	0.699746
	RF	0.8015	0.496503	0.722646	0.812	0.515654	0.712468	0.817	0.525424	0.709924	0.813	0.517691	0.707379
	LGBM	0.807	0.506087	0.740458	0.815	0.521024	0.725191	0.794	0.484034	0.732824	0.777	0.459168	0.75827
	Avg	0.777	0.46133	0.718193	0.769375	0.462469	0.720738	0.763375	0.455559	0.729644	0.75475	0.441958	0.734097
Mean	**LR**	0.7185	0.381616	0.697201	0.715	0.377931	0.697201	0.7125	0.375683	0.699746	0.7135	0.377049	0.70229
	SVC	0.769	0.445498	0.717557	0.7665	0.440895	0.70229	0.774	0.453249	0.727735	0.7755	0.454397	0.709924
	RF	0.771	0.450077	0.745547	0.781	0.464115	0.740458	0.7935	0.482759	0.712468	0.797	0.488774	0.720102
	LGBM	0.799	0.492487	0.750636	0.806	0.504488	0.715013	0.803	0.499086	0.694656	0.8055	0.503663	0.699746
	Avg	0.764375	0.44242	0.727735	0.767125	0.446857	0.713741	0.77075	0.452694	0.708651	0.772875	0.455971	0.708016
Iterative	**LR**	0.671	0.342823	0.735369	0.671	0.342823	0.735369	0.671	0.342823	0.735369	0.671	0.342823	0.735369
	SVC	0.8005	0.494505	0.687023	0.801	0.495379	0.681934	0.797	0.48816	0.681934	0.799	0.491682	0.676845
	RF	0.8265	0.545276	0.704835	0.822	0.535783	0.704835	0.83	0.555324	0.676845	0.8215	0.535714	0.687023
	LGBM	0.8265	0.546939	0.681934	0.826	0.546012	0.679389	0.8255	0.546025	0.664122	0.8255	0.545267	0.6743
	Avg	0.781125	0.482386	0.702290	0.78	0.479999	0.700382	0.780875	0.483083	0.689568	0.77925	0.478872	0.693384
MICE	**LR**	0.7225	0.385593	0.694656	0.7245	0.387464	0.692112	0.719	0.380481	0.684478	0.72	0.383543	0.699746
	SVC	0.792	0.480207	0.709924	0.7885	0.474832	0.720102	0.785	0.468908	0.709924	0.793	0.48199	0.715013
	RF	0.7925	0.482315	0.763359	0.79	0.478049	0.748092	0.7995	0.493197	0.737913	0.8035	0.5	0.709924
	LGBM	0.806	0.504303	0.745547	0.804	0.500855	0.745547	0.8185	0.527372	0.735369	0.798	0.490787	0.745547
	Avg	0.77825	0.463105	0.728372	0.77675	0.4603	0.726463	0.7805	0.46749	0.716921	0.778625	0.46408	0.717558

Tables 5, 6, 7 present the performance metrics of models trained on imputed Wine datasets with MCAR, MAR and MNAR missing data respectively.

Table. 5. Results of training on imputed Wine datasets with MCAR missing data

MCAR Wine										
	Model	18.14%			30.73%			45.35%		
		Accuracy	Precision	Recall	Accuracy	Precision	Recall	Accuracy	Precision	Recall
Listwise	**LR**	0.796875	0.408163	0.851064	0.81875	0.438202	0.829787	0.809375	0.425532	0.851064
	SVC	0.85625	0.506667	0.808511	0.865625	0.529412	0.765957	0.86875	0.537313	0.765957
	RF	0.86875	0.533333	0.851064	0.84375	0.481013	0.808511	0.865625	0.532258	0.702128
	LGBM	0.853125	0.5	0.87234	0.8625	0.519481	0.851064	0.84375	0.478873	0.723404
	Avg	0.84375	0.487041	0.845745	0.847656	0.492027	0.81383	0.846875	0.493494	0.760638
Mean	**LR**	0.803125	0.418367	0.87234	0.79375	0.405941	0.87234	0.796875	0.41	0.87234
	SVC	0.859375	0.513514	0.808511	0.8375	0.46988	0.829787	0.85	0.493506	0.808511
	RF	0.84375	0.483146	0.914894	0.85	0.494253	0.914894	0.85	0.494253	0.914894
	LGBM	0.8375	0.47191	0.893617	0.8375	0.471264	0.87234	0.834375	0.465909	0.87234
	Avg	0.835938	0.471734	0.872341	0.829688	0.460335	0.87234	0.832813	0.465917	0.867021
Iterative	**LR**	0.784375	0.392157	0.851064	0.7875	0.39604	0.851064	0.809375	0.427083	0.87234
	SVC	0.86875	0.534247	0.829787	0.8625	0.519481	0.851064	0.875	0.547945	0.851064
	RF	0.859375	0.512195	0.893617	0.86875	0.530864	0.914894	0.865625	0.526316	0.851064
	LGBM	0.840625	0.476744	0.87234	0.834375	0.464286	0.829787	0.840625	0.475	0.808511
	Avg	0.838281	0.478836	0.861702	0.838281	0.477668	0.861702	0.847656	0.494086	0.845745
MICE	**LR**	0.7875	0.398058	0.87234	0.784375	0.392157	0.851064	0.809375	0.425532	0.851064
	SVC	0.86875	0.534247	0.829787	0.859375	0.512821	0.851064	0.875	0.547945	0.851064
	RF	0.85	0.494253	0.914894	0.865625	0.52439	0.914894	0.86875	0.534247	0.829787
	LGBM	0.83125	0.460674	0.87234	0.853125	0.5	0.87234	0.8625	0.518519	0.893617
	Avg	0.834375	0.471808	0.87234	0.840625	0.482342	0.872341	0.853906	0.506561	0.856383

Table. 6. Results of training on imputed Wine datasets with MAR missing data

MAR Wine										
	Model	23.53%			34.48% (+ MCAR)			42.30%		
		Accuracy	Precision	Recall	Accuracy	Precision	Recall	Accuracy	Precision	Recall
Listwise	**LR**	0.828125	0.45122	0.787234	0.815625	0.428571	0.765957	0.809375	0.420455	0.787234
	SVC	0.853125	0.5	0.659574	0.821875	0.434211	0.702128	0.80625	0.363636	0.425532
	RF	0.8625	0.525424	0.659574	0.84375	0.477612	0.680851	0.846875	0.48	0.510638
	LGBM	0.86875	0.538462	0.744681	0.8375	0.467532	0.765957	0.85625	0.507692	0.702128
	Avg	0.853125	0.503777	0.712766	0.829688	0.451982	0.728723	0.829688	0.442946	0.606383
Mean	**LR**	0.790625	0.397959	0.829787	0.8	0.414141	0.87234	0.8	0.412371	0.851064
	SVC	0.85625	0.507042	0.765957	0.8375	0.469136	0.808511	0.8375	0.467532	0.765957
	RF	0.853125	0.5	0.87234	0.84375	0.481481	0.829787	0.8625	0.519481	0.851064
	LGBM	0.8375	0.47191	0.893617	0.825	0.448276	0.829787	0.828125	0.455556	0.87234
	Avg	0.834375	0.469228	0.840425	0.826563	0.453259	0.835106	0.832031	0.463735	0.835106
Iterative	**LR**	0.821875	0.445652	0.87234	0.8125	0.43299	0.893617	0.815625	0.434783	0.851064
	SVC	0.865625	0.527027	0.829787	0.83125	0.453333	0.723404	0.85	0.493151	0.765957
	RF	0.85625	0.506329	0.851064	0.859375	0.512821	0.851064	0.86875	0.535211	0.808511
	LGBM	0.853125	0.5	0.893617	0.8375	0.469136	0.808511	0.859375	0.512821	0.851064
	Avg	0.849219	0.494752	0.861702	0.835156	0.46707	0.819149	0.848438	0.493992	0.819149
MICE	**LR**	0.8125	0.430108	0.851064	0.8125	0.43299	0.893617	0.81875	0.43956	0.851064
	SVC	0.8625	0.52	0.829787	0.85625	0.507042	0.765957	0.84375	0.479452	0.744681
	RF	0.85625	0.506173	0.87234	0.859375	0.513158	0.829787	0.859375	0.513514	0.808511
	LGBM	0.834375	0.464286	0.829787	0.84375	0.481928	0.851064	0.846875	0.488095	0.87234
	Avg	0.841406	0.480142	0.845745	0.842969	0.48378	0.835106	0.842188	0.480155	0.819149

Table 7. Results of training on imputed Wine datasets with MNAR missing data

MNAR Wine										
	Model	23.77%			32.60% (+ MCAR)			45.97% (+ MCAR/MAR)		
		Accuracy	Precision	Recall	Accuracy	Precision	Recall	Accuracy	Precision	Recall
Listwise	LR	0.796875	0.404255	0.808511	0.796875	0.404255	0.808511	0.828125	0.452381	0.808511
	SVC	0.8375	0.463768	0.680851	0.809375	0.397059	0.574468	0.859375	0.52	0.553191
	RF	0.853125	0.5	0.702128	0.825	0.44	0.702128	0.853125	0.5	0.553191
	LGBM	0.834375	0.461538	0.765957	0.83125	0.454545	0.744681	0.8375	0.454545	0.531915
	Avg	0.830469	0.45739	0.739362	0.815625	0.423965	0.707447	0.844531	0.481732	0.611702
Mean	LR	0.79375	0.402062	0.829787	0.790625	0.395833	0.808511	0.809375	0.423913	0.829787
	SVC	0.83125	0.452055	0.702128	0.834375	0.460526	0.744681	0.8625	0.52381	0.702128
	RF	0.83125	0.455696	0.765957	0.8375	0.469136	0.808511	0.85	0.493151	0.765957
	LGBM	0.825	0.445783	0.787234	0.825	0.448276	0.829787	0.834375	0.464286	0.829787
	Avg	0.820313	0.438899	0.771277	0.821875	0.443443	0.797873	0.839063	0.47629	0.781915
Iterative	LR	0.803125	0.418367	0.87234	0.80625	0.421053	0.851064	0.809375	0.423913	0.829787
	SVC	0.853125	0.5	0.787234	0.8625	0.521739	0.765957	0.8625	0.52381	0.702128
	RF	0.846875	0.486842	0.787234	0.84375	0.481013	0.808511	0.85	0.493151	0.765957
	LGBM	0.840625	0.47561	0.829787	0.83125	0.455696	0.765957	0.834375	0.464286	0.829787
	Avg	0.835938	0.470205	0.819149	0.835938	0.469875	0.797872	0.839063	0.47629	0.781915
MICE	LR	0.796875	0.40625	0.829787	0.80625	0.419355	0.829787	0.803125	0.416667	0.851064
	SVC	0.840625	0.474359	0.787234	0.85	0.493333	0.787234	0.878125	0.560606	0.787234
	RF	0.88125	0.561644	0.87234	0.86875	0.534247	0.829787	0.859375	0.513889	0.787234
	LGBM	0.8375	0.46988	0.829787	0.83125	0.45679	0.787234	0.834375	0.460526	0.744681
	Avg	0.839063	0.478033	0.829787	0.839063	0.475931	0.808511	0.84375	0.487922	0.792553

DISCUSSION

Listwise deletion, which is a highly problematic method from the statistical point of view and is rarely recommended, has proven in some cases to be able to produce datasets that yield prediction quality that is as good as when sophisticated imputation methods are used. The method performs better with the Wine dataset, where the target metric is accuracy, and performs worse with the Churn dataset when the target metric is recall. In particular, for Churn on datasets with MCAR missing data, the method showed mixed and unpredictable results. In some cases, the obtained value of the target metric was not worse than the results obtained using other methods, but the same model could have very different metric values on different datasets, which made it difficult to predict the result. On datasets with MAR missing data, the recall value was as good as the other methods, but it also often increased at the cost of the accuracy value, so the overall quality of the models was lower. Similar results can be seen on the datasets with MNAR missing data. In addition, on these datasets, the highest recall score was achieved using logistic regression, which usually shows the worst results. Thus, for the recall as target metric, good results using this method are not uncommon, but the best quality models are obtained on datasets that have exclusively MCAR mechanism — otherwise, the overall quality of the model decreases.

For the Wine dataset with the accuracy target metric, the method's performance was significantly higher. Despite the loss of a large amount of information, the predictive quality of the obtained models was not inferior to other methods. It is worth noting that the value of the recall metric was significantly lower than that of the other methods, especially as the number of missing data increased, which resulted in lower overall model quality. The most balanced models were obtained on datasets containing only MCAR missing data: accuracy ranged from 79.7% to

86.9%, recall from 76.7% to 87%, which matches the quality of models obtained using more complex methods. On the datasets with MAR and MNAR missing data, the trained models showed good results in terms of accuracy (79.7–86.9%), but the values of the recall metric were significantly lower (51–80.1%).

In summary, in both problems, it was observed that the method is not a reliable choice for the recall metric, as satisfactory and predictable results were obtained only on MCAR missing data. At the same time, the method is able to show very good results when working with the accuracy metric, but is still limited by the MCAR missing data mechanism and the percentage of missing data to obtain balanced models for the metrics. Due to the general unreliability and unpredictability of the method, it can be concluded that it is not the best choice for solving such problems, but its use does not necessarily mean obtaining unsatisfactory results, because predictive models can often learn to correctly identify the features of the target classes of the problem even using datasets with biased statistical parameters.

Mean imputation generally showed more reliable results on most datasets than listwise deletion, as the quality of trained models fluctuated less regardless of the type and number of missing data. For this method, the best results were achieved with SVC, RF or LGBM classifiers, while the method performed worse with logistic regression. From a statistical point of view, a significant problem with this method is the reduction of data variability and weakening of correlations between variables (which was observed for the datasets imputed with this method), but, as in the case of listwise deletion, machine learning models are able to learn to identify features of the target class even with statistically skewed data, and they do so with greater success for the mean imputation method. Using this method, satisfactory results were obtained on MCAR and MAR missing data for both datasets for three out of four methods: for Churn, the accuracy was in the range of 71–81% in almost all cases, recall was 71–76%, and only when using logistic regression were the results worse (recall 67–68%); for the Wine dataset, the accuracy was 79–86.2%, recall was 80–91.5% (exceptions are two cases of SVC method on MAR, where recall was 76.6%). The training results on MNAR missing data were of lower quality, with a noticeable decrease in recall compared to other methods: for Churn, the metric had results of 69.4–75%, and for Wine — 70–83%. The best results in these cases were achieved using LGBM (Churn) and RF or SVC (Wine). In summary, using mean imputation method is a relatively good choice, as the models trained on these datasets were of high quality more often and had more predictable results than those using listwise deletion. In addition, the method also performs better because the range of values obtained for the metrics, even for the worst outcomes, is smaller, making the results more predictable.

Multiple imputation in the Python implementation of IterativeImputer from the scikit-learn library showed unsatisfactory results for the Churn dataset. The models often did not meet the minimum required classification quality. Across all the missing data mechanisms, it was observed that this method worked best with logistic regression and support vector machine — in particular, logistic regression repeatedly showed significantly better results on the recall metric when using this method compared to the complete data (up to 73.5%) — but fell short on other metrics (67–68% accuracy). The SVC models performed relatively well (accuracy 80–81%, recall 70–74.5%), except for datasets with MNAR-type missing data (accuracy 80%, recall 68%). The RF and LGBM models consistently had low recall values in combination with this method (67–70%), regardless of the mechanism and number of gaps.

On the Wine dataset, by contrast, the method performed quite well on MCAR and MAR missing data, especially when SVC and RF models were used

(accuracy 83–87.5%, recall 82.9–91.5% with two exceptions on MAR data for SVC). Logistic regression generally performed worse than the other models, but often had the highest recall, which was also observed on the Churn dataset. On the MNAR missing data, the metrics were also quite high and did not fall short of other imputation methods.

In general, this method proved to be quite unpredictable and data-dependent, as there was a significant difference in quality between models trained on different groups of datasets. In addition, specifically for the case of maximising recall, the method showed unsatisfactory results, although it was able to create powerful models for the Wine task with the target accuracy metric.

Multiple imputation in the implementation of the R MICE library proved to be the best, providing the most consistently high results for all metrics, which were closest to the performance after training on the complete datasets. The method performed well on all datasets regardless of the type, combination and number of gaps. For the Churn dataset, it worked best when combined with the RF and LGBM algorithms, with the RF algorithm even performing better in some cases using this imputation method than after training on the full dataset (MAR dataset). On the Wine models, the method also showed excellent results, delivering high scores on both the target and recall metrics. The method worked best with RF and SVC models. In general, the method had the highest level of reliability and predictability of results, and the models trained on the datasets with this imputation had consistently high prediction quality with the least fluctuations. Overall, this particular implementation of the multiple imputation method proved to be the most successful choice among studied methods for solving the problem of processing missing data.

CONCLUSIONS

The widespread problem of missing data is becoming especially relevant today due to the rapid development of artificial intelligence and machine learning technologies, which create a growing need for large amounts of high-quality data, as most algorithms require complete datasets. A large number of different methods for processing missing data have been created to solve the problem of missing data, while preserving the statistical parameters of the data for the success of further modelling. An important issue is the compatibility of imputation methods and predictive models, as different methods have different levels of quality and predictability of modelling results.

In this paper, an impact of the selected imputation methods on the quality of forecasting models is analysed. The best results were obtained using the multiple imputation method in the implementation from the R MICE library. Training on data using this method most reliably produced results that had high scores on quality metrics and were characterised by smaller quality fluctuations compared to other methods. The Python implementation of the multiple imputation method was less reliable, as its effectiveness strongly depended on the target metric and the specifics of the available data.

It has also been observed that statistically unreliable imputation methods, such as mean imputation or listwise deletion, do not necessarily lead to poor prediction results, as quite often predictive models are able to learn to recognise the target class even in the case of biased parameters. Therefore, their use, although riskier and more dependent on the characteristics of the available data, can also produce satisfactory results, which may not be inferior in quality to training using more complex methods.

REFERENCES

1. Donald B. Rubin, "Inference and Missing Data," *Biometrika*, vol. 63, no. 3, pp. 581–592, 1976.
2. Craig K. Enders, *Applied Missing Data Analysis*; 1 ed. The Guilford Press, 2010, 377 p.
3. Therese D. Pigott, "A review of methods for missing data," *Educational Research and Evaluation*, vol. 7, no. 4, pp. 353–383, 2001.
4. Luke Oluwaseye Joel, Wesley Doorsamy, and Babu Sena Paul, "A Review of Missing Data Handling Techniques for Machine Learning," *International Journal of Innovative Technology and Interdisciplinary Sciences (IJITIS)*, vol. 5, no. 3, pp. 971–1005, 2022. doi: https://doi.org/10.15157/IJITIS.2022.5.3.971-1005
5. Helen Bridge, Thomas Schindler, "The perils of the unknown: Missing data in clinical studies," *Medical Writing*, 27(1), pp. 56–59, 2018.
6. Tlamelo Emmanuel, Thabiso Maupong, Dimane Mpoeleng, Thabo Semong, Banyatsang Mphago, and Oteng Tabona, "A survey on missing data in machine learning," *Journal of Big Data*, 8(1), article no. 140, 2021. doi: 10.1186/s40537-021-00516-9
7. Hyun Kang, "The prevention and handling of the missing data," *Korean Journal of Anesthesiology*, 64(5), pp. 402–406, 2013. doi: 10.4097/kjae.2013.64.5.402
8. Tolou Shadbahr et al., "The impact of imputation quality on machine learning classifiers for datasets with missing values," *Communications medicine*, vol. 3, article no. 139, 2023. doi: 10.1038/s43856-023-00356-z
9. Jale Bektas, Turgay Ibrikci, and Ismail Ozcan, "The impact of imputation procedures with machine learning methods on the performance of classifiers: An application to coronary artery disease data including missing values," *Biomedical Research*, 29(13), pp. 2780–2785, 2018. doi: 10.4066/biomedicalresearch.29-18-199
10. George Paterakis, Stefanos Fafalios, Paulos Charonyktakis, Vassilis Christophides, and Ioannis Tsamardinos, "Do we really need imputation in AutoML predictive modeling?" *ACM Transactions on Knowledge Discovery from Data*, 18(6), 2024. doi: 10.1145/3643643
11. Katarzyna Woźnica, Przemyslaw Biecek, *Does imputation matter? Benchmark for predictive models*, 2020. doi: 10.48550/arXiv.2007.02837
12. A. Popov, O. Makarenko, and P. Bidyuk, "Rozv'iazannia zadachi zapovnennia propuskiv danykh alternatyvnymy metodamy pry stvorenni prohnoznykh modelei [Solving missing data imputation problem using alternative methods in predictive model creation]," *Proceedings of the II All-Ukrainian Scientific and Practical Conference "System Sciences and Informatics", December 4–8, 2023, Kyiv: National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"*, pp. 201–206.

Received 10.04.2024

INFORMATION ON THE ARTICLE

Andrii Yu. Popov, ORCID: 0009-0001-4783-7401, Educational and Research Institute for Applied System Analysis of the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Ukraine, e-mail: popovandrii1403@gmail.com

ПОРІВНЯННЯ ЕФЕКТИВНОСТІ МЕТОДІВ ЗАПОВНЕННЯ ПРОПУЩЕНИХ ДАНИХ ПІД ЧАС РОЗРОБЛЕННЯ МОДЕЛЕЙ ПРОГНОЗУВАННЯ / А.Ю. Попов

Анотація. Наявність пропущених даних є поширеною проблемою в аналізі даних та машинному навчанні. У роботі проаналізовано залежності якості прогнозування моделей машинного навчання від використаних методів оброблення пропущених даних на етапі підготовки даних до навчання моделей. Досліджуваними методами є аналіз повних спостережень, заповнення середнім та дві реалізації методу множинного заповнення — мовами Python та R. Обраними класифікаторами є логістична регресія, метод випадкового лісу, метод опорних векторів та Light Gradient Boosting Machine (LGBM). Якість прогнозних моделей оцінюється за метриками accuracy, precision та recall. Розглянуто два набори даних із задачами класифікації, що мають різні цільові метрики. Найкращі результати досягнуто з використанням алгоритму множинного заповнення у реалізації мовою R у поєднанні з класифікаторами випадкового лісу та LGBM.

Ключові слова: пропущені дані, методи заповнення, прогнозні моделі, машинне навчання.

UDC 517.925
DOI: 10.20535/SRIT.2308-8893.2025.1.04

IDENTIFICATION OF NONLINEAR SYSTEMS WITH PERIODIC EXTERNAL ACTIONS (PART III)

V. GORODETSKYI

Abstract. The article considers the problem of identifying a mathematical model in the form of a system of ordinary differential equations. The identified system can have constant and periodic coefficients. The source of information for solving the problem is time series of observed variables. The article studies the effect of uniformly distributed noise on the identification result. To solve the problem, the algorithm proposed by the author in previous works was used. It is shown that the method has different sensitivity to noise depending on which of the observed variables is contaminated with noise. The implementation of the method is illustrated by numerical examples of identifying nonlinear differential equations with polynomial right-hand sides.

Keywords: identification, ordinary differential equation, periodic coefficient, constant coefficient, uniformly distributed noise.

INTRODUCTION

When developing mathematical methods for studying various physical systems, it is necessary to evaluate reliability of their results if applied to systems in real world. One of the common tasks in applied mathematics is the problem of identifying a mathematical model of a certain process. The initial data for solving this problem can be the observed variables of the process. If we study real systems, the results of measurements of the observed variables may contain noise [1–3]. This circumstance can complicate the identification of the model.

BACKGROUND AND TASK FOR RESEARCH

We follow the results obtained in [4; 5]. There, the problem of identifying a system of n ordinary differential equations with constant and periodic coefficients $c_{ij}(t)$ $(i=1,...,n;\ j=1,...,m)$ was considered. The initial data for identification was time series of observed variables $x_i(t)$, $t\in[0;t_e], t_e>0$. To solve the problem, the theorem proved in [4] was used. According to this theorem, simple relationships that are used to identify equations with constant coefficients can be used to identify differential equations with periodic coefficients. For this purpose, the calculations must use the values of the functions $x_i(t)$ at moments of time t_j, separated from each other by the value qT, $q\in 1,2,3,...$, where T is the period of the periodic coefficients. In other words, this time moments obey the relations

$$t_1=t_0+\tau,\ t_2=t_0+2\tau,\ ...,\ t_m=t_0+m\tau;\ \ t_0\ge 0,\ \tau>0,\ t_m\le t_e, \qquad (1)$$

where $\tau=qT$.

© *Publisher IASA at the Igor Sikorsky Kyiv Polytechnic Institute*, 2025

Examples of the method application were demonstrated in [4; 5]. In this study, we try to apply the proposed algorithm to identify equations by observed variables with noise. As in the mentioned articles, we use as an example a system of the form

$$\begin{cases} \dot{x}_1 = -x_2 - x_3, \\ \dot{x}_2 = x_1 - dx_2, \\ \dot{x}_3 = c_{30}(t) + c_{33}(t)x_3 + c_{36}(t)x_1x_3, \end{cases} \quad (2)$$

obtained on the basis of the Rössler system [6]. The parameters of the third equation of the system (2) are as following:

$$c_{30}(t) = 0.5 + 0.4\sin\left(\frac{2\pi t}{T_0}\right),\ c_{33}(t) = -20,\ c_{36}(t) = 5,\ T_0 = 2\,s.$$

The generalized structure for the purpose of identifying the desired equation has the form

$$\dot{x}_3 = c_{30}(t) + c_{31}(t)x_1 + c_{32}(t)x_2 + c_{33}(t)x_3 + c_{34}(t)x_1^2 + c_{35}(t)x_1x_2 +$$

$$+ c_{36}(t)x_1x_3 + c_{37}(t)x_2^2 + c_{38}(t)x_2x_3 + c_{39}(t)x_3^2. \quad (3)$$

Next, we will consider how adding noise to different observables affects the identification result.

IDENTIFICATION OF AN EQUATION WITH A VARIABLE $x_1(t)$ AFFECTED BY NOISE

For the study, we add noise with a uniform distribution to the observable $x_1(t)$. The noise value is $\Delta u_1 = 0.01 \cdot \Delta x_1$, $\Delta x_1 = |x_{1\max} - x_{1\min}|$, $x_{1\max}$ and $x_{1\min}$ are, respectively, the maximum and the minimum of the observable $x_1(t)$ over the studied interval of 100 s. Fig. 1 shows a fragment of the time series $x_1(t)$ with added noise.

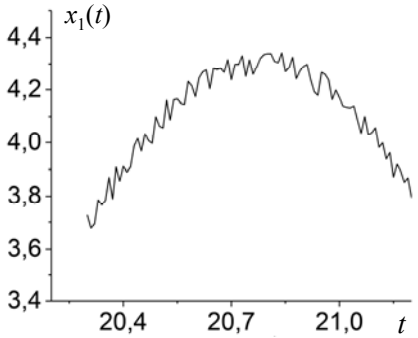


Fig. 1. Time series of $x_1(t)$ contaminated with noise

The first stage of model identification with this algorithm is to find those values of τ (see (1)) that can be equal to or multiples of the expected T. The values obtained by applying the algorithm are presented in Table 1. The table shows in bold the τ values that are repeated or multiples of other values. This means that they may be the sought-for values of T or multiples of this value. For example, the values 1.04, 3.01, 4.88, 5.30, 5.67, 10.96 are repeated. The set: 2.65, 5.30, 10.60 is also highlighted in bold because the second and the third values of it are multiples of the first one. Similarly, the value 6.00 is a multiple of 3.00. The values 2.20

and 7.70 are multiples of 1.10, which is not in the table, but which may potentially be the sought-for period. For the same reason, 6.00, 8.00, 10.00 are highlighted, which are multiples of 2.00, which is not in the table.

T a b l e 1. The result of applying the algorithm for observable $x_1(t)$ with noise

№	The τ values calculated for the coefficients of the third equation of system (2)									
	$c_{30}(t)$	$c_{31}(t)$	$c_{32}(t)$	$c_{33}(t)$	$c_{34}(t)$	$c_{35}(t)$	$c_{36}(t)$	$c_{37}(t)$	$c_{38}(t)$	$c_{39}(t)$
1	9.25	5.24	5.18	4.08	5.95	7.26	1.84	**6.00**	1.72	2.63
2	0.88	1.27	3.34	1.29	5.31	5.86	**2.65**	6.03	3.66	0.86
3	**7.70**	**10.96**	**3.00**	8.18	8.98	**6.00**	5.18	6.64	**3.01**	**5.67**
4	**1.04**	3.31	9.91	2.98	5.91	3.20	5.36	0.81	5.85	2.58
5	8.23	2.30	10.94	10.95	3.33	7.72	**5.67**	7.07	8.23	7.45
6	**3.01**	4.36	5.86	**3.01**	4.74	**5.30**	5.39	7.79	**10.60**	1.91
7	9.17	**4.88**	5.91	5.32	**10.96**	4.97	9.07	8.92	9.06	1.08
8	4.81	**10.00**	1.85	3.49	9.80	5.44	**3.01**	7.99	**6.00**	7.44
9	**4.88**	**2.20**	4.79	7.58	**8.00**	5.98	5.91	1.53	10.66	9.24
10	2.98	4.43	7.99	**5.30**	1.48	7.59	1.47	9.50	4.02	**1.04**

We have to, based on the data in Table 1, reject the excess coefficients of the desired equation and estimate the type and values of the remaining coefficients. For this, we use the second part of the algorithm. Namely, for each value selected in Table 1, we solve a system of the form

$$\mathbf{C} = \mathbf{A}^{-1}\mathbf{B}, \tag{4}$$

where $\mathbf{C}$ is the vector of the required coefficients of the third equation of system (2), and $\mathbf{B}$ is the vector of values $\dot{x}_i(t_k)$, $k = 0,...,m$, $\mathbf{A}$ is the matrix of function $f_j(\mathbf{x}(t_k))$ values, $j = 0,...,m$, $\mathbf{x} = \{x_1,...,x_n\}$. In this study we consider as functions $f_j(\mathbf{x}(t_k))$ the products of x_i in each monomial of the right-hand side of (3). We set some interval of change t_0 from (1) and obtain the calculated values of the coefficient on this interval. The resulting time series of coefficients allow us to estimate the type of coefficient and the possible value of T.

To begin with, let us try to identify zero coefficients in the analyzed equation, provided that the selected value of τ from Table 1 can be the desired period T. For example, Fig. 2 shows the time series of the calculated coefficient $c_{34}^c(t)$ for $\tau = 1.04\ s$ on an interval of 4 s.

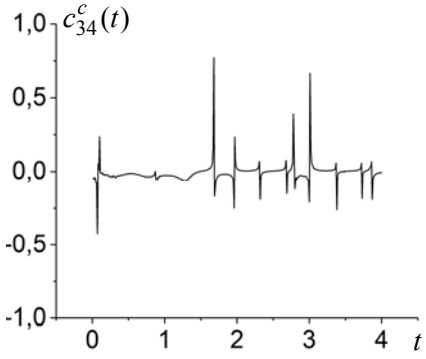


Fig. 2. Time series of calculated coefficient $c_{34}^c(t)$

As can be seen, this coefficient has values close to zero on this segment. At the same time, its greatest deviation from zero, including at singular points, is $|c_{34}^c(t)| < 1$. We will assume that if these conditions are met, this coefficient is a candidate for zeroing. Such a criterion was applied to all

coefficients for all selected τ from Table 1. The results of the analysis are presented in Table 2, where the coefficients that may be zero in the desired equation are marked with a "+" sign. Note that this table has been supplemented with the values $\tau = 1.10\ s$ and $\tau = 2.00\ s$, which, as explained above, may also be values of the period T.

Table 2. Results of the analysis of time series obtained for the selected values of τ from Table 1

№	**Possible values of *T* or its multiples**															
	1.04	1.10	2.20	7.70	2.00	6.00	8.00	10.00	3.00	3.01	2.65	5.30	10.60	4.88	5.67	10.96
c_{34}	+				+											
c_{35}					+	+			+				+	+		
c_{37}	+				+	+		+	+				+	+		+

According to Table 2, the most likely candidates for zeroing are coefficients c_{35} and c_{37}. Taking $c_{35} = c_{37} = 0$ and performing an analysis similar to the previous one, we obtain Table 3.

Table 3. The same as in Table 2 with $c_{35} = c_{37} = 0$

№	**Possible values of *T* or its multiples**									
	1.04	2.00	4.00	6.00	8.00	10.00	3.00	10.60	4.88	10.96
c_{31}					+					
c_{32}		+		+	+					
c_{34}		+	+	+	+	+				+

Note that Table 3 does not include the τ values from Table 2, for which (according to Table 2) it is not possible to determine the coefficients that may be subject to zeroing. It should be noted that the value $\tau = 4.00s$ is added to the table because it is a multiple of $\tau = 2.00s$. One can also pay attention to the graph $c_{30}(t)$ obtained, for example, at $\tau = 4.00s$ (see Fig. 3). We can already assume from it that $T = 2.00s$.

Based on Table 3, the next step should be to zero out the coefficients included in it. As a result, time series of the remaining coefficients $c_{30}^{c}(t)$, $c_{33}^{c}(t)$, $c_{36}^{c}(t)$, $c_{38}^{c}(t)$, $c_{39}^{c}(t)$ were obtained.

For further analysis, let us consider graphs $c_{38}^{c}(t)$ and $c_{33}^{c}(t)$ shown in Fig. 4. Despite the large number of points in which the calculated value of the coefficient $c_{38}^{c}(t)$ deviates significantly from zero, we can assume that $c_{38} = 0$. Graph $c_{39}^{c}(t)$ looks similar, which also allows us

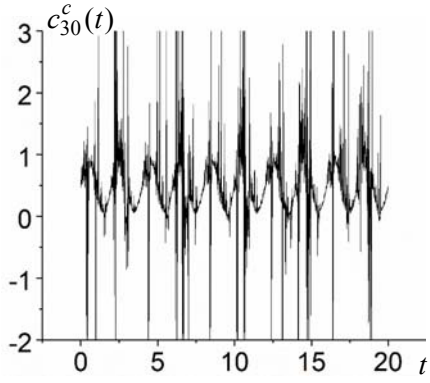


Fig. 3. Time series of calculated coefficient $c_{30}^{c}(t)$

to exclude it from the desired equation. On the contrary, most points of graph $c_{33}^{c}(t)$ clearly have values different from zero. Graph $c_{36}^{c}(t)$ looks similar to $c_{33}^{c}(t)$. Therefore, at the next step, it is advisable to zero out the coefficients c_{38} and c_{39}. As a result, this method gives us the structure of equation analogous to the apriori equation (2). Time series of calculated coefficients are shown in Fig. 5.

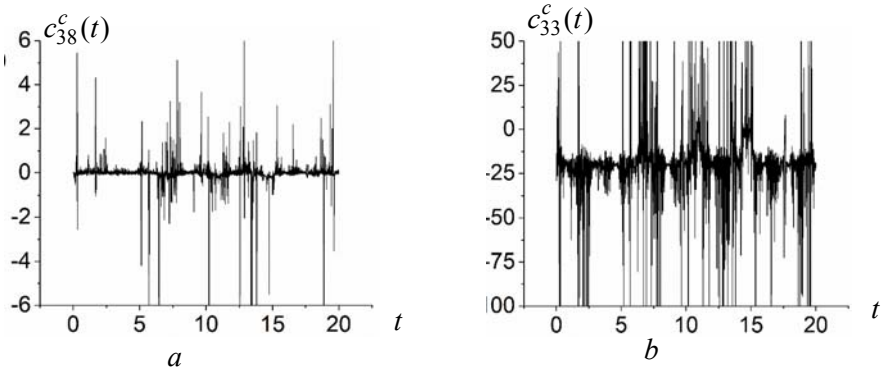


Fig. 4. Time series of calculated coefficients $c_{38}^{c}(t)$ and $c_{33}^{c}(t)$

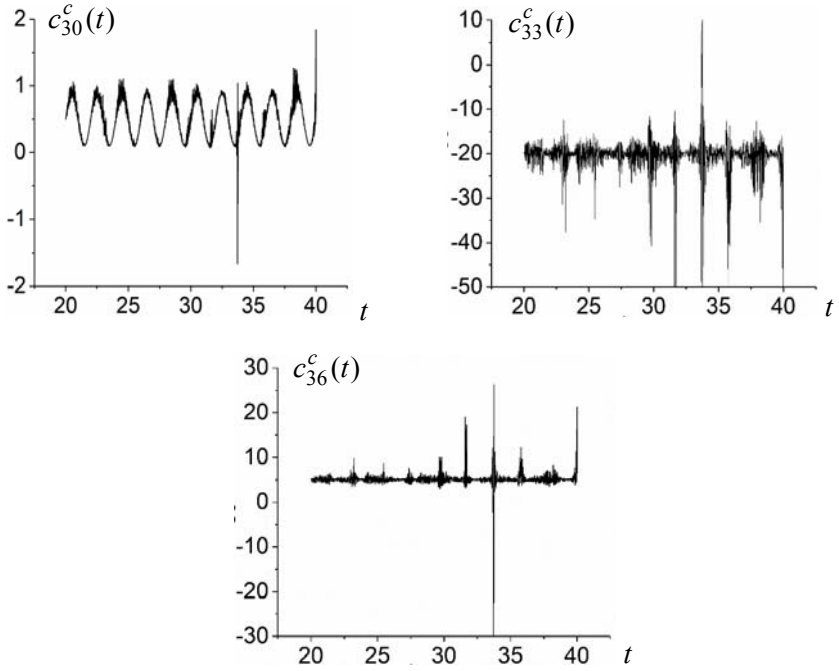


Fig. 5. Time series of calculated coefficients of the identified equation with the desired structures

IDENTIFICATION OF AN EQUATION WITH A VARIABLE $x_2(t)$ AFFECTED BY NOISE

A study similar to that performed in the previous section was also performed for a variable with the same noise level $\Delta u_2 = 0.01 \cdot \Delta x_2$. The result is shown in Table 4.

Table 4. The result of applying the algorithm for observable $x_2(t)$ with noise 1%

№	The τ values calculated for the coefficients of the third equation of system (2)									
	$c_{30}(t)$	$c_{31}(t)$	$c_{32}(t)$	$c_{33}(t)$	$c_{34}(t)$	$c_{35}(t)$	$c_{36}(t)$	$c_{37}(t)$	$c_{38}(t)$	$c_{39}(t)$
1	2.98	**4.00**	**10.00**	**4.00**	**4.00**	**8.00**	**4.00**	**10.00**	**2.00**	**4.00**
2	2.76	**8.00**	**2.00**	**8.00**	**8.00**	**10.00**	**8.00**	**2.00**	**10.00**	**10.00**
3	1.88	**10.00**	**8.00**	**10.00**	**10.00**	**2.00**	**10.00**	**4.00**	**8.00**	**8.00**
4	3.34	**2.00**	7.63	**2.00**	**2.00**	**4.00**	**2.00**	**6.00**	**4.00**	0.86
5	8.71	1.27	**4.00**	10.95	9.80	5.91	1.88	**8.00**	10.67	9.24

Based on the corollary of Theorem [4], we can conclude that the equation has a single variable coefficient $c_{30}(t)$, and the other coefficients are constant. That is, in this case, the identification occurs in the same way as for equations without noise, see [4, 5]. Moreover, a similar result was obtained by increasing the noise level added to the observable $x_2(t)$. Fig. 6 shows this variable with noise $\Delta u_2 = 0.2 \cdot \Delta x_2$, and Table 5 demonstrates the result of applying the algorithm.

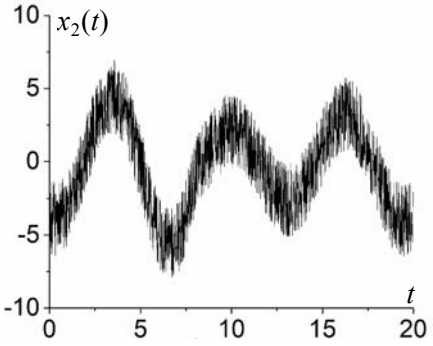


Fig. 6. Time series of $x_2(t)$ contaminated with noise 20%

Table 5. The result of applying the algorithm for observable $x_2(t)$ with noise 20%

№	The τ values calculated for the coefficients of the third equation of system (2)									
	$c_{30}(t)$	$c_{31}(t)$	$c_{32}(t)$	$c_{33}(t)$	$c_{34}(t)$	$c_{35}(t)$	$c_{36}(t)$	$c_{37}(t)$	$c_{38}(t)$	$c_{39}(t)$
1	10.75	**8.00**	**6.00**	**8.00**	**8.00**	**4.00**	10.35	**6.00**	**6.00**	**8.00**
2	1.20	**10.00**	**4.00**	1.46	**10.00**	**6.00**	**8.00**	**10.00**	**10.00**	**10.00**
3	1.05	**2.00**	**10.00**	**10.00**	**2.00**	**10.00**	**10.00**	**8.00**	**4.00**	6.73
4	6.89	5.29	**8.00**	0.72	**4.00**	**8.00**	0.95	**2.00**	**8.00**	**2.00**
5	2.60	1.86	**2.00**	8.75	**6.00**	**2.00**	**4.00**	**4.00**	**2.00**	**4.00**

Based on the comparison of these two tables, it can be concluded that the algorithm has low sensitivity to noise in this case. This can also be illustrated by Table 6, which presents the calculated values of the constant coefficients at 20% noise for two different t_0. It is clear from the table that in the structure (3) all terms that include the variable x_2 are subject to zeroing. Therefore, in the subsequent steps of identification, only the observables x_1 and x_3 will be used, which in this case are noise-free. This simplifies the task.

Table 6. The calculated values of the constant coefficients at 20% noise for two different t_0

t_0	Coefficients								
	c_{31}	c_{32}	c_{33}	c_{34}	c_{35}	c_{36}	c_{37}	c_{38}	c_{39}
$t_{01} = 0.15s$	-1.307 10^{-5}	-7.991 10^{-6}	-20.008	-1.288 10^{-6}	2.425 10^{-6}	5.002	1.367 10^{-7}	3.321 10^{-4}	4.862 10^{-4}
$t_{02} = 0.4s$	-3.51 10^{-6}	2.514 10^{-6}	-20.026	8.848 10^{-7}	-3.002 10^{-7}	5.006	3.638 10^{-7}	6.584 10^{-4}	1.585 10^{-3}

IDENTIFICATION OF AN EQUATION WITH A VARIABLE $x_3(t)$ AFFECTED BY NOISE

Unlike the previous case, the variable x_3 is present in the desired equation, both in the right-hand side and in the left-hand side in the form of a time derivative. A fragment of the noisy series $x_3(t)$ with $\Delta u_3 = 0.01 \cdot \Delta x_3$ is shown in Fig. 7, *a*.

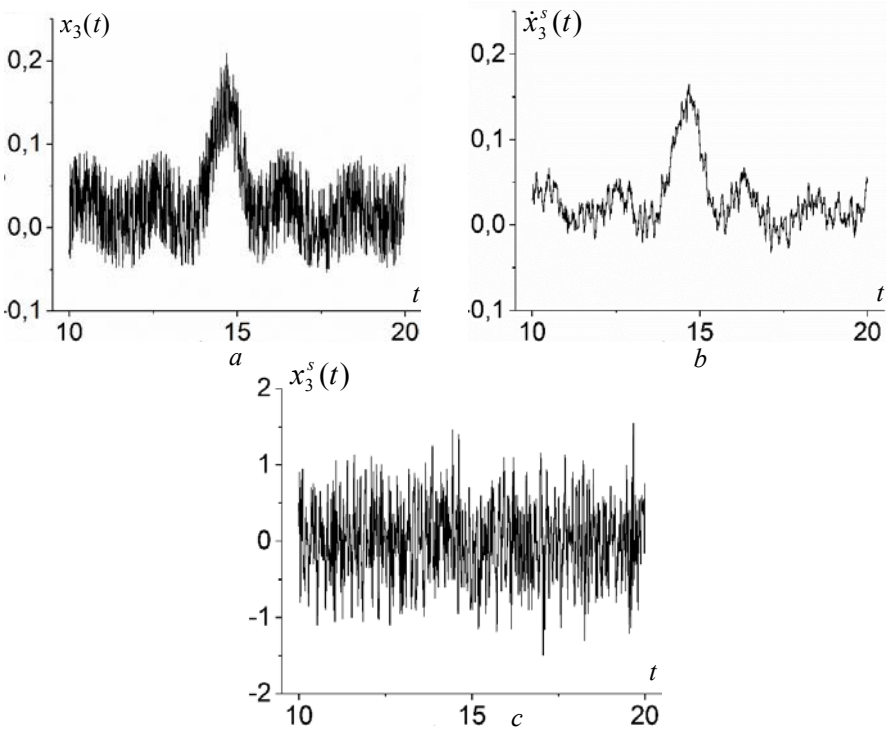


Fig. 7. Time series of $x_3(t)$: *a* — fragment of time series $x_3(t)$ contaminated with noise 1%; *b* — the same fragment after smoothing; *c* — time derivative of graph (*b*)

In order to reduce noise, smoothing was performed using the moving average method according to the formula

$$x_{3\nu}^s = \frac{x_{3\nu-3} + x_{3\nu-2} + x_{3\nu-1} + x_{3\nu} + x_{3\nu+1} + x_{3\nu+2} + x_{3\nu+3}}{7}, \qquad (5)$$

where $x_{3\nu}^s$ is the value of the function $x_3(t)$ at the point with the number ν after smoothing by formula (5). A fragment of the function $x_3^s(t)$ is shown in Fig. 7, *b*. To form the vector **B** of the left sides in the system (4), it is necessary to perform numerical differentiation of the function $x_3^s(t)$. For this, the formula

$$\dot{x}_{3\nu}^s = \frac{x_{3\nu+1}^s - x_{3\nu-1}^s}{2\Delta t}$$

was used, where Δt is a step of time series $x_3(t)$ representation. The time series $\dot{x}_{3\nu}^s(t)$ is shown in Fig. 7, *c*, which demonstrates that noise is significantly amplified when numerical differentiation is computed.

Unfortunately, the application of the algorithm did not allow us to obtain an adequate result. Obviously, the reason for this is insufficient smoothing of the time series noise and the appearance of significant computational noise during numerical differentiation.

DISCUSSION AND CONCLUSIONS

From the previous sections it is clear that the most difficult identification is when the variables $x_1(t)$ and $x_3(t)$, which are included in the desired equation, are contaminated with noise. The results obtained can be explained using Cramer's rule. When determining the coefficients of equation (3), it will have the form:

$$c_{3j} = \frac{\det(\mathbf{A}_j)}{\det(\mathbf{A})},\ j = 0,...,m,$$

where $\det(\mathbf{A})$ is the main determinant of the system of linear algebraic equations (4), formed taking into account conditions (1), $\det(\mathbf{A}_j)$ is the determinant obtained by replacing the *j*-th column of the determinant $\det(\mathbf{A})$ with the vector $\mathbf{B}$ from (4).

Let the variable with noise be $x_1(t)$. Then, to find, for example, the coefficient c_{31}, we create matrix $\mathbf{A}_1$ by replacing column 1 with $\mathbf{B}$ in matrix $\mathbf{A}$, which in this case consists of the time derivatives of the values of $x_3(t)$. That is, we replace the column of values of noisy $x_1(t)$ with the column $\mathbf{B}$ without noise. Here we assume that the differentiation of the variable $x_3(t)$ without noise is performed correctly, without significant errors. Therefore, such a replacement, at a minimum, should not increase the error in calculating c_{31}.

On the contrary, if the observed variable with noise is $x_3(t)$, then when differentiating it numerically, computational noise will appear, see Fig. 7, *c*. Then, when calculating, for example, the coefficient c_{33}, we replace the column number 3 in $\mathbf{A}$ with vector $\mathbf{B}$. The column number 3 of $\mathbf{A}$ initially contains the variable $x_3(t)$, which contains noise. However, comparing Fig. 7, *b* and 7, *c*, we see that the noise in the column $\mathbf{B}$ is much greater and can of course be a source of significant errors.

Taking into account the above, for successful identification of systems from time series of observations with noise, sometimes it is necessary to use various noise filtering methods [7–12] that are more effective than (5). The use of more effective, although more complex, methods of numerical differentiation [13–15] is also justified.

REFERENCES

1. P.R. Deboeck, S.M. Boker, "Modeling noisy data with differential equations using observed and expected matrices," *Psychometrika*, vol. 75, no. 3, pp. 420–437, Sept. 2010. doi: 10.1007/S11336-010-9168-2
2. H. Liang, H. Miao, and H. Wu, "Estimation of constant and time-varying dynamic parameters of HIV infection in a nonlinear differential equation model," *Ann. Appl. Stat*., no. 4(1), pp. 460–483, Mar. 1, 2010. doi: 10.1214/09-AOAS290
3. J.O. Ramsay, G. Hooker, D. Campbell, and J. Cao, "Parameter estimation for differential equations: a generalized smoothing approach," *J. Roy. Statistic. Soc., Ser. B*, vol. 69, no. 5, pp. 741–796, Nov. 2007. doi: 10.1111/j.1467-9868.2007.00610.x

4. V. Gorodetskyi, “Identification of nonlinear systems with periodic external actions (Part 1),” *System Research & Information Technologies*, no. 3, pp. 93–106, 2024. doi: 10.20535/SRIT.2308-8893.2024.3.06
5. V. Gorodetskyi, “Identification of nonlinear systems with periodic external actions (Part 2),” *System Research & Information Technologies*, no. 4, pp. 66–76, 2024. doi: 10.20535/SRIT.2308-8893.2024.4.05
6. O.E. Rössler, “An equation for continuous chaos,” *Phys. Lett. A.*, vol. 57, no. 5, pp. 397–398, 1976. doi: 10.1016/0375-9601(76)90101-8
7. M. Mafi, H. Martin, M. Cabrerizo, J. Andrian, A. Barreto, and M. Adjouadi, “A comprehensive survey on impulse and Gaussian denoising filters for digital images,” *Sign. Proc*., vol. 157, pp. 236–260, Apr. 2019. doi: 10.1016/j.sigpro.2018.12.006
8. L. Tan, J. Jiang, “Finite Impulse Response Filter Design,” in *Digital Signal Processing (Third Edition): Fundamentals and Applications*, 2019, pp. 229–313.
9. J.P. do Vale Madeiro, P.C. Cortez, J.M. da Silva Monteiro Filho, and P.R.F. Rodrigues, “Techniques for Noise Suppression for ECG Signal Processing,” in *Developments and Applications for ECG Signal Processing. Modeling: Segmentation, and Pattern Recognition*, pp. 53–87, 2019. doi: 10.1016/B978-0-12-814035-2.00009-8
10. S.V. Vaseghi, *Advanced Digital Signal Processing and Noise Reduction. (Fourth Edition)*. London, UK: Wiley and Sons, 2008.
11. Y. Chen, S. Fomel, “Random noise attenuation using local signal-and-noise orthogonalization,” *Geophys.*, vol. 80, no. 6, pp. WD1–WD9, Nov.-Dec., 2015. doi: 10.1190/GEO2014-0227.1
12. T. Kelemenová, O. Benedik, and I. Koláriková, “Signal noise reduction and filtering,” *Acta Mechatronica*, vol. 5, no. 2, pp. 29–34, Jun. 2020. doi: 10.22306/am.v5i2.65
13. I. Knowles, R.J. Renka, “Methods for numerical differentiation of noisy data,” *Electr. J. Dif. Eq.,* Conference 21, pp. 235–246, 2014, presented at Variational and Topological Methods: Theory, Applications, Numerical Simulations, and Open Problems, 2012.
14. R. Chartrand, “Numerical differentiation of noisy, nonsmooth data,” *Intern. Schol. Res. Netw. Appl. Math.,* vol. 2011, Art. no. 164564. doi: 10.5402/2011/164564
15. K. Ahnert, M. Abel, “Numerical differentiation of experimental data: local versus global methods,” *Comput. Phys. Commun.* vol. 177, no. 10, pp. 764–774, Nov. 2007. doi: 10.1016/j.cpc.2007.03.009

Received 14.01.2025

INFORMATION ON THE ARTICLE

Viktor G. Gorodetskyi, ORCID: 0000-0003-4642-3060, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”, Ukraine, e-mail: v.gorodetskyi@ukr.net

ІДЕНТИФІКАЦІЯ НЕЛІНІЙНИХ СИСТЕМ З ПЕРІОДИЧНИМИ ЗОВНІШНІМИ ДІЯМИ (Частина ІІІ) / В.Г. Городецький

Анотація. Розглянуто проблему ідентифікації математичної моделі у вигляді системи звичайних диференціальних рівнянь. Ідентифікована система може мати сталі та періодичні коефіцієнти. Джерелом інформації для розв’язання поставленої задачі є часові ряди спостережуваних змінних. Досліджено вплив шуму з рівномірним розподілом на результат ідентифікації. У ході дослідження використовувався алгоритм, запропонований автором у попередніх працях. Розглянуто особливості застосування цього алгоритму для даної задачі. Показано, що метод має різну чутливість до шуму залежно від того, яка із спостережуваних змінних забруднена шумом. Реалізацію методу проілюстровано чисельними прикладами ідентифікації нелінійних диференціальних рівнянь із поліноміальними правими частинами.

Ключові слова: ідентифікація, звичайне диференціальне рівняння, періодичний коефіцієнт, сталий коефіцієнт, шум із рівномірним розподілом.

UDC 004.942:537.525:621.325
DOI: 10.20535/SRIT.2308-8893.2025.1.05

NUMERICAL ALGORITHM FOR CALCULATION OF THE VACUUM CONDUCTIVITY OF A NON-LINEAR CHANNEL FOR TRANSPORTING A SHORT-FOCUS ELECTRON BEAM IN THE TECHNOLOGICAL EQUIPMENT

I. MELNYK, A. POCHYNOK, M. SKRYPKA

Abstract. In the article, based on solving the equations of vacuum technology, an iterative algorithm for calculating vacuum conductivity and the geometric parameters of a curvilinear channel for transporting a short-focus electron beam is proposed and studied. For such a type of channel, the dependence of its radius on the longitudinal coordinate is described by a power function. The proposed algorithm is based on the numerical solution of a set of nonlinear equations using the Steffensen method. The results of the test calculations are presented. The provided tests confirm the stability of the proposed algorithm's convergence for correct pressure and pumping speed values in electron-beam technological equipment. Such curved transport channels can be used in electron beam equipment based on high-voltage glow discharge electron guns intended for welding, melting metals, and the deposition of thin films. The criterion for the optimal geometry of a nonlinear channel is the minimum power loss of the electron beam during its transportation while ensuring the required pressure drop between the discharge and technological chambers.

Keywords: electron beam, electron beam technologies, electron beam transportation, nonlinear electron beam transportation channel, vacuum conductivity of the transportation channel, high-voltage glow discharge electron gun, vacuum technology equation, set of nonlinear equations, Steffensen method.

INTRODUCTION

The development of electron beam technologies is very important today, and such advanced technologies are widely applied in different branches of industry, including metallurgy, mechanical engineering, energetic industry, electronics, instrument-making industry, automotive industry, as well as aircraft and space industry [1–44]. Generally, it is caused by the many important advantages of the electron beam as a technological instrument, which are as follows [1–10].

1. The total power and power density of the electron beam are extremely high. Naturally, in industrial technological electron guns, the total power can reach hundreds of kW, and the power density can be up to 10^9 W/m^2.

2. Ease of control and changing of the geometric and energy focal parameters of the electron beam using electric and magnetic fields.

3. Carrying out a technological operation under conditions of medium and high vacuum, which ensures the repeatability of technological process parameters during their control, the purity of the treated materials, and, as a result, the high quality of products.

In particular, electron beam methods for purifying refractory metals and ceramic materials are widely used today in metallurgy [36–41]. For example, with

© *Publisher IASA at the Igor Sikorsky Kyiv Polytechnic Institute*, 2025

the development of modern electronics, it is now very important to obtain pure silicon for use in electronics for the production of effective and high-quality microchips [36]. Advanced technologies for refining refractory metals are very important for the production of reliable details for the automotive, aircraft, and space industries [40; 41].

Recently, in the mechanical engineering, aviation, and space industries, electron beam technologies for three-dimensional metal printing have found wide application and are gradually becoming quite cheap, highly efficient, and allow the economical use of metal raw materials and electricity [21–25]. Such technologies also make it possible to obtain high-strength and reliable parts for the chemical industry, aviation, and space industries. Usually, the mechanical and chemical properties of metals produced using three-dimensional printing are unique, and it is really impossible to obtain metals of such quality using traditional metallurgy methods [42–44].

In the electronics industry, it is effective to use the technology of welding with point-focus [14–19] and profile [20–22] electron beams for sealing the housings of electronic devices and welding metal and ceramic contacts. For example, in papers [14; 15], the possibility of peer-reviewed welding of contacts of cryogenic electronic devices with a short-duration pulsed point-focus electron beam has been considered [16–19]. Profile beam welding is a very low-cost technology that provides high productivity and can be easily automated [20–22].

Another effective use of electron beam technologies in modern production is the application of stoichiometric ceramic coatings, which contain active gases, in particular oxides, carbides, sulfides, nitrides, etc. [23–32]. For example, multilayer coatings made of rare earth metal oxides are effective for forming insulating coatings on electric vehicle contacts, as well as heat-protective coatings for internal combustion engines and jet engines [23–32]. Carbide and sulfide dielectric films are used in microelectronics for the manufacture of high-quality capacitors as well as for transmitting and receiving devices for communication microwave electronic equipment [34; 35]. It was shown in the papers [11; 12; 23–32], that the best way to obtain such coatings is electron beam evaporation in a vacuum with stimulation of a chemical reaction between metal vapor and the residual gas by igniting an auxiliary low-voltage discharge.

It is clear, that the main part of any electron beam technological equipment is an electron gun, which ensures the generation of an electron beam with specified energy and geometric parameters for predetermined conditions of the technological process. Therefore, when one designing the electron beam technological installation, an extremely important engineering aspect is always the coordination of the physical operating conditions of the electron gun with the parameters of the technological process that is being performed [1–10]. It is especially important to ensure an appropriate pressure range in the area of electron beam formation and in the area of technological operation.

In general, it should be noted that today the elaboration of electron guns for technological use is carried out mainly in two directions: improving the designs of traditionally used guns with heated cathodes [1–10] and the development of electron guns, the generation of beams in which is carried out in a fundamentally different way, for example, through field emission [1–10], photoemission [1–10] or the ignition of various types of gas discharges [11–13; 45–47].

HIGH-VOLTAGE GLOW DISCHARGE ELECTRON GUNS AND THE PARTICULARITIES OF THEIR OPERATION IN INDUSTRIAL TECHNOLOGICAL EQUIPMENT

In the papers [11–13; 23–32], it was pointed out that in the physical conditions of low vacuum, on the order of 0.1–10 Pa, the High-Voltage Glow Discharge (HVGD) electron guns usually operate stably and very reliably. The undeniable advantages of such types of electron sources in comparison with traditionally used electron guns with heated cathodes are the following [11–13; 23–32].

1. The ability of HVGD electron guns to operate in a soft and medium vacuum in the environment of various gases, in particular noble and active ones. This makes it easy to coordinate the gun parameters with the required parameters of the technological process. Typically, the pressure in the discharge chamber of the gun lies in the range of 1–10 Pa. For welding and melting electron beam equipment, it leads to a very important technical and economic effect of simplification of technological installations [16; 17], and the technological process of deposition of high-quality ceramic films in the medium of active gas is generally extremely difficult to implement without the use of HVGD electron guns [23–32]. To coordinate the operating parameters of HVGD electron guns and in the area where technological operation is being performed, pressure decoupling is used through an electron beam transport channel with a limited radius. This makes it possible to maintain the required value of pressure in the HVGD combustion area and in the technological chamber with a controlled injection of gas into the electron gun and continuous pumping of the technological chamber [48; 49].

2. The simplicity of the design of HVGD electron guns and the possibility of their assembly and disassembly in order to replace used components, in particular, the HVGD cold cathode [11; 12]. Rough and precise estimates of the operating physical conditions of the cold HVGD cathode and its surface temperature are given in the papers [16; 17].

3. Ease of debugging HVGD electron guns and ensuring their operation as part of technological electron beam installation [11; 12]. The only point related to the complexity of assembling HVGD electron guns is ensuring the alignment of the design parts [21; 22].

4. The relative simplicity of vacuum evacuation technological equipment, since there is no need to ensure the operation of the HVGD electron guns under high vacuum physical conditions [11; 12].

5. Ease of regulation of the electron beam power at a stable value of the accelerating voltage. There are two ways existed for control the power of the electron beam: aerodynamic, through a controlled change of pressure in the gap of HVGD lighting by regulating the gas flow [48; 49], and electrical, through the ignition of an auxiliary discharge and changing the ion concentration in the anode plasma [17–19]. A generalized description of the operating algorithm of the digital current control system of the HVGD electron gun based on the methods of discrete mathematics and the theory of finite state machines is given in [50].

THE STATE OF DEVELOPMENT OF TECHNOLOGICAL EQUIPMENT WITH HIGH-VOLTAGE GLOW DISCHARGE ELECTRON GUNS AND THE CONSIDERED PROBLEM OF SIMULATION OF TRANSPORT CHANNEL

However, despite the technical and economic advantages of HVGD electron guns described above, there are also certain technical difficulties associated with their

industrial application. They are primarily related to the coordination of the physical operating conditions of HVGD guns with the pressure parameters of the technological process [11; 12]. For welding electron guns and guns intended for melting metals and ceramic materials for the purpose of cleaning them, it is very important to separate the HVGD combustion zone from the zone of product processing [16; 17; 36–41], and for the process of deposition ceramic coatings, it is important to ensure the required pressure in the HVGD combustion zone and in the area of metal's vapor interaction with the operation gas [23–32]. Therefore, usually, to ensure stable operation of HVGD electron guns as part of technological equipment, a pressure decoupling is used between the HVGD combustion area and the area of the technological operation. For this purpose, electron beam transport channels with a limited cross-**sectional** radius are usually used. Then the pressure difference between the HVGD region and technological chamber is ensured through a controlled injection of gas into the electron gun and continuous pumping of the technological chamber. The corresponding block diagram of the electron beam technological installation is given in [48; 49], in the simplified form it is presented at Fig. 1 [48; 49].

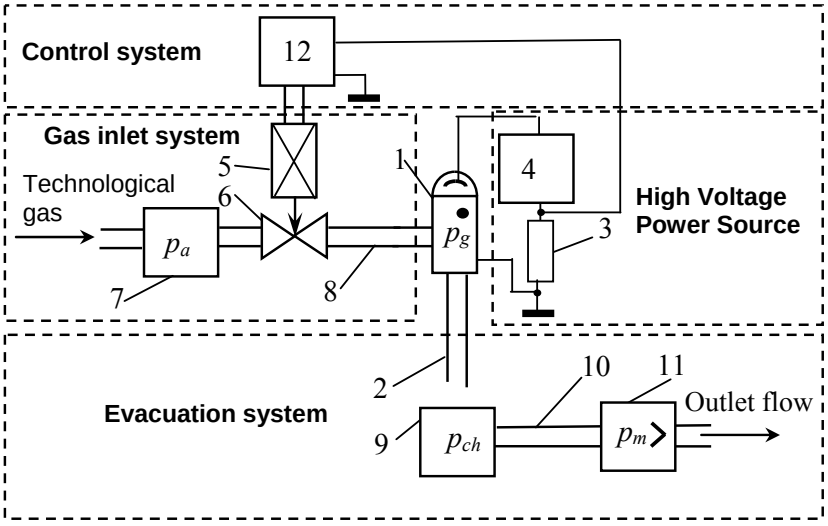


Fig. 1. Block diagram of the pumping and power supply system for an electron beam installation with a HVGD gun: 1 — HVGD electron gun; 2 — channel for guiding an electron beam into the technological chamber; 3 — gun current sensor; 4 — high voltage power supply; 5 — electromagnetic valve for inputting gas into the HVGD electron gun; 9 — technological chamber; 10 — pumping channel; 11 — vacuum pump; 12 — electronic or microcontroller system for automatically changing the gun current; p_a — atmospheric pressure; p_{ch} — pressure in the technological chamber; p_g — pressure in the HVGD electron gun; p_m — minimum pressure in the technological chamber that can be provided by applied pumping means

Typically, transport channels of cylindrical and conical cross-sections are used to guide a short-focus electron beam. Corresponded relations of vacuum technology for calculation the conductivity of such channels relative to the difference of pressure and gas flow are given in the manual books [51–57].

The aim of this paper is to analyze the possibilities of using transport channels with a nonlinear cross-section. An assessment of the vacuum conductivity of a nonlinear channel with power dependence of its radius r on longitudinal coordinate z under the physical conditions of carrying out technological operations for the deposition of ceramic films and coatings has been provided. For electron beam

welding and melting equipment, similar estimates can also be used [1–10; 51–57]. For providing generalized analyze of considered nonlinear function the standard mathematical approaches of function theory and methods for calculating integrals of power functions has been applied [58; 59]. To determine the conductivity of the vacuum channel, well-known equations of vacuum technology were used, and the required length of the channel was determined by numerically solving the complex nonlinear equation using the Steffensen method [60–64].

BASIC ANALYTICAL RELATIONS AND FORMALIZING THE OPTIMIZATION TASK FOR CHOOSING THE GEOMETRY PARAMETERS OF A NON-LINEAR TRANSPORT CHANNEL

In the general case, to determine the pressure distribution along the length of an axisymmetric transportation channel under the condition of the molecular regime of gas flow in it, the basic equation of vacuum technology and the Knudsen equation are used [47–53]. The corresponding system of algebraic and integral equations is written in general form as follows [47–53]:

$$p_{ch} = \frac{p_g U_{ch}}{U_{ch} + S_p}; U_{ch} = \frac{4\bar{v}}{3\int\limits_0^{l_{ch}} \frac{Hdl}{S_{cr}^2}}; \bar{v} = \sqrt{\frac{8R_0 T}{\pi M}}, \quad (1)$$

where U_{ch} is the vacuum conductivity of the channel; H is the perimeter of the transportation channel in cross section; S_{cr} is its area; R_0 is the universal gas constant; T is the gas temperature; M is its molecular weight of gas atoms; p_g and p_{ch} are the pressure in the gun volume and in the technological chamber, according to Fig. 1; U_{ch} is the vacuum conductivity of the transportation channel; S_p is the speed of the pumping system; $\bar{v}$ is the average thermal velocity of movement of gas molecules; l_{ch} is the length of the channel.

At a relatively high pressure in the discharge chamber of the HVGD gun, an intermediate gas flow regime is observed in the electron beam transport channel. In this case, to calculate the channel conductivity, the corresponding correction factor J is introduced [47–53]:

$$J = \frac{1 + 202(R_{\text{in}} + R_{\text{out}})\bar{p} + 2653((R_{\text{in}} + R_{\text{out}}))^2}{1 + 236(R_{\text{in}} + R_{\text{out}})}; \; U_i = JU_m, \quad (2)$$

Using relations (1), it is possible to determine analytically the vacuum conductivity of where the index m corresponds to the molecular gas flow regime one and the index i for the intermediate one; R_{in} — input radius of beam transporting channel; R_{out} — it output radius correspondently; $\bar{p}$ is the average pressure in the transportation channel; J is a semi-empirical coefficient for listing the conductivity value.

a nonlinear channel for transporting an electron beam with an exponent of $1/\alpha$. The cross section of such a channel, depending on the longitudinal coordinate z, is determined as follows:

$$r(z) = A(z + z_0)^{(1/\alpha)}, \tag{3}$$

where

$$A = \left(\frac{l_{ch}}{R_{\text{out}}^{\alpha} - R_{\text{in}}^{\alpha}} \right)^{(1/\alpha)}, \quad z_0 = \left(\frac{R_{\text{in}}}{A} \right)^{\alpha} = \frac{R_{\text{in}}^{\alpha}(R_{\text{out}}^{\alpha} - R_{\text{in}}^{\alpha})}{l_{ch}}, \tag{4}$$

or

$$r(z) = \left(\left(\frac{l_{cn}}{R_{\text{out}}^{\alpha} - R_{\text{in}}^{\alpha}} \right) \left(z + \frac{R_{\text{in}}^{\alpha}(R_{\text{out}}^{\alpha} - R_{\text{in}}^{\alpha})}{l_{ch}} \right) \right)^{(1/\alpha)}. \tag{5}$$

Taking into account the well-known rules of integration of exponential function, solving of integral equation of system (1) giving the following result:

$$U_{ch}(R_{\text{in}}, R_{\text{out}}, l_{ch}, \alpha) = \begin{cases} \dfrac{8\bar{v}\left(\dfrac{\alpha-3}{\alpha}\right)}{3\pi A\left((l_{ch} + z_0)^{\frac{\alpha-3}{\alpha}} - z_0^{\frac{\alpha-3}{\alpha}} \right)}, & \alpha \neq 3; \\ \dfrac{8\bar{v}}{3\pi A(\ln(l_{ch} + z_0) - \ln(z_0))}, & \alpha = 3, \end{cases}$$

or in the form of arithmetic-logic relation [65]:

$$U_{ch}(R_{\text{in}}, R_{\text{out}}, l_{ch}, \alpha) = \frac{8\bar{v}}{3\pi} \left((\alpha = 3) \cdot \frac{1}{A(\ln(l_{ch} + z_0) - \ln(z_0))} + \right.$$

$$\left. + ((\alpha < 3) \,|\, (\alpha > 3)) \cdot \frac{\left(\dfrac{\alpha-3}{\alpha}\right)}{A\left((l_{ch} + z_0)^{\frac{\alpha-3}{\alpha}} - z_0^{\frac{\alpha-3}{\alpha}} \right)} \right). \tag{6}$$

Taking into account relations (3), (4), as well as first relation of set of equations (1), one can obtain the following nonlinear relation for the value l_{ch} relatively to transversal variable z:

$$l_{ch}(z) = \begin{cases} \dfrac{8\bar{v}\left(\dfrac{\alpha-3}{\alpha}\right)(p_{ch} - p_g)}{3\pi A J p_{ch} S_p\left((l_{ch} + z_0)^{\frac{\alpha-3}{\alpha}} - z_0^{\frac{\alpha-3}{\alpha}} \right)}, & \alpha \neq 3; \\ \dfrac{8\bar{v}(p_{ch} - p_g)}{3\pi A J p_{ch} S_p(\ln(l_{ch} + z_0) - \ln(z_0))}, & \alpha = 3, \end{cases}$$

or, in the form of arithmetic-logic relation [65]:

$$l_{ch}(\mathrm{z}) = \frac{8\bar{v}(p_{ch} - p_g)}{3\pi J p_{ch} S_p}\left((\alpha < 3) \,|\, (\alpha > 3) \frac{\left(\frac{\alpha - 3}{\alpha}\right)}{A\left((l_{ch} + z_0)^{\frac{\alpha-3}{\alpha}} - z_0^{\frac{\alpha-3}{\alpha}} \right)} + \right.$$

$$\left. + (\alpha = 3) \cdot \frac{1}{A \ln(l_{ch} + z_0) - \ln(z_0)} \right). \quad (7)$$

In further considerations, let us assume that $\alpha > 3$. Therefore, the special case $\alpha = 3$ in arithmetic-logic relation (7) is out of consideration. To solve the nonlinear equation systems (5), (6) with respect to the parameter l_{ch}, let we introduce the corresponding auxiliary variables:

$$a = R_{\text{out}}^{\alpha}; b = R_{\text{in}}^{\alpha}. \quad (8)$$

With this substitution the analytical relations (4), (5) are rewritten as follows:

$$b^2 - b(a + (r(z))^{\alpha}) - ((r(z))^{\alpha} a + z l_{ch}) = 0. \quad (9)$$

Clear, that (9) is the quadratic equation relatively to parameter *b* and for fixed value of *a* and knowing set of (*r*, *z*) coordinate it can be solved analytically.

Let formulate the task of finding nonlinear channel parameters as the task of optimization [61; 62]. Assume, that optimization is provided by four geometry parameters of channel, namely: R_{in}, R_{out}, α and l_{ch}. In such conditions for four basic points $\mathbf{P}_1(r_1, z_1)$, $\mathbf{P}_2(r_2, z_2)$, $\mathbf{P}_3(r_3, z_3)$ and $\mathbf{P}_4(r_4, z_4)$ the relations (8), (9) are rewritten as the following set of nonlinear equations, which can be solved numerically:

$$\begin{cases} R_{\text{in}}^{2\alpha} - R_{\text{in}}^{\alpha}(R_{\text{out}}^{\alpha} + (r_1(z_1))^{\alpha}) - ((R_{\text{out}} r_1(z_1))^{\alpha} + z_1 l_{ch}) = 0; \\ R_{\text{in}}^{2\alpha} - R_{\text{in}}^{\alpha}(R_{\text{out}}^{\alpha} + (r_2(z_2))^{\alpha}) - ((R_{\text{out}} r_2(z_2))^{\alpha} + z_2 l_{ch}) = 0; \\ R_{\text{in}}^{2\alpha} - R_{\text{in}}^{\alpha}(R_{\text{out}}^{\alpha} + (r_3(z_3))^{\alpha}) - ((R_{\text{out}} r_3(z_3))^{\alpha} + z_3 l_{ch}) = 0; \\ R_{\text{in}}^{2\alpha} - R_{\text{in}}^{\alpha}(R_{\text{out}}^{\alpha} + (r_4(z_4))^{\alpha}) - ((R_{\text{out}} r_4(z_4))^{\alpha} + z_4 l_{ch}) = 0. \end{cases} \quad (10)$$

The criterium of optimization for the task of beam transporting in nonlinear guiding channel is minimum current loses of electron beam in the case of providing required difference of pressure in the electron gun and in the technological chamber $p_g - p_{ch}$, correspond to Fig. 1 [1–10]. For defining current losses in the transporting channel, the boundary trajectory of electron beam has to be calculated. In such case beam losses on the small elementary range of longitudinal coordinate *dz* are defined as follows [1–10]:

$$\beta_b(n) = \sqrt{\frac{I_b(n)}{2\pi j_0(n)}}; \quad \frac{I_b(n)}{\pi r_b^2(n)} = j_0 \exp\left(\frac{r_b(n)}{\beta_b(n)}\right)^2;$$

$$dI_b(n) = \pi \beta_b^2(n) j_0(n) \left\{ \exp\left[-\left(\frac{r_{ch}(n) - \beta(n)}{2} \right)^2 \right] - \exp\left[-\left(\frac{r_{ch}(n)}{2} \right)^2 \right] \right\}, \quad (11)$$

$$I_b(n+1) = I_b(n) + dI_b(n),\ \ r_{ch}(n) = r_{ch}(ndz),\ \ \ r_{ch}(n+1) = r_{ch}((n+1)dz),\ \ dz = \left[\frac{l_{ch}}{N_{it}}\right],$$

where I_b — beam current; β_b — beam parameter, which characterized the dissipation of electrons by the velocity; n — current number of iteration, N_{it} — total number of iterations [1–10]. Generally, algorithm of calculation of boundary trajectories of electron beam in the case of its' propagation in the free space with the constant pressure have been described in the papers [66–72]. Also, in this papers an advanced method of interpolation and approximation of boundary trajectories of electron beam, based on root-polynomial function and giving very small error, has been proposed and analyzed. Using such approach, the optimization task in this case is formulated as follows [61; 62]:

$$\begin{pmatrix} R_{\text{in}} \\ R_{\text{out}} \\ l_{ch} \\ \alpha \end{pmatrix} = \min\left(\sum_{z=0}^{l_{ch}} dI_b(z)\right). \qquad (12)$$

Since the channel radius, defined from relations (4), (5), is depend on channel geometry parameter, the coordinates of basic points $\mathbf{P_1} - \mathbf{P_4}$, as a result of solving set of equations (10), are correspond to criterium (12). Usually, it is enough to define 4 point on the beam boundary trajectory, for which beam radius have the maximal value and choose the channel radius by the simple relation [1–10; 45–48]:

$$r(z) = 3\beta_b r_b(z). \qquad (13)$$

Relation (13), from the point of view of the physics of HVGD [12; 45–47], is explained by the fact that with a large number of collisions of beam electrons with residual gas atoms, the longitudinal distribution of current density $j(r)$ corresponds to the Gaussian law with a very high probability, more than 99% [1–4].

Generally, analyzing beam current losses using relations (11)–(13) is the separate sophisticated problem for future research, which consideration is not the subject of this paper. However, in any case, numerical solving of the set of equations (10) with the known set of basic points $\mathbf{P_1} - \mathbf{P_4}$ coordinates is a separate complex problem because this set of equations has been formed by using the stiff power function (5) [63; 64]. The corresponding proposed algorithm for solving the set of equations (10), based on the Steffensen method [63; 64], will be described in the next part of the article.

THE NUMERICAL ALGORITHM FOR DEFINING THE GEOMETRY PARAMETERS OF A NON-LINEAR TRANSPORT CHANNEL

Having solved the quadratic equation (9) for the variable b and substituting, instead of the common variables z and r, its values z_1 and r_1, it is easy to obtain following relation for the function $f_b(a, l_{ch}, \alpha)$:

$$b = f_b(a, l_{ch}, \alpha) = \frac{1}{2}(a + r_1(z_1))^{\alpha} +$$

$$+ \sqrt{a^2 + 2(r_1(z_1))^{\alpha} + (r_1(z_1))^{\alpha} - 4(z_1 l_{ch} - a r_1(z_1)^{\alpha})}. \quad (14)$$

Further, if one rewrites relation (9) with respect to the variable a and substitutes, instead of the common variables z and r, their values z_2 and r_2, then the corresponding relation for the function $f_a(b, l_{ch}, \alpha)$ will be written as follows:

$$a = f_a(b, l_{ch}, \alpha) = \frac{z_2 l_{ch} + b(r_2(z_2))^{\alpha} - b^2}{2(r_2(z_2))^{\alpha} - b}. \quad (15)$$

Similar, from relation (9) the function for calculation the channel length l_{ch} by substitutes, instead of the common variables z and r, their values z_3 and r_3, is written as follows:

$$l_{ch} = f_l(a, b, \alpha) = \frac{(r_3(z_3))^{\alpha}(a - b) - b^2 + ab}{z_3}. \quad (16)$$

And finally, the function for defining the exponent α in relations (3), (4), through the known values a, b, l_{ch}, and substitutes, instead of the common variables z and r, their values z_4 and r_4, is written as follows:

$$\alpha = f_{\alpha}(a, b, l_{ch}) = \frac{\ln(z_4 l_{ch} - b^2 + ab) - \ln(a + b)}{\ln(r_4(z_4))}. \quad (17)$$

Taking into account relation (14)–(17), the values a, b, l_{ch}, and α, which are satisfied for set of equations (10) for given coordinates of basic points $\mathbf{P_1} - \mathbf{P_4}$, can be defined numerically using the Steffensen method of solving sets of nonlinear equations [62–64]. Corresponding numerical relations for iterative calculation of these values are written as follows:

$$b_n =$$

$$= \frac{f_b^2(a_{n-1}, l_{ch_{n-1}}, \alpha_{n-1}, r_1, z_1)}{f_b(a_{n-1} + f_a(b_{n-1} + f_b(a_{n-1}, l_{ch_{n-1}}, r_1, z_1), l_{ch_{n-1}}, \alpha_{n-1}, r_1, z_1), l_{ch_{n-1}}, \alpha_{n-1}, r_1, z_1) - f_b(a_{n-1}, l_{ch_{n-1}}, \alpha_{n-1}, r_1, z_1)}; \quad (18)$$

$$a_n =$$

$$= \frac{f_a^2(b_n, l_{ch_{n-1}}, \alpha_{n-1}, r_2, z_2)}{f_a(b_n + f_b(a_{n-1} + f_a(b_n, l_{ch_{n-1}}, r_2, z_2), l_{ch_{n-1}}, \alpha_{n-1}, r_2, z_2), l_{ch_{n-1}}, r_2, z_2) - f_a(b_n, l_{ch_{n-1}}, \alpha_{n-1}, r_2, z_2)}; \quad (19)$$

$$\alpha_n = \frac{f_{\alpha}^2(a_n, b_n, l_{ch_{n-1}}, r_3, z_3)}{f_{\alpha}((a_n + f_a(b_n, l_{ch_{n-1}}, \alpha_{n-1}, r_2, z_2)), b_n, l_{ch_{n-1}}, r_3, z_3) - f_{\alpha}(a_n, b_n, l_{ch_{n-1}}, r_3, z_3)}; \quad (20)$$

$$l_{ch_n} = \frac{f_l^2(a_n, b_n, \alpha_{n-1}, r_4, z_4)}{f_l((a_n + f_a(b_n, l_{ch_{n-1}}, \alpha_n, r_2, z_2)), b_n, \alpha_n, r_4, z_4) - f_l(a_n, b_n, \alpha_{n-1}, r_4, z_4)}, \quad (21)$$

where n is number of current iteration.

Iterative calculations using the given relations (14)–(21) have been carried out until the vacuum conductivity of the transport channel reaches a value that

provides the required pressure difference $p_g - p_{ch}$, corresponding to Fig. 1, while the channel conductivity have been calculated from relation (6). Correspondently, iterative calculations have been considered completed if the modulus of the difference in channel conductivities at the previous and current iterations did not exceed $10^{-6}\,\frac{\text{Pa}\cdot\text{m}^3}{\text{s}}$, namely:

$$\delta = \left|U_{ch_n} - U_{ch_{n-1}}\right| < 10^{-6}\,\frac{\text{Pa}\cdot\text{m}^3}{\text{s}}, \tag{22}$$

where δ is achieved accuracy of calculations on the current iteration n.

The corresponding flow chart of the described iterative algorithm of calculations, which have been carried out using relations (6; 7; 14–22), is presented in Fig. 2.

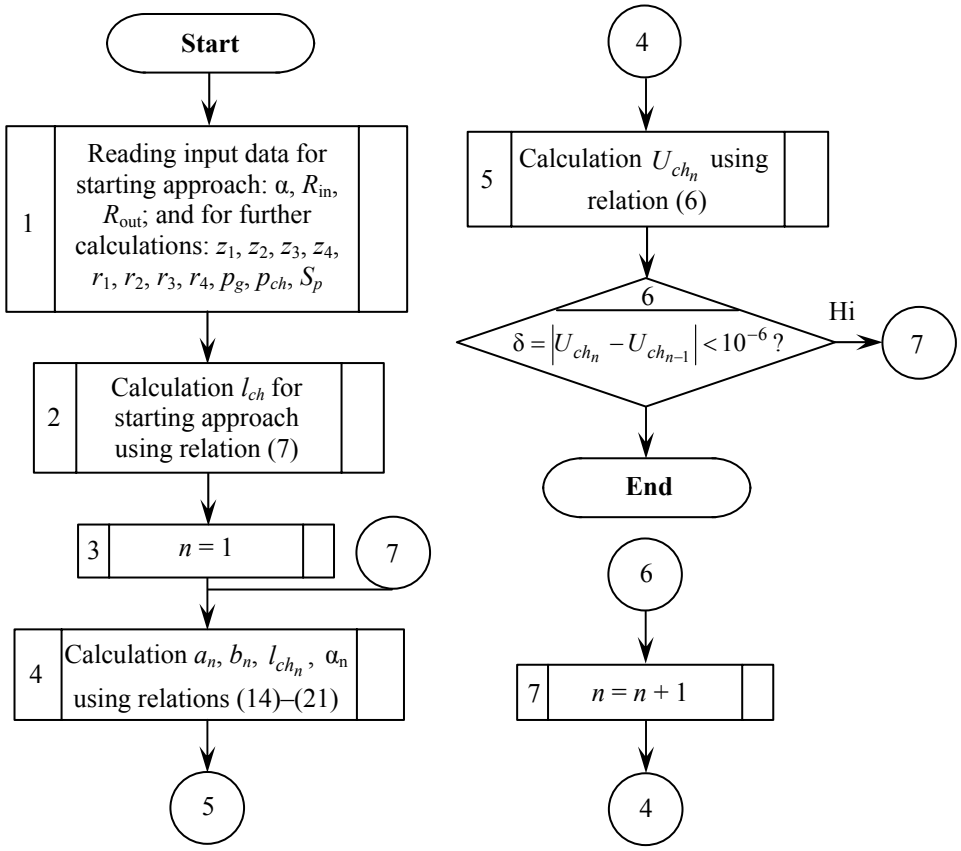


Fig.2. Flow chart of described iterative algorithm for define geometry parameters of nonlinear vacuum channel for transporting electron beam

The testing results of using proposed and described algorithm for solving the task of designing electron-beam vacuum equipment will be presented in the next part of the article. All numerical calculations have been provided using programming manes, as well as numerical and graphic libraries of MATLAB software for scientific ang technical calculations [73]. Among advanced programming means of MATLAB structure and matrix approach have been implemented [65; 73].

OBTAINED RESULTS OF NUMERICAL EXPERIMENTS AND ITS DISCUSSION

Let's considering and analyzing in this part of article the several task of defining the geometry parameters of nonlinear transporting channel. In all tasks, which will be considered below, on the start iteration such geometry parameters of nonlinear tube have been taken: $R_{\text{in}} = 0.006\,\text{m}$, $R_{\text{out}} = 0.015\,\text{m}$, $\alpha = 3.5$. All calculations have been provided for such parameters of electron-beam equipment, corresponding to Fig. 1: $p_g = 5\,\text{Pa}$, $p_{ch} = 0.1\,\text{Pa}$, $S_p = 0.001\frac{\text{m}^3}{\text{s}}$. Starting value for the length of channel has been calculated using relation (7). Therefore, only the coordinates of basic points have been changed in considered tests.

Task 1. $z_1 = 0.01\,\text{m}$, $z_2 = 0.02\,\text{m}$, $z_3 = 0.03\,\text{m}$, $z_4 = 0.04\,\text{m}$; $r_1 = 0.017\,\text{m}$, $r_2 = 0.02\,\text{m}$, $r_3 = 0.021\,\text{m}$, $r_4 = 0.022\,\text{m}$. Obtained solution: $R_{\text{in}} = 0.0429\,\text{m}$, $R_{\text{out}} = 0.0825\,\text{m}$, $l_{ch} = 0.0691\,\text{m}$, $\alpha = 5.0496$. This solution has been obtained after 23 iterations. Obtained graphic dependence $r_{ch}(z)$ for this task is presented at Fig. 3 as straight line.

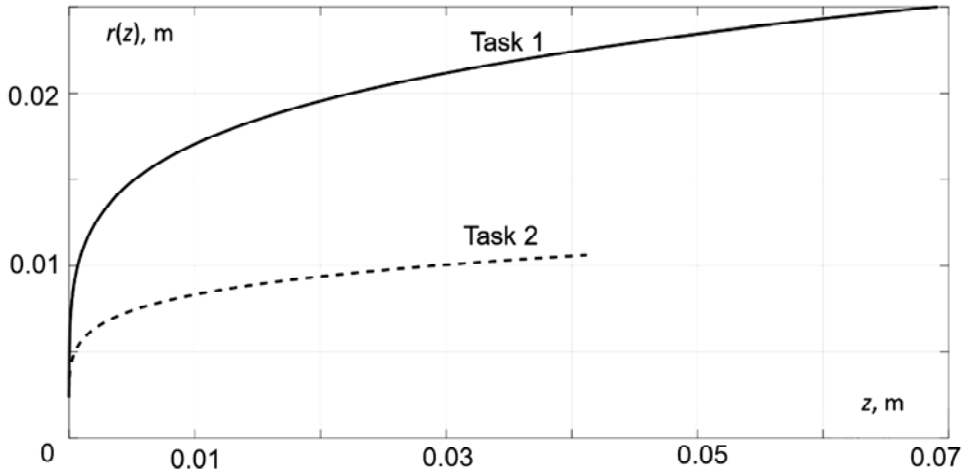


Fig. 3. Graphic dependences $r_{ch}(z)$, obtained using iterative relations (14)–(21) for data sets of Task 1 and Task 2

Task 2. $z_1 = 0.01\,\text{m}$, $z_2 = 0.02\,\text{m}$, $z_3 = 0.03\,\text{m}$, $z_4 = 0.04\,\text{m}$; $r_1 = 0.008\,\text{m}$, $r_2 = 0.009\,\text{m}$, $r_3 = 0.01\,\text{m}$, $r_4 = 0.012\,\text{m}$. Obtained solution: $R_{\text{in}} = 0.0701\,\text{m}$, $R_{\text{out}} = 0.1128\,\text{m}$, $l_{ch} = 0.0412\,\text{m}$, $\alpha = 5.7743$. This solution has been obtained after 28 iterations. Obtained graphic dependence $r_{ch}(z)$ for this task is presented at Fig. 3 as dash line.

Task 3. $z_1 = 0.01\,\text{m}$, $z_2 = 0.02\,\text{m}$, $z_3 = 0.03\,\text{m}$, $z_4 = 0.04\,\text{m}$; $r_1 = 0.0066\,\text{m}$, $r_2 = 0.0085\,\text{m}$, $r_3 = 0.01\,\text{m}$, $r_4 = 0.011\,\text{m}$. Obtained solution: $R_{\text{in}} = 0.0062\,\text{m}$, $R_{\text{out}} = 0.0153\,\text{m}$, $l_{ch} = 0.2234\,\text{m}$, $\alpha = 2.7278$. This solution has been obtained after 13 iterations. Obtained graphic dependence $r_{ch}(z)$ for this task for all range of changing of longitudinal coordinate z is presented at Fig. 4, *a*, and for the start

range of changing z coordinate, correspondently, in Fig. 4, *b*. This solution is reflected in Fig. 4 *a*, *b*, as straight line.

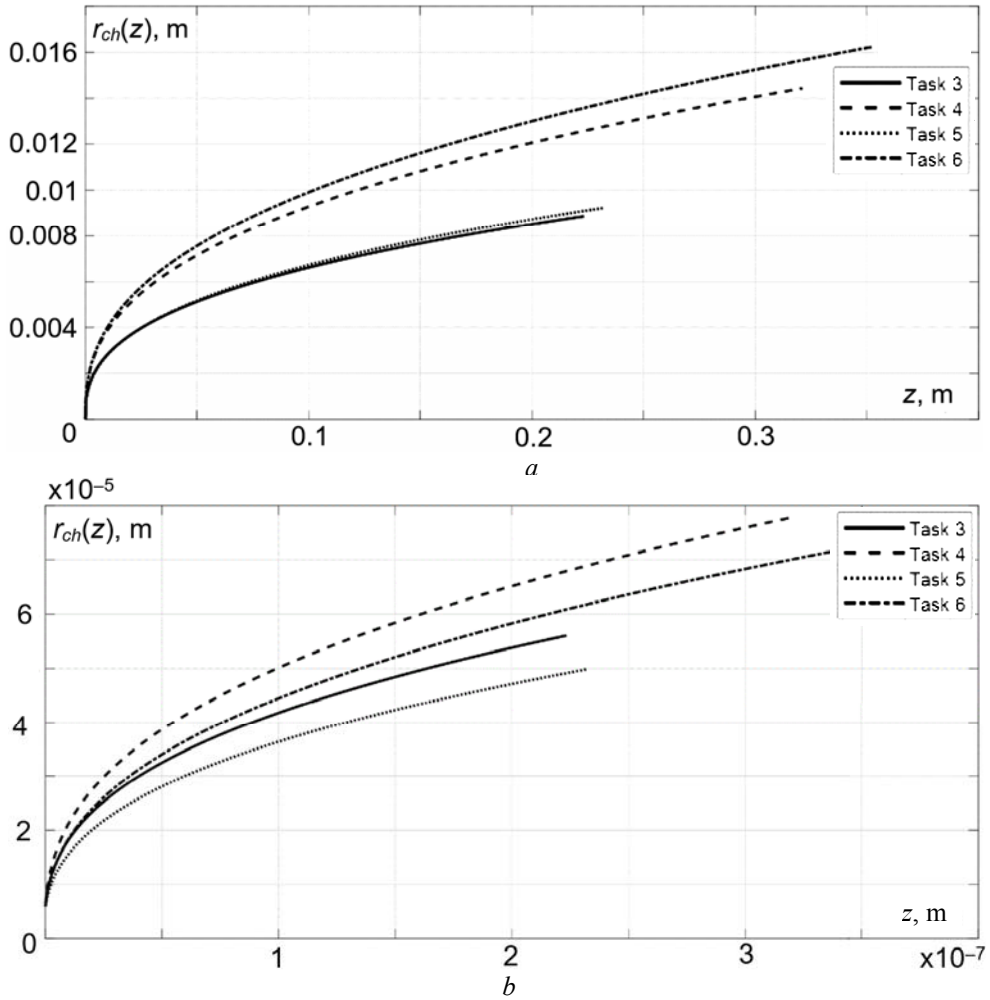


Fig. 4. Graphic dependences $r_{ch}(z)$, obtained using iterative relations (14)–(21) for data sets of Tasks 3, 4, 5 and 6 in the all range (*a*) and in start range (*b*) of changing of longitudinal coordinate z

Task 4. $z_1 = 0.01\,\text{m}$, $z_2 = 0.02\,\text{m}$, $z_3 = 0.03\,\text{m}$, $z_4 = 0.04\,\text{m}$; $r_1 = 0.0093\,\text{m}$, $r_2 = 0.0121\,\text{m}$, $r_3 = 0.0141\,\text{m}$, $r_4 = 0.0157\,\text{m}$. Obtained solution: $R_{\text{in}} = 0.0059\,\text{m}$, $R_{\text{out}} = 0.0151\,\text{m}$, $l_{ch} = 0.32211\,\text{m}$, $\alpha = 2.64456$. This solution has been obtained after 15 iterations. Obtained graphic dependence $r_{ch}(z)$ for this task for all range of changing of longitudinal coordinate z is presented at Fig. 4, *a*, and for the start range of changing z coordinate, correspondently, in Fig. 4, *b*. This solution is reflected in Fig. 4 *a*, *b*, as dash line.

Task 5. $z_1 = 0.01\,\text{m}$, $z_2 = 0.02\,\text{m}$, $z_3 = 0.03\,\text{m}$, $z_4 = 0.04\,\text{m}$; $r_1 = 0.0067\,\text{m}$, $r_2 = 0.0087\,\text{m}$, $r_3 = 0.0102\,\text{m}$, $r_4 = 0.0113\,\text{m}$. Obtained solution: $R_{\text{in}} = 0.0061\,\text{m}$, $R_{\text{out}} = 0.0149\,\text{m}$, $l_{ch} = 0.2323\,\text{m}$, $\alpha = 2.6456$. This solution has been obtained after 17 iterations. Obtained graphic dependence $r_{ch}(z)$ for this task for all range of

changing of longitudinal coordinate z is also presented at Fig. 4, *a*, and for the start range of changing z coordinate, correspondently, in Fig. 4, *b*. This solution is reflected in Fig. 4 *a*, *b*, as dot line.

Task 6. $z_1 = 0.01\,\text{m}$, $z_2 = 0.02\,\text{m}$, $z_3 = 0.03\,\text{m}$, $z_4 = 0.04\,\text{m}$; $r_1 = 0.0099\,\text{m}$, $r_2 = 0.013\,\text{m}$, $r_3 = 0.0152\,\text{m}$, $r_4 = 0.0171\,\text{m}$. Obtained solution: $R_{\text{in}} = 0.0063\,\text{m}$, $R_{\text{out}} = 0.0152\,\text{m}$, $l_{ch} = 0.352\,\text{m}$, $\alpha = 2.5546$. This solution has been obtained after 14 iterations. Obtained graphic dependence $r_{ch}(z)$ for this task for all range of changing of longitudinal coordinate z is also presented at Fig. 4, *a*, and for the start range of changing z coordinate, correspondently, in Fig. 4, *b*. This solution is reflected in Fig. 4 *a*, *b*, as dash-dot line.

From the obtained calculation results and graphical dependencies presented in Fig. 3 and Fig. 4, it is clear that the iterative numerical algorithm based on relations (14)–(21), the flow chart of which is presented in Fig. 2, converges stably for various sets of numerical data for base points $\mathbf{P_1} - \mathbf{P_4}$. It should be noted that this is a very important practical result, since the numerical solution of sets of equations containing power and logarithmic functions with a high degree of rigidity cannot always be implemented using standard numerical methods [62–64].

A good proof of the stable convergence of the proposed iterative method is that for close values of the base point data sets, the solutions to the tasks posed are almost identical. This is clearly visible in the obtained solutions for Task 3 and Task 5. Indeed, as can be seen from the graphical dependencies shown in Fig. 4, *a*, the solutions obtained for given sets of points with close coordinates practically coincide.

It should also be noted that all the considered examples are of a practical nature, and the numerical data sets given in problems 1–6 correspond to the actual dimensions of the channels for transporting electron beams in industrial technological equipment. For example, with an input diaphragm radius of $8\,\text{mm}$ and a radius of a cylindrical beam transportation channel of $16\,\text{mm}$, a pressure drop from $5\,\text{Pa}$ to $0.1\,\text{Pa}$ at a pumping speed of $0.001\,\frac{\text{m}^3}{\text{s}}$ is provided with a transportation channel length of 0.28 m. It is possible in the model task to consider such a transporting channel as a nonlinear one described by the function (4) with parameter $\alpha = 20$. The calculation results for such a model of the beam transport channel give a result of $0.24\,\text{m}$, which is in good agreement with experimental data, taking into account the complexity of the simulation of real vacuum systems [51–57]. To further harmonize theoretical and experimental data, empirical coefficients can be introduced into calculation formulas (14)–(17). In any case, the stable convergence of iterative formulas (18)–(21) is an undeniable advantage of the proposed algorithm for solving important practical problems of modern electron beam technologies.

To carry out further theoretical research in order to solve complex practical and engineering problems of modern industrial electron beam technologies, it is necessary to combine the numerical method described in this article for calculating the geometric parameters of the vacuum channel for transporting an electron beam with the previously proposed modern methods of interpolation and approximation of the boundary trajectories of an electron beam propagating in soft

vacuum [66–72]. In this way, the important practical problem of optimizing beam current losses in the guide channel, described by equations (11)–(13), can be solved.

All research work described in this paper has been provided in the Scientific and Educational Laboratory of Electron Beam Technological Devices of the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnical Institute".

CONCLUSION

The test numerical experiments carried out in the research work showed that the proposed algorithm for the numerical calculation of the geometric parameters of a nonlinear vacuum channel for transporting a short-focus electron beam, specified by iterative relations (14)–(21) and shown in the form of a flow chart in Fig. 2, converges stably for the pumping speed of the vacuum chamber and the range of pressures in the electron gun and in the technological chamber used in electron beam technological installations. The obtained results of numerical calculations are in good agreement with experimental data. The divergence of calculated and experimental data for test tasks did not exceed 10%. The implementation of the proposed algorithm together with the solution of problems of interpolation and approximation of the boundary trajectory of an electron beam in a single software package will allow, at the initial stage of designing electron beam technological equipment, to estimate the energy losses of the electron beam in the transportation channel and to study complexly the issue of the efficiency of applying of nonlinear beam transportation channels in industrial installations, taking into account the complexity of their production.

Currently, for the manufacture of beam transportation channels with nonlinear geometry, electron beam technologies of three-dimensional metal printing can be applied. In such cases, the manufacturing technology of curvilinear tubes is significantly simplified, and the energy costs as well as the consumption of the material used are also valuably reduced. The accuracy of manufacturing a nonlinear channel using the electron beam three-dimensional printing method is also much higher than when using traditional mechanical processing methods, and the mechanical and thermodynamic properties of the resulting metal are always much better [43; 44].

The results presented in the article may be of great practical interest to experts involved in the development of electron beam equipment and its applying in modern industry. Experts in the field of computational mathematics and numerical methods may be interested in the proposed algorithm for the numerical solution of a stiff set of nonlinear equations (10). The distinguishing feature of this algorithm stably converges for different values of the power function exponent $1/\alpha$.

REFERENCES

1. J.D. Lawson, *The Physics of Charged-Particle Beams*. Oxford: Clarendon Press, 1977, 446 p. Available: https://www.semanticscholar.org/paper/The-Physics-of-Charged-Particle-Beams-Stringer/80b5ee5289d5efd8f480b516ec4bade0aa529ea6
2. M. Reiser, *Theory and Design of Charged Particle Beams*. John Wiley & Sons, 2008, 634 p. Available: https://www.wiley.com/en-us/Theory+and+Design+of+Charged+Particle+Beams-p-9783527617630

3. M. Szilagyi, *Electron and Ion Optics*. Springer Science & Business Media, 2012, 539 p. Available: https://www.amazon.com/Electron-Optics-Microdevices-Miklos-Szilagyi/dp/1461282470
4. S.J.R. Humphries, *Charged Particle Beams*. Courier Corporation, 2013, 834 p. Available: https://library.uoh.edu.iq/admin/ebooks/76728-charged-particle-beams---s.-humphries.pdf
5. S. Schiller, U. Heisig, and S. Panzer, *Electron Beam Technology*. John Wiley & Sons Inc, 1995, 508 p. Available: https://books.google.com.ua/books/about/ Electron_Beam_Technology.html?id=QRJTAAAAMAAJ&redir_esc=y
6. R.A. Bakish, *Introduction to Electron Beam Technology*. Wiley, 1962, 452 p. Available: https://books.google.com.ua/books?id=GghTAAAAMAAJ&hl=uk&source=gbs_similarbooks
7. R.C. Davidson, H. Qin, *Physics of Intense Charged Particle Beams in High Energy Accelerators*. World Scientific, Singapore, 2001, 604 p. Available: https://books.google.com.ua/books/about/Physics_Of_Intense_Charged_Particle_Beam.html?id=5M02DwAAQBAJ&redir_esc=y
8. H. Schultz, *Electron Beam Welding*. Woodhead Publishing, 1993, 240 p. Available: https://books.google.com.ua/books?id=I0xMo28DwcIC&hl=uk&source=gbs_book_similarbooks
9. G. Brewer, *Electron-Beam Technology in Microelectronic Fabrication*. Elsevier, 2012, 376 p. Available: https://books.google.com.ua/books?id=snU5sOQD6noC&hl=uk&source=gbs_similarbooks
10. I. Brodie, J.J. Muray, *The Physics of Microfabrication*. Springer Science & Business Media, 2013, 504 p. Available: https://books.google.com.ua/books?id=GQYHCAAAQBAJ&hl=uk&source=gbs_similarbooks
11. S.V. Denbnovetsky, I.V. Melnyk, V.G. Melnyk, B.A. Tugai, and S.B. Tuhai, "High voltage glow discharge electron guns and its advanced application examples in electronic industry," *2016 International Conference Radio Electronics & Info Communications (UkrMiCo), Kyiv, Ukraine, 2016*.
12. I.V. Melnyk, "Numerical simulation of distribution of electric field and particle trajectories in electron sources based on high-voltage glow discharge," *Radioelectronic and Communication Systems*, vol. 48, no. 6, pp. 61–71, 2005. doi: https://doi.org/10.3103/S0735272705060087
13. J.I. Etcheverry, N. Mingolo, J.J. Rocca, and O.E. Martınez, "A Simple Model of a Glow Discharge Electron Beam for Materials Processing," *IEEE Transactions on Plasma Science*, vol. 25, no. 3, pp. 427–432, June, 1997. doi: 10.1109/27.597256
14. A.A. Druzhinin, I.P. Ostrovskii, Y.N. Khoverko, N.S.Liakh-Kaguy, and A.M. Vuytsyk, "Low temperature characteristics of germanium whiskers," *Functional materials 21*, no. 2, pp. 130–136, 2014. Available: http://dspace.nbuv.gov.ua/bitstream/handle/123456789/ 120404/02-Druzhinin.pdf?sequence=1
15. A.A. Druzhinin, I.A. Bolshakova, I.P. Ostrovskii, Y.N. Khoverko, and N.S. Liakh-Kaguy, "Low temperature magnetoresistance of InSb whiskers," *Materials Science in Semiconductor Processing*, vol. 40, pp. 550–555, 2015. Available: https://academic-accelerator.com/search?Journal=Druzhinin
16. I. Melnyk, S. Tuhai, M. Surzhykov, I. Shved, V. Melnyk, and D. Kovalchuk, "Analytical Estimation of the Deep of Seam Penetration for the Electron-Beam Welding Technologies with Application of Glow Discharge Electron Guns," *2022 IEEE 41-st International Conference on Electronics and Nanotechnology (ELNANO), 2022*, pp. 1–5. doi: 10.1109/ELNANO54667_2022_9927071
17. I.V. Melnyk, "Estimating of current rise time of glow discharge in triode electrode system in case of control pulsing," *Radioelectronic and Communication Systems*, vol. 56, no. 12, pp. 51–61, 2017. doi: 10.3103/S0735272713120066
18. S.V. Denbnovetskiy, V.G. Melnyk, I.V. Melnyk, B.A. Tugay, and S.B. Tugay, "Generation of electron beams in high voltage glow discharge in pulse regime. Electronics and nanotechnology," *Proceedings of the XXXII International Scientific Conference*

ELNANO 2012, April, 10–12, 2012, Kyiv, Ukraine, pp. 40–41. Available: https://www.researchgate.net/profile/M-Jafarov-Or-Ma-Dzhafarov-2/publication/ 272208438_Nanonegatron_Phenomenon_in_ZnS1-xSex_Films_Deposited_from_Solution/links/5a95b5d2aca27214056941aa/Nanonegatron-Phenomenon-in-ZnS1-xSex-Films-Deposited-from-Solution.pdf#page=50

19. S.V. Denbnovetskiy, V.G. Melnyk, I.V. Melnyk, B.A. Tugay, and S.B. Tugay, "Investigation of electron-ion optics of pulse technological glow discharge electron guns," *2013 IEEE XXXIII International Scientific Conference Electronics and Nanotechnology. ELNANO-2013*. Available: https://ieeexplore.ieee.org/document/ 6552052
20. I.V. Melnik, "Simulation of geometry of high voltage glow discharge electrodes' systems, formed profile electron beams. Proceedings of SPIE," *Seventh Seminar on Problems of Theoretical and applied Electron and Ion Optics*, vol. 6278, pp. 627809-1–627809-13, 2006, doi: https://doi.org/10.1117/12.693202
21. I.V. Melnyk, A.V. Pochynok, "Modeling of electron sources for high voltage glow discharge forming profiled electron beams," *Radioelectronics and Communications Systems*, vol. 62, issue 6, pp. 251–261, 2019. Available: https://link.springer.com/journal/11976/volumes-and-issues/62-6?page=1
22. I. Melnyk, V. Melnyk, B. Tugai, S. Tuhai, N. Mieshkova, and A. Pochynok, "Simplified Universal Analytical Model for Defining of Plasma Boundary Position in the Glow Discharge Electron Guns for Forming Conic Hollow Electron Beam," *2019 IEEE 39th International Con–ference on Electronics and Nanotechnology (ELNANO). Conference proceedings, April 16-18, 2019, Kyiv, Ukraine*, pp. 548–552. Available: https://ieeexplore.ieee.org/document/8783454
23. T.O. Prikhna et al., "Electron-Beam and Plasma Oxidation-Resistant and Thermal-Barrier Coatings Deposited on Turbine Blades Using Cast and Powder Ni(Co)CrALY(Si) Alloys I. Fundamentals of the Production Technology, Structure, and Phase Composition of Cast NiCrAlY Alloys," *Powder Metallurgy and Metal Ceramics*, vol. 61, issue 1-2, pp. 70–76, 2022. doi: 10.1007/s11106-022-00320-x
24. T.O. Prikhna et al., "Electron-Beam and Plasma Oxidation-Resistant and Thermal-Barrier Coatings Deposited on Turbine Blades Using Cast and Powder Ni(Co)CrAlY(Si) Alloys Produced by Electron-Beam Melting II. Structure and Chemical and Phase Composition of Cast CoCrAlY Alloys," *Powder Metallurgy and Metal Ceramicsthis*, vol. 61, issue 3-4, pp. 230–237, 2022. doi: 10.1007/s11106-023-00333-0
25. I.M. Grechanyuk et al., "Electron-Beam and Plasma Oxidation-Resistant and Thermal-Barrier Coatings Deposited on Turbine Blades Using Cast and Powder Ni(Co)CrAlY(Si) Alloys Produced by Electron Beam Melting IV. Chemical and Phase Composition and Structure of Cocralysi Powder Alloys and Their Use," *Powder Metallurgy and Metal Ceramics*, vol. 61, issue 7-8, pp. 459–464, 2022. doi: 10.1007/s11106-022-00310-z
26. M.I. Grechanyuk et al., "Electron-Beam and Plasma Oxidation-Resistant and Thermal-Barrier Coatings Deposited on Turbine Blades Using Cast and Powder Ni (Co)CrAlY (Si) Alloys Produced by Electron Beam Melting III. Formation, Structure, and Chemical and Phase Composition of Thermal-Barrier Ni(Co)CrAlY/ZrO_2–Y_2O_3 Coatings Produced by Physical Vapor Deposition in One Process Cycle," *Powder Metallurgy and Metal Ceramics*, vol. 61, issue 5-6, pp. 328–336, 2022. doi: 10.1007/s11106-022-00320-x
27. V.G. Grechanyuk et al., "Copper and Molybdenum-Based Nanocrystalline Materials," *Metallophysics and Advanced Technologies*, vol. 44, no.7, pp. 927–942, 2022. doi: https://doi.org/10.15407/mfint.44.07.0927
28. M.I. Grechanyuk et al., "Massive Microporous Composites Condensed from the Vapour Phase," *Nanosistemi, Nanomateriali, Nanotehnologii*, vol. 20, no. 4, pp. 883–894, 2022. Available: https://www.imp.kiev.ua/nanosys/media/pdf/2022/4/nano_vol20_iss4_p0883p0894_2022.pdf
29. M.I. Grechanyuk, V.G. Grechanyuk, A.M. Manulyk, I.M. Grechanyuk, A.V. Kozyrev, and V.I. Gots, "Massive Dispersion-Strengthened Composition Mate-

rials with Metal Matrix Condensed from the Vapour Phase," *Nanosistemi, Nanomateriali, Nanotehnologii*, vol. 20, no. 3, pp. 683–692, 2022. Available: https://www.imp.kiev.ua/nanosys/media/pdf/2022/3/nano_vol20_iss3_p0683p0692_2022.pdf

30. N.I. Grechanyuk, V.P. Konoval, V.G. Grechanyuk, G.A. Bagliuk, and D.V. Myroniuk, "Properties of Cu–Mo Materials Produced by Physical Vapor Deposition for Electrical Contacts," *Powder Metallurgy and Metal Ceramicsthis*, 2021, vol. 60, no. 3–4, pp. 183–190. doi: 10.1007/s11106-021-00226-0
31. N.I. Grechanyuk, V.G. Grechanyuk, "Precipitation-Strengthened and Microlayered Bulk Copper- and Molybdenum-Based Nanocrystalline Materials Produced by High-Speed Electron-Beam Evaporation–Condensation in Vacuum: Structure and Phase Composition," *Powder Metallurgy and Metal Ceramics*, vol. 56, no. 11–12, pp. 633–646, 2018. doi: https://doi.org/10.1007/s11106-018-9938-4
32. N.I. Grechanyuk et al., "Laboratory electron-beam multipurpose installation L-2 for producing alloys, composites, coatings, and powders," *Powder Metallurgy and Metal Ceramics*, vol. 56, no. 1–2, pp. 147–159, 2017.
33. A. Zakharov, S. Rozenko, S. Litvintsev, and M. Ilchenko, "Trisection Bandpass Filter with Mixed Cross-Coupling and Different Paths for Signal Propagation," *IEEE Microwave Wireless Component Letters*, vol. 30, no. 1, pp. 12–15, Jan. 2020. doi: 10.1109/LMWC.2019.2957207
34. A. Zakharov, S. Litvintsev, and M. Ilchenko, "Trisection Bandpass Filters with All Mixed Couplings," *IEEE Microwave Wireless Components Letter*, vol. 29, no. 9, pp. 592–594, 2019. Available: https://ieeexplore.ieee.org/abstract/document/8782802
35. A. Zakharov, S. Rozenko, and M. Ilchenko, "Varactor-tuned microstrip bandpass filter with loop hairpin and combline resonators," *IEEE Transactions on Circuits Systems. II. Experimental Briefs*, vol. 66, no. 6, pp. 953–957, 2019. Available: https://ieeexplore.ieee.org/document/8477112
36. T. Kemmotsu, T. Nagai, and M. Maeda, "Removal Rate of Phosphorous form Melting Silicon," *High Temperature Materials and Processes*, vol. 30, issue 1-2, pp. 17–22, 2011. Available: https://www.degruyter.com/journal/key/htmp/30/1-2/html
37. J.C.S. Pires, A.F.B. Barga, and P.R. May, "The purification of metallurgically grade silicon by electron beam melting," *Journal of Materials Processing Technology*, vol. 169, no. 1, pp. 347–355, 2005. Available: https://www.academia.edu/9442020/The_purification_of_metallurgical_grade_silicon_by_electron_beam_melting
38. D. Luo, N. Liu, Y. Lu, G.Zhang, and T. Li, "Removal of impurities from metallurgically grade silicon by electron beam melting," *Journal of Semiconductors*, vol. 32, issue 3, article ID 033003, 2011. Available: http://www.jos.ac.cn/en/article/doi/10.1088/1674-4926/32/3/033003
39. D. Jiang, Y. Tan, S. Shi, W. Dong, Z. Gu, and R. Zou, "Removal of phosphorous in molten silicon by electron beam candle melting," *Materials Letters*, vol. 78, pp. 4–7, 2012. Available: https://www.sciencedirect.com/science/article/abs/pii/ S0969806X1530133X
40. A. Mitchell, T. Wang, "Electron beam melting technology review," *Proceedings of the Conference "Electron Beam Melting and Refining — State of the Art 2000," Reno, NV, USA, 2000*, ed. R. Bakish, pp. 2–11.
41. D.V. Kovalchuk, N.P. Kondraty, "Electron-beam remelting of titanium – problems and development prospects," *Titan 2009*, no. 1(23), pp. 29–38.
42. J. Zhang et al., "Fine equiaxed β grains and superior tensile property in Ti–6Al–4V alloy deposited by coaxial electron beam wire feeding additive manufacturing," *Acta Metallurgica Sinica (English Letters)*, 33(10), pp. 1311–1320, 2020. doi: 10.1007/s40195-020-01073-5
43. D. Kovalchuk, O. Ivasishin, "Profile electron beam 3D metal printing," in *Additive Manufacturing for the Aerospace Industry*. Elsevier Inc., 2019, pp. 213–233.
44. M. Wang et al., "Microstructure and mechanical properties of Ti-6Al-4V cruciform structure fabricated by coaxial electron beam wire-feed additive manufacturing," *Journal of Alloys and Compounds*, vol. 960. article 170943. doi: https://doi.org/10.1016/j.jallcom.2023.170943

45. B.M. Smirnov, *Theory of Gas Discharge Plasma*. Springer, 2015, 433 p. Available: https://www.amazon.com/Theory-Discharge-Springer-Optical-Physics/dp/3319110640
46. M.A. Lieberman, A.J. Lichtenberg, *Principles of Plasma Discharges for Materials Processing*. New York: Wiley Interscience, 1994, 572 p. Available: https:// people.physics.anu.edu.au/~jnh112/AIIM/c17/Plasma_discharge_fundamentals.pdf
47. Yu.P. Raizer, *Gas Discharge Physics*. New York: Springer, 1991, 449 p. Available: https://d-nb.info/910692815/04
48. S.V. Denbnovetsky, I.V. Melnyk, V.G. Melnyk, B.A. Tugai, and S.B. Tuhai, "Simulation of dependences of discharge current of high voltage glow discharge electron guns from parameters of electromagnetic valve," *2017 IEEE 37th International Conference on Electronics and Nanotechnolog*y. doi: 10.1109/ELNANO.2017.7939781
49. I.V. Melnyk, V.G. Melnyk, B.A. Tugai, and S.B. Tuhai, "Investigation of Complex Control System for High Voltage Glow Discharge Electron Sources," *The Second IEEE International Conference on Information-Communication Technologies and Radioelectronics UkrMiCo'2017. Collections of Proceedings of the Scientific and Technical Conference, 11-15 September, Odesa, Ukraine, 2017*, pp. 295–299. Available: https://ieeexplore.ieee.org/document/8095394
50. I. Melnyk, S. Tuhai, M. Surzhykov, and I. Shved, "Discrete Vehicle Automation Algorithm Based on the Theory of Finite State Machine," in *M. Klymash, A. Luntovskyy, M. Beshley, I. Melnyk, A. Schill, Editors. Emerging Networking in the Digital Transformation Age. Lecture Notes in Electrical Engineering*, vol 965, pp. 231–245. Springer, 2023. doi: https://doi.org/10.1007/978-3-031-24963-1_13
51. G. Lewin, *Fundamentals of Vacuum Science and Technology*. McGraw-Hill, 1965, 248 p. Available: https://www.amazon.com/Fundamentals-Vacuum-Science-Technology-Gerhard/dp/B0000CMJUA
52. J.M. Lafferty, *Foundations of Vacuum Science and Technology*. John Wiley & Sons, 1998, 756 p. Available: http://vacmarket.com/assets/Technical-Library/Foundations-of-vacuum-science-and-technology/Foundations-of-vacuum-science-and-technology-1998.pdf
53. M.H. Hablanian, *High-Vacuum Technology: A Practical Guide*; Second Edition (Mechanical Engineering). Marcel Dekker Inc., 568 p. Available: https://www.amazon.com/High-Vacuum-Technology-Practical-Mechanical-Engineering/dp/0824798341
54. K. Jousten, *Handbook of Vacuum Technology*. John Wiley & Sons, 2016, 1050 p. Available: https://books.google.com.ua/books/about/Handbook_of_Vacuum_echnology. tml?id=IlBgDAAAQBAJ&redir_esc=y
55. D.J. Hata, *Introduction to Vacuum Technology*. Pearson Prentice Hall, 2008, 187 p. Available: https://books.google.com.ua/books?id=PH4bAQAAMAAJ&hl=uk&source=s_similarbooks
56. D.J. Hucknall, *Vacuum Technology and Applications*. Elsevier, 2013, 328 p. Available: https://books.google.com.ua/books?id=6tr8BAAAQBAJ&hl=uk&source=gbs_similarbooks
57. P.K. Naik, *Vacuum: Science, Technology and Applications*. CRC Press, 2018, 260 p. Available: https://books.google.com.ua/books?id=9kpnDwAAQBAJ&hl=uk&source=gbs_similarbooks
58. I.N. Bronshtein, K.A. Semendyayev, G. Musiol, and H. Mühlig, *Handbook of Mathematics*; 5th Edition, Springer, 2007, 1164 p.
59. *Handbook on Mathematical Functions with Formulas, Graphs and Mathematical Tables*; Edited by Abramovich Milton and Stegun Irene: National Bureau of Standards, Applied Mathematic Series, 55, Washington, 1964, 1046 p.
60. G.M. Phillips, *Interpolation and Approximation by Polynomials*. Springer, 2023, 312 p. Available: http://bayanbox.ir/view/2518803974255898294/George-M.-Phillips-Interpolation-and-Approximation-by-Polynomials-Springer-2003.pdf

61. N. Draper, H. Smith, *Applied Regression Analysis*; 3 Edition. Wiley Series, 1998, 706 p. Available: https://www.wiley.com/en-us/Applied+Regression+Analysis,+3rd+Edition-p-9780471170822
62. C. Mohan, K. Deep, *Optimization Techniques*. New Age Science, 2009, 628 p. Available: https://www.amazon.com/Optimization-Techniques-C-Mohan/dp/ 1906574219
63. M.K. Jain, S.R.K. Iengar, and R.K. Jain, *Numerical Methods for Scientific & Engineering Computation*. New Age International Pvt. Ltd., 2010, 733 p. Available: https://www.google.com.ua/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwippcuT7rX8AhUhlYsKHRfBCG0QFnoECEsQAQ&url=https%3A%2F%2Fwww.researchgate.net%2Fprofile%2FAbiodun_Opanuga%2Fpost%2Fhow_can_solve_a_non_linear_PDE_using_numerical_method%2Fattachment%2F59d61f7279197b807797de30%2FAS%253A284742038638596%25401444899200343%2Fdownload%2FNumerical%2BMethods.pdf&usg=AOvVaw0MjNl3K877lVWUWw-FPwmV
64. S.C. Chapra, R.P. Canale, *Numerical Methods for Engineers*; 7th Edition. McGraw Hill, 2014, 992 p. Available: https://www.amazon.com/Numerical-Methods-Engineers-Steven-Chapra/dp/007339792X
65. I. Melnyk, A. Luntovskyy, "Estimation of Energy Efficiency and Quality of Service in Cloud Realizations of Parallel Computing Algorithms for IBN," in Klymash, M., Beshley, M., Luntovskyy, A. (eds) *Future Intent-Based Networking. Lecture Notes in Electrical Engineering*, vol. 831, Springer, Cham, pp. 339–379. doi: https://doi.org/10.1007/978-3-030-92435-5_20
66. Melnyk, S.Tuhai, and A. Pochynok, "Calculation of Focal Paramters of Electron Beam Formed in Soft Vacuum at the Plane which Sloped to Beam Axis," *The Forth IEEE International Conference on Information-Communication Technologies and Radioelectronics UkrMiCo'2019. Collections of Proceedings of the Scientific and Technical Conference, Odesa, Ukraine, September 9–13, 2019*. Available: https://ieeexplore.ieee.org/document/9165328
67. I. Melnyk, S. Tuhai, and A. Pochynok, "Interpolation of the Boundary Trajectories of Electron Beams by the Roots from Polynomic Functions of Corresponded Order," *2020 IEEE 40th International Conference on Electronics and Nanotechnology (ELNANO)*, pp. 28–33. doi: 10.1109/ELNANO50318.2020.9088786
68. I. Melnik, S. Tugay, and A. Pochynok, "Interpolation Functions for Describing the Boundary Trajectories of Electron Beams Propagated in Ionised Gas," *15-th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering* (*TCSET – 2020*), pp. 79–83. doi: 10.1109/TCSET49122.2020.235395
69. I.V. Melnyk, A.V. Pochynok, "Study of a Class of Algebraic Functions for Interpolation of Boundary Trajectories of Short-Focus Electron Beams," *System Researches and Information Technologies*, no. 3, pp. 23–39, 2020. doi: https://doi.org/10.20535/SRIT.2308-8893.2020.3.02
70. I. Melnyk, S. Tuhai, M. Skrypka, A. Pochynok, and D. Kovalchuk, "Approximation of the Boundary Trajectory of a Short-Focus Electron Beam using Third-Order Root-Polynomial Functions and Recurrent Matrixes Approach," *2023 International Conference on Information and Digital Technologies (IDT)*, Zilina, Slovakia, 2023, pp. 133–138, doi: 10.1109/IDT59031.2023.10194399
71. I. Melnyk, A. Pochynok, "Basic algorithm for approximation of the boundary trajectory of short-focus electron beam using the root-polynomial functions of the fourth and fifth order," *System Research and Information Technologies*, no. 3, pp. 127–148, 2023. doi: https://doi.org/10.20535/SRIT.2308-8893.2023.3.10
72. I. Melnyk, S. Tuhai, M. Skrypka, T. Khyzhniak, and A. Pochynok, "A New Approach to Interpolation and Approximation of Boundary Trajectories of Electron Beams for Realizing Cloud Computing Using Root-Polynomial Functions," in *2023 Information and Communication Technologies and Sustainable Development.*

ICT&SD 2022. Lecture Notes in Networks and Systems, vol 809, pp. 395–427. Springer, Cham, 2023. doi: https://doi.org/10.1007/978-3-031-46880-3_24

73. J.H. Mathews, K.D. Fink, *Numerical Methods Using MATLAB*; Third Edition. Amazon, 1998, 720 p. Available: https://www.abebooks.com/book-search/title/ numerical-methods-using-matlab/author/john-mat hews-kurtis-fink/?cm_mmc=ggl-_-COMUS_ ETA_DSA--naa--naa&gclid=CjwKCAiAh9qdBh AOEiwAvxIok6hZ7 XHTvi420 qugGwqNZ20QF4PyaaJai-74Z0EK2c3dbVRqo P17hoCP2wQAvDBwE

Received 23.01.2024

INFORMATION ON THE ARTICLE

Igor V. Melnyk, ORCID: 0000-0003-0220-0615, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Ukraine, e-mail: imelnik@phbme.kpi.ua

Alina V. Pochynok, ORCID: 0000-0001-9531-7593, Research Institute of Electronics and Microsystem Technology of the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Ukraine, e-mail: alina_pochynok@yahoo.com

Mykhailo Yu. Skrypka, ORCID: 0009-0006-7142-5569, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Ukraine, e-mail: scientetik@gmail.com

ЧИСЛОВИЙ АЛГОРИТМ РОЗРАХУНКУ ВАКУУМНОЇ ПРОВІДНОСТІ НЕЛІНІЙНОГО КАНАЛУ ТРАНСПОРТУВАННЯ КОРОТКОФОКУСНОГО ЕЛЕКТРОННОГО ПУЧКА У ТЕХНОЛОГІЧНОМУ ОБЛАДНАННІ / І.В. Мельник, А.В. Починок, М.Ю. Скрипка

Анотація. На основі розв'язування рівнянь вакуумної техніки запропоновано і досліджено ітераційний алгоритм розрахунку вакуумної провідності та геометричних параметрів криволінійного каналу транспортування короткофокусного електронного пучка, для якого залежність радіуса каналу від поздовжньої координати описують степеневою функцією. Запропонований алгоритм заснований на числовому розв'язуванні системи нелінійних рівнянь методом Стеффенсена. Наведені результати тестових розрахунків підтверджують стійку збіжність запропонованого алгоритму для реальних значень тиску та швидкості відкачування у технологічному обладнанні. Такі криволінійні канали транспортування можуть бути використані в електронно-променевому обладнанні на основі гармат високовольтного тліючого розряду, призначеному для зварювання, плавлення металів та для нанесення тонких плівок. Критерієм оптимальності геометрії нелінійного каналу є мінімальні втрати потужності електронного пучка під час його транспортування за умови забезпечення необхідного перепаду тиску між розрядною та технологічною камерами.

Ключові слова: електронний пучок, електронно-променеві технології, транспортування електронного пучка, нелінійний канал транспортування електронного пучка, вакуумна провідність каналу транспортування, електронна гармата високовольтного тліючого розряду, рівняння вакуумної техніки, система нелінійних рівнянь, метод Стеффенсена.

UDC 621.396.96
DOI: 10.20535/SRIT.2308-8893.2025.1.06

METHOD OF POLARIZATION SELECTION OF NAVIGATION OBJECTS IN ADVERSE WEATHER CONDITIONS USING STATISTICAL PROPERTIES OF RADIO SIGNALS

D. KORBAN, O. MELNYK, S. KURDIUK, O. ONISHCHENKO, V. OCHERETNA, O. SHCHERBINA, O. KOTENKO

Abstract. This research article is devoted to studying and applying polarization selection for navigation objects in difficult atmospheric conditions. It provides a novel application of Stokes parameters in radar signal processing for navigation objects, validated by experimental data. The main emphasis is on using the statistical properties of the polarization parameters of partially polarized echo signals. The article discusses in detail the statistical properties of the polarization parameters of partially polarized echo signals, which can be used to improve the accuracy of ship radiolocation systems. The study is based on analyzing experimental data collected in various atmospheric conditions. The results indicate the effectiveness of polarization selection in improving the stability and accuracy of radar navigation systems in various atmospheric conditions. The use of statistical methods allows the navigation system to adapt to changing conditions, ensuring reliability in different scenarios. Polarization selection based on the statistical properties of polarization parameters is a promising method to improve navigation in high atmospheric humidity, fog, and other complex atmospheric conditions. It can be used in the development of modern navigation systems.

Keywords: safety of navigation, atmospheric conditions, statistical properties, partially polarized, echo signals, radar systems, navigation equipment, bridge resources, maritime transport, radiolocation, ship handling and maneuvering.

INTRODUCTION

Over the past decades, high-precision navigation systems have become critical for a wide range of applications, including autonomous vehicles, unmanned aerial vehicles, maritime navigation, and others. However, the effectiveness of these systems can be significantly reduced in difficult atmospheric conditions. The development of new methods that ensure stable and efficient navigation in such conditions is an urgent problem. Thus, the growing need for high-precision radio navigation systems requires the development of new methods to ensure stable and efficient navigation in various atmospheric conditions. In this context, polarization selection of navigation objects seems to be a promising research direction. Polarization selection is becoming an object of active research to solve these

© *Publisher IASA at the Igor Sikorsky Kyiv Polytechnic Institute*, 2025

problems. This method is based on the use of statistical properties of polarization parameters of echo signals of partially polarized waves. Light polarization has the potential to improve signal quality and ensure navigation stability in conditions of limited visibility and atmospheric instability.

The literature review encompasses a range of topics related to radar systems, maritime navigation, and signal processing. The discussed works address the issues of ship radar systems, construction of modern high-precision intelligent control systems for marine vessels [1], navigation support for ship traffic control [2], analysis of prediction methods for determining the parameters of ship equipment [3; 4]. In addition, the issues of modeling of radar signals for objects of complex spatial configuration [5] and scattering of electromagnetic waves by radar objects [6] are investigated.

The paper [7] contributes insights into the selection of radar signals from navigation objects in the presence of atmospheric formations. Other works cover constraints on spatial measurements in phased-array radar due to atmospheric effects [9], radar detection and identification methods for objects with resonant sizes [8], and the influence of propagation medium on maritime direction measurements [10–13], address challenges in measuring the range of low-altitude targets within the tropospheric waveguide over the sea and the measurement of the Doppler frequency of signals reflected from targets beyond the radio horizon over the sea.

In [14], a comparison of ship radars operating in different frequency bands was made. The comprehensive manual on radar, AIS, and target tracking for marine radar users presented in [15]. The work [16] examined the suitability of ARPA for the automatic assessment of AIS targets. In [17] discussed ways to enhance awareness of maritime situations through the integration of shipborne radars, [18] explores radars for maritime domain awareness.

In [19; 20] provided essential information on radar technology and the challenges in the public administration of autonomous shipping, respectively. The article [21] addresses multi-level challenges in the public administration of navigation safety. Sources [22–30], cover various aspects of maritime safety, vessel operational efficiency, vulnerability assessment, information security, environmental efficiency, fuzzy controllers in vessel motion control and energy efficient motion modes contributing significantly to the understanding of modern maritime operations, safety and technology.

The references [31–43] contribute to the understanding and improvement of maritime transportation and navigation process safety. These studies cover aspects such as public safety management, maritime situational awareness, vessel equipment vulnerability assessment, vessel information security risks, maritime transportation security, autonomous vessels, vessel traffic management systems and energy efficient modes of propulsion systems. They provide valuable insights into maritime safety issues and developments, offering potential solutions and improvements for safe navigation [44; 45] in different conditions.

Thus the methodology proposed in this paper provides a comprehensive and data-driven approach to solving problems related to polarization selection. Through a thorough literature review, the approach seamlessly integrates statistical analysis, radar measurements, and practical considerations, offering a sound basis for effective navigation decision making. Synthesizing these elements not only improves the understanding of polarization selection methods, but also makes a valuable contribution to the improvement of shipboard radar systems in various atmospheric conditions.

MATERIALS AND METHODS

Safe navigation is of paramount importance to ensure the integrated safety of maritime operations. Ship handling and maneuvering in challenging atmospheric conditions require accurate and reliable navigation systems. Ship radar, being a cornerstone of maritime navigation technology, relies on advanced polarization techniques to accurately detect and analyze echo signals. The utilization of polarimetric data in radar systems enhances the precision of object identification, providing critical information for safe ship handling and maneuvering. By integrating polarimetric insights into radar technology, maritime operators gain a more comprehensive understanding of their surroundings, enabling proactive measures for collision avoidance and ensuring a higher level of safety in challenging atmospheric conditions.

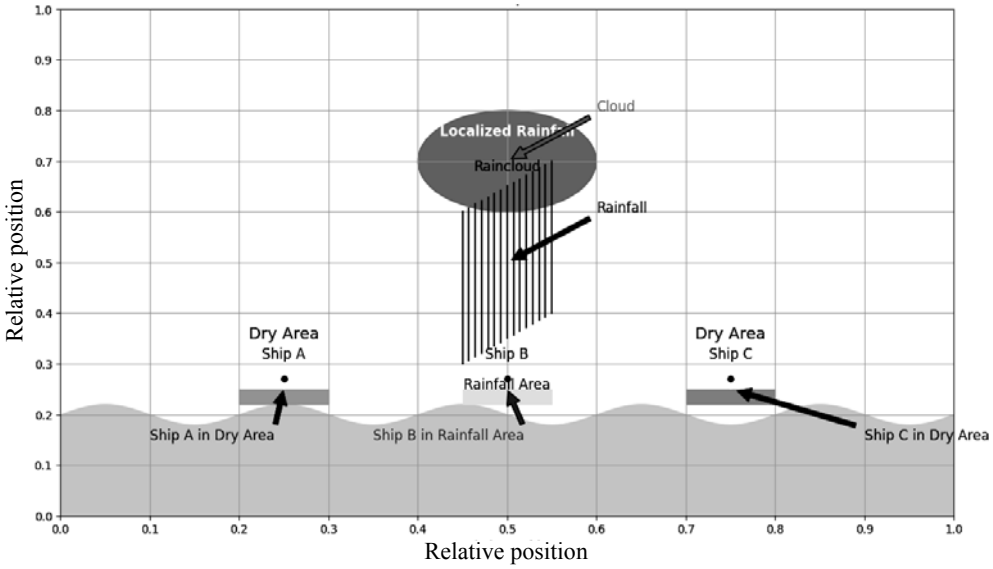
Practical use of polarization parameters of electromagnetic wave in shipboard radar polarization complexes (SRPC) in solving the problem of polarization selection of navigation objects located in difficult conditions atmospheric environment on the way of the ship, due to the need to use microwave elements and antenna devices that allow radiation with subsequent analysis of their polarization parameters. The following antenna devices were used as the SRPC antenna an omnipolarized antenna with controlled polarization to radiation, which allows to optimally realize the energy capabilities of SRPC by representing polarization by real energy Stokes parameters.

During the radar observation of navigation objects against the background of atmospheric formations, the echo signals arriving at the input of the SRPC receiver will be reflections from a complex object. Assuming that the Stokes vector of a complex object has normally distributed components in an arbitrary basis, the one-dimensional law of distribution of the Stokes parameters of the navigation object and the atmospheric formation is used to determine the properties of the echo signal of a complex object received at the SRPC input. The Stokes energy parameters are quadrature with respect to the field strength of a partially polarized electromagnetic wave and uniquely determine its polarization. To fully determine the probability density of the Stokes parameters of the echo signal of a partially polarized electromagnetic wave scattered by a complex object, it is necessary to calculate their mean values, variances, and standard deviations.

To address the challenge of polarization selection for navigation objects amidst atmospheric formations, statistical properties of Stokes parameters of partially polarized electromagnetic waves from the complex object's lunar signals were utilized. Solving the specified task involves the incorporation of a priori information characterizing objects of radar observation systems. This a priori information encompasses the number of observed atmospheric formations (precipitation of varying intensity) that create erroneous markers on the radar indicator. Additionally, it includes a set of features describing the observed objects, as well as the probability distribution laws of the feature set.

The set of features that characterize the recognized objects are the energy polarization parameters of Stokes, which form the predictor, whose components are the Stokes parameters themselves (S_1, S_2, S_3, S_4). The probabilistic characteristics of the predictor $\overline{S}$ are the laws of distribution of the Stokes parameters of the navigation object and the atmospheric formation.

The echo signals of a navigation object are equivalent to the presence of a polarized component in the echo signal of the total partially polarized wave of a complex object. The fluctuating component of the echo signal of the total partially polarized wave of a complex object caused by reflection from an atmospheric formation corresponds in its statistical properties to the echo signal of a partially polarized wave subject to the central limit theorem of probability theory. In the case of a navigation object in the area of an atmospheric formation (Figure), the reflection of an electromagnetic wave irradiating two objects (a complex object) with different electrical conductivity (metal and water in liquid or solid state) at the same time is equivalent to the presence of a polarized component (navigation object) and a fluctuating component (atmospheric formation).



Radar observation of a navigation object against the background of an atmospheric formation

When receiving an echo signal of a partially polarized electromagnetic wave simultaneously from two objects (a complex object), all four Stokes parameters are recorded simultaneously, and their criterion values are set based on the values. At the same time, for a navigation object, polarization selection of its echo signals is performed against the background of echo signals from an atmospheric formation.

To solve the problem of polarization selection of echo signals of a navigation object, the laws of distribution of predictors are used $W(\overline{S}/HO)$ navigation object and $W(\overline{S}/AU)$ of the atmospheric formation, which in the theory of recognition are functions of the probability of the feature vectors of predictors $\overline{S}$, which will include the Stokes parameters of the navigation object and the atmospheric formation. These laws show the probability of forming predictors $\overline{S}$ з with given values of the Stokes parameters, provided that the echo signals are generated by the navigation object and the atmospheric formation, and the distribution of the reflectivity of the predictors $\overline{S}$ can be described by normal laws. Since these predictor $\overline{S}$ distribution laws intersect, the problem of polarization selection of the navigation object is solved using the maximum likelihood rule, which is defined by the following inequality:

$$\frac{W(S_n/HO)}{W(S_n/AO)} \geq 1. \qquad (1)$$

Expression (1) for the four Stokes parameters of the echo signal of a partially polarized wave is written as follows:

$$\frac{W(S_1/HO)}{W(S_1/AO)} \geq 1,\ \frac{W(S_2/HO)}{W(S_2/AO)} \geq 1,\ \frac{W(S_3/HO)}{W(S_3/AO)} \geq 1,\ \frac{W(S_4/HO)}{W(S_4/AO)} \geq 1. \quad (2)$$

Since the reflectivity distribution of the Stokes parameters can be described by normal laws, then inequalities (2) are given in the form:

$$\frac{W(S_1/HO)}{W(S_1/AU)} = \frac{\frac{1}{\sqrt{2\pi}\sigma_{HO}} e^{-\frac{(S_1-m_{HO})^2}{2\sigma_{HO}^2}}}{\frac{1}{\sqrt{2\pi}\sigma_{AU}} e^{-\frac{(S_1-m_{AO})^2}{2\sigma_{AO}^2}}}; \quad \frac{W(S_2/HO)}{W(S_2/AU)} = \frac{\frac{1}{\sqrt{2\pi}\sigma_{HO}} e^{-\frac{(S_2-m_{HO})^2}{2\sigma_{HO}^2}}}{\frac{1}{\sqrt{2\pi}\sigma_{AU}} e^{-\frac{(S_2-m_{AO})^2}{2\sigma_{AO}^2}}};$$

$$\frac{W(S_3/HO)}{W(S_3/AU)} = \frac{\frac{1}{\sqrt{2\pi}\sigma_{HO}} e^{-\frac{(S_3-m_{HO})^2}{2\sigma_{HO}^2}}}{\frac{1}{\sqrt{2\pi}\sigma_{AU}} e^{-\frac{(S_3-m_{AO})^2}{2\sigma_{AO}^2}}};$$

$$\frac{W(S_4/HO)}{W(S_4/AU)} = \frac{\frac{1}{\sqrt{2\pi}\sigma_{HO}} e^{-\frac{(S_4-m_{HO})^2}{2\sigma_{HO}^2}}}{\frac{1}{\sqrt{2\pi}\sigma_{AU}} e^{-\frac{(S_4-m_{AO})^2}{2\sigma_{AO}^2}}}, \quad (3)$$

where S_1, S_2, S_3, S_4 — measured Stokes parameters of a complex SRPC observation object; σ_{HO}, σ_{AU} — statistical parameter — the standard deviation for the general population of a series of observations of the Stokes parameters of the navigation object and the atmospheric formation, respectively; m_{HO}, m_{AU} — statistical parameter — mathematical expectations of the Stokes parameters of the echo signals of the navigation object and the atmospheric formation, respectively; $\sigma_{HO}^2, \sigma_{AU}^2$ — statistical parameter — the variances of the Stokes parameters of the echo signals of the navigation object and the atmospheric formation, respectively.

Transforming the right-hand side of equations (3) according to the expression for the exponential function $e^{a+b} = e^a - e^b = \frac{e^a}{e^b}$ allows us to obtain the following dependencies for the right-hand side of the distribution laws of the predictor of the navigation object and the atmospheric formation.

To provide a comprehensive understanding of the underlying physics, we include the mathematical model that relates the Stokes parameters of the electromagnetic wave to the radar object's properties. The Stokes parameters (S_1, S_2, S_3, S_4) describe the polarization state of an electromagnetic wave and are

derived from the electric field components. The reflected signals can be represented as the result of the interaction of the electromagnetic wave with the target, possessing certain radar characteristics. The Stokes parameters can be expressed through the target characteristics as follows:

Total power of the wave S_1 related to the reflection coefficient R of the target:

$$S_1 = E_i^2 R ,$$

where E_i^2 is the amplitude of the incident wave.

Horizontal and vertical polarizations are related to the scattering components R_h and R_v:

$$S_2 = E_i^2 (R_h - R_v) .$$

Polarization at 45° and –45°:

$$S_2 = E_i^2 (R_{45} - R_{-45}) ,$$

where R_{45} and R_{-45R} are the scattering components at 45° and –45° respectively

Right-hand and left-hand circular polarizations:

$$S_3 = E_i^2 (R_{rcp} - R_{lcp}) ,$$

where R_{rcp} and R_{lcp} are the scattering components of right-hand and left-hand circular polarization respectively.

Considering the statistical properties of the Stokes parameters, the maximum likelihood method can be applied to determine the target parameters. The probability of the measured values of the Stokes parameters S_i for a target and atmospheric conditions can be written as:

$$P(S_i \theta) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(S_i - \mu)^2}{2\sigma^2} \right),$$

where θ represents the target parameters, and μ and σ are the mean and standard deviation of the Stokes parameters respectively.

The Stokes parameters (S_1, S_2, S_3, S_4) are derived from the received radar signals by first decomposing the signals into their horizontal (E_H) and vertical (E_V) components using polarimetric radar techniques. The Stokes parameters are then calculated using the following relations of magnitudes of the horizontal and vertical components of the electric field, respectively and the complex conjugate product of these components, which captures the cross-polarization information.

The derived Stokes parameters are then used in the statistical decision-making process to determine the presence of a radar target. This involves applying the maximum likelihood rule and evaluating the probability densities of the Stokes parameters for both the navigation object and the atmospheric formations. By following these steps, the radar system can effectively differentiate between the echo signals of navigation objects and atmospheric formations, enhancing the accuracy and reliability of radar-based navigation in challenging conditions.

These parameters are subsequently used in the statistical decision-making process, applying the maximum likelihood rule to evaluate the probability densities and determine the presence of a radar target amidst atmospheric formations.

Let us transform the right-hand side of the equation for the first Stokes parameter and obtain:

$$\frac{\frac{1}{\sqrt{2\pi}\sigma_{HO}}e^{-\frac{(S_1-m_{HO})^2}{2\sigma_{HO}^2}}}{\frac{1}{\sqrt{2\pi}\sigma_{AU}}e^{-\frac{(S_1-m_{AU})^2}{2\sigma_{AU}^2}}}=\frac{\sigma_{AU}e^{-\frac{(S_1-m_{HO})^2}{2\sigma_{HO}^2}}}{\sigma_{HO}e^{-\frac{(S_1-m_{AU})^2}{2\sigma_{AU}^2}}}=$$

$$=\frac{\sigma_{AU}}{\sigma_{HO}}e^{\frac{(\sigma_{HO}^2-\sigma_{AU}^2)}{2\sigma_{HO}^2\sigma_{AU}^2}S_1^2+\frac{(\sigma_{AU}^2m_{HO}-\sigma_{HO}^2m_{AU})}{\sigma_{HO}^2\sigma_{AU}^2}S_1+\frac{(\sigma_{HO}^2m_{AU}^2-\sigma_{AU}^2m_{HO}^2)}{2\sigma_{HO}^2\sigma_{AU}^2}}\geq 1; \qquad (4)$$

Similar results are obtained for the second, third, and fourth Stokes parameters:

$$\frac{\frac{1}{\sqrt{2\pi}\sigma_{HO}}e^{-\frac{(S_2-m_{HO})^2}{2\sigma_{HO}^2}}}{\frac{1}{\sqrt{2\pi}\sigma_{AU}}e^{-\frac{(S_2-m_{AU})^2}{2\sigma_{AU}^2}}}=\frac{\sigma_{AU}e^{-\frac{(S_2-m_{HO})^2}{2\sigma_{HO}^2}}}{\sigma_{HO}e^{-\frac{(S_2-m_{AU})^2}{2\sigma_{AU}^2}}}=$$

$$=\frac{\sigma_{AU}}{\sigma_{HO}}e^{\frac{(\sigma_{HO}^2-\sigma_{AU}^2)}{2\sigma_{HO}^2\sigma_{AU}^2}S_1^2+\frac{(\sigma_{AU}^2m_{HO}-\sigma_{HO}^2m_{AU})}{\sigma_{HO}^2\sigma_{AU}^2}S_1+\frac{(\sigma_{HO}^2m_{AU}^2-\sigma_{AU}^2m_{HO}^2)}{2\sigma_{HO}^2\sigma_{AU}^2}}\geq 1; \qquad (5)$$

$$\frac{\frac{1}{\sqrt{2\pi}\sigma_{HO}}e^{-\frac{(S_3-m_{HO})^2}{2\sigma_{HO}^2}}}{\frac{1}{\sqrt{2\pi}\sigma_{AU}}e^{-\frac{(S_3-m_{AU})^2}{2\sigma_{AU}^2}}}=\frac{\sigma_{AV}e^{-\frac{(S_3-m_{HO})^2}{2\sigma_{HO}^2}}}{\sigma_{HO}e^{-\frac{(S_3-m_{AU})^2}{2\sigma_{AU}^2}}}=$$

$$=\frac{\sigma_{AU}}{\sigma_{HO}}e^{\frac{(\sigma_{HO}^2-\sigma_{AU}^2)}{2\sigma_{HO}^2\sigma_{AU}^2}S_1^2+\frac{(\sigma_{AU}^2m_{HO}-\sigma_{HO}^2m_{AU})}{\sigma_{HO}^2\sigma_{AU}^2}S_1+\frac{(\sigma_{HO}^2m_{AU}^2-\sigma_{AU}^2m_{HO}^2)}{2\sigma_{HO}^2\sigma_{AU}^2}}\geq 1; \qquad (6)$$

$$\frac{\frac{1}{\sqrt{2\pi}\sigma_{HO}}e^{-\frac{(S_4-m_{HO})^2}{2\sigma_{HO}^2}}}{\frac{1}{\sqrt{2\pi}\sigma_{AU}}e^{-\frac{(S_4-m_{AU})^2}{2\sigma_{AU}^2}}}=\frac{\sigma_{AU}e^{-\frac{(S_4-m_{HO})^2}{2\sigma_{HO}^2}}}{\sigma_{HO}e^{-\frac{(S_4-m_{AU})^2}{2\sigma_{AU}^2}}}=$$

$$= \frac{\sigma_{AU}}{\sigma_{HO}} e^{\frac{(\sigma^2_{HO}-\sigma^2_{AU})}{2\sigma^2_{HO}\sigma^2_{AU}}S_1^2 + \frac{(\sigma^2_{AU}m_{HO}-\sigma^2_{HO}m_{AU})}{\sigma^2_{HO}\sigma^2_{AU}}S_1 + \frac{(\sigma^2_{HO}m^2_{AU}-\sigma^2_{AU}m^2_{HO})}{2\sigma^2_{HO}\sigma^2_{AU}}} \geq 1. \quad (7)$$

Let us denote the statistical parameters of degree e in equations (4)–(7) by the coefficients l_1, l_2, l_3 i.e:

$$l_1 = \frac{(\sigma^2_{HO}-\sigma^2_{AU})}{2\sigma^2_{HO}\sigma^2_{AU}}; \quad l_2 = \frac{(\sigma^2_{AU}m_{HO}-\sigma^2_{HO}m_{AU})}{\sigma^2_{HO}\sigma^2_{AU}}; \quad l_3 = \frac{(\sigma^2_{HO}m^2_{AU}-\sigma^2_{AU}m^2_{HO})}{2\sigma^2_{HO}\sigma^2_{AU}}, \quad (8)$$

then inequalities (4)–(7) will be written in terms of the coefficients l_1, l_2, l_3 as follows:

$$\frac{\sigma_{AU}}{\sigma_{HO}} e^{l_1S_1^2 + l_2S_1 + l_3} \geq 1; \quad (9)$$

$$\frac{\sigma_{AU}}{\sigma_{HO}} e^{l_1S_2^2 + l_2S_2 + l_3} \geq 1; \quad (10)$$

$$\frac{\sigma_{AU}}{\sigma_{HO}} e^{l_1S_3^2 + l_2S_3 + l_3} \geq 1; \quad (11)$$

$$\frac{\sigma_{AU}}{\sigma_{HO}} e^{l_1S_4^2 + l_2S_4 + l_3} \geq 1. \quad (12)$$

After logarithmizing on the basis of *e*, inequalities (9)–(12) are written as follows:

$$l_1S_1^2 + l_2S_1 + l_3 \geq ln\frac{\sigma_{HO}}{\sigma_{AU}}; \quad (13)$$

$$l_1S_2^2 + l_2S_2 + l_3 \geq ln\frac{\sigma_{HO}}{\sigma_{AU}}; \quad (14)$$

$$l_1S_1^2 + l_2S_1 + l_3 \geq ln\frac{\sigma_{HO}}{\sigma_{AU}}; \quad (15)$$

$$l_1S_4^2 + l_2S_4 + l_3 \geq ln\frac{\sigma_{HO}}{\sigma_{AU}}. \quad (16)$$

Solution of the obtained inequalities (13)–(16) with respect to the Stokes parameters S_1, S_2, S_3, S_4 of echo signals of a partially polarized wave of a complex object allows to obtain their criterion values $S_{1kr}, S_{2kr}, S_{3kr}, S_{4kr}$. Then, for all the Stokes parameters measured by SRPC $S_{1msr}, S_{2msr}, S_{3msr}, S_{4msr}$ their values will be greater than or equal to their criterion values, i.e.:

$$S_{1msr} \geq S_{1kr},\ S_{2msr} \geq S_{2kr},\ S_{3msr} \geq S_{3kr},\ S_{4msr} \geq S_{4kr}$$

and inequalities (13)–(16) will be valid. The operator of the SRPC shall decide on the presence on the indicator or display of the ship's computer of the echo signal only of the navigation object located at a given distance from the SRPC.

If the conditions of the above inequalities are not met, a decision is made on the presence of an atmospheric formation on the indicator of the SRPC or the display of the ship's computer, i.e.

$$S_{1msr} < S_{1kr},\ S_{2msr} < S_{2kr},\ S_{3msr} < S_{3kr},\ S_{4msr} < S_{4kr}.$$

The basis for the use of the basic concepts of mathematical statistics and probability theory is the randomness of echo signals of Stokes parameters in the process of radar observation of navigation objects located in the zone of atmospheric formations by SRPC. The use of the maximum likelihood algorithm in solving the problem of polarization selection of navigation objects makes it possible to assign an echo signal to the object for which the likelihood function is greater.

The polarization selection of navigation objects using the maximum likelihood rule uses a smaller amount of a priori radar information and therefore the application of this rule is practically justified. When studying the values of the Stokes parameters of echo-signals for a navigation object and an atmospheric formation, their probability density distributions are close to normal, and the set of random values of the Stokes parameters of echo-signals of a navigation object and an atmospheric formation is considered as a random process of a complex object.

The values of the Stokes parameters of the navigation object and the atmospheric formation at discrete points in space and time were obtained for a certain period of time based on the results of radar measurements of the SRPC in the Zaporizhzhia region (Ukraine), taking into account the intensity of random precipitation and taking into account the radar information obtained in parallel with the MRL-5 meteorological radar. According to the obtained radar measurements of the SRPC, the echo signals of the first Stokes parameter of the navigation object and the atmospheric formation for a certain intensity of precipitation (in this case $I = 12$ mm/hour), statistical processing of the results of measuring the first Stokes parameter S_{1HO} and S_{1AU}, which are presented in Table 1 and Table 2, respectively.

Table 1. Results of radar observations of the SRPC echo-signal of a navigation object located in the zone of atmospheric formation and parameters of their statistical processing

Grades S_{1HO}	f	$P = f / N$	S_c	d	fd	d^2	fd^2
0.00 – 0.49	4	0.040	0.245	- 0.40	- 1.60	0.16	0.64
0.50 – 0.99	9	0.091	0.745	- 0.30	- 2.70	0.09	0.81
1.00 – 1.49	10	0.101	1.245	- 0.20	- 2.00	0.04	0.40
1.50 – 1.99	15	0.152	1.745	- 0.10	- 1.50	0.01	0.15
2.00 – 2.49	21	0.212	2.245	0.00	0.00	0.00	0.00
2.50 – 2.99	13	0.131	2.745	0.10	1.30	0.01	0.13
3.00 – 4.49	10	0.101	3.245	0.20	2.00	0.04	0.40
3.50 – 3.99	9	0.091	3.745	0.30	2.70	0.09	0.81
4.00 – 4.49	5	0.051	4.245	0.40	2.00	0.16	0.80
4.50 – 4.99	3	0.030	4.745	0.50	1.50	0.25	0.75
$\sum f=N$	**99**	**1.000**			**$\sum$ 1.70**		**$\sum$ 4.90**

Table 2. Results of radar observations of the SRPC echo-signal of atmospheric formation and parameters of their statistical processing

Grades S_{1AU}	f	$P = f / N$	$\overline{S}_1$	d	fd	d^2	fd^2
0.00 – 0.49	3	0.033	0.245	- 0.30	- 0.90	0.09	0.27
0.50 – 0.99	2	0.022	0.745	- 0.20	- 0.40	0.04	0.08
1.00 – 1.49	19	0.209	1.245	- 0.10	- 1.90	0.01	0.19
1.50 – 1.99	35	0.385	1.745	0.00	0.00	0.00	0.00
2.00 – 2.49	20	0.220	2.245	0.10	2.00	0.01	0.20
2.50 – 2.99	6	0.066	2.745	0.20	1.20	0.04	0.24
3.00 – 4.49	4	0.044	3.245	0.30	1.20	0.09	0.36
3.50 – 3.99	2	0.023	3.745	0.40	0.80	0.16	0.32
$\sum f=N$	**91**	**1.000**			$\sum$ **2.00**		$\sum$ **1.66**

Statistical processing of the results of radar observations of the navigation object and the atmospheric formation was carried out using statistical methods. In Tables 1 and 2, for each gradation of Stokes parameters S_{1HO} и S_{1AU} indicates the frequency f (number of cases). Sum of frequencies $\sum f$ of all grades of Stokes parameters is equal to the total number of observations N. Relationship $P = f / N$ is the probability of occurrence of a given gradation. The average value S_1 for each gradation is calculated using the following formula:

$$\overline{S}_1 = A + i\frac{\sum fd}{N}, \tag{17}$$

where A — the middle of the gradation with the largest amount of data S_1, that is, for a navigation object $A = 2.245$, and for atmospheric formation $A = 1.745$; i — the size of the gradation (for our case $i = 5$); $d = \dfrac{S_c - A}{i}$ — a positive or negative number indicating the number of the gradation.

The mathematical expectation for a navigation object and an atmospheric formation is determined using the following relationship:

$$\overline{S}_{1HO(AU)} = m_{S_{iHH(AU)}} = A + i\frac{\sum fd}{N}. \tag{18}$$

The mean square deviation determines the degree of variability of the random first Stokes parameter for the navigation object and the atmospheric formation, which is determined by the formula:

$$\sigma = i\sqrt{\frac{\sum fd^2}{N} - \left(\frac{\sum fd}{N}\right)^2}. \tag{19}$$

The variance, as the square of the standard deviation, is determined by the following formula:

$$\sigma^2 = \sigma^2\frac{N}{N-1}. \tag{20}$$

Based on the results of Tables 1 and 2 and the above formulas, we calculated the relevant statistical characteristics, i.e.:

$$m_{S_{1HO}} = 3.1; \quad \sigma_{S_{1HO}} = 1.1; \quad \sigma^2_{S_{1HO}} = 1.21.$$

$$m_{S_{1AU}} = 1.85; \quad \sigma_{S_{1AU}} = 0.66; \quad \sigma^2_{S_{1AU}} = 0.44.$$

Using the statistical parameters of the echo signals of the navigation object and the atmospheric formation, we calculated the coefficients l_1, l_2, l_3, which have the following values: $l_1 = 0.72$, $l_2 = -0.88$, $l_3 = -0.44$. After substituting them into inequality (9), we obtain:

$$0.72S_1^2 - 0.88S_1 - 0.55 > 0. \tag{21}$$

Solving inequality (21), we obtain two criterion values of the first Stokes parameter $S_{1kr(1)} > 1.68$ and $S_{1kr(2)} < -0.46$. Criterion value of the first Stokes parameter $S_{1kr(2)} < -0.46$ is rejected for physical reasons, since the probability of its occurrence for radar detection of the SRPC navigation object is zero, according to radar observations of a complex object. Therefore, the value of the first Stokes parameter $S_{1kr(1)} = 1.68$ will satisfy the solution of the problem of polarization selection of a navigation object located in the zone of atmospheric formation (precipitation of a certain intensity) and, when radar measurements of the echo signals of the first Stokes parameter are performed, will meet the following conditions $S_{1kr(1)} \geq 1.68$. That is, the task of polarization selection of the navigation object is solved. which is located in the zone of atmospheric formation (precipitation intensity of 12 mm/h). In this case, only the echo signal of the navigation object will be present on the SRPC indicator or on the computer display.

The polarization selection of navigation objects using the maximum likelihood rule (1) uses a smaller amount of a priori radar information and therefore the application of this rule is practically justified. To verify the fulfillment of condition (1), we solve equation (3) using the obtained statistical parameters.

For a navigation object: $m_{S_{1HO}} = 3.1$; $\sigma_{S_{1HO}} = 1.1$; $\sigma^2_{S_{1HO}} = 1.214$; $S_{1HO} = 2.2$.

$$W(S_1/HO) = \frac{1}{\sqrt{2\pi}\sigma_{HO}} e^{-\frac{(S_1 - m_{HO})^2}{2\sigma^2_{HO}}} = \frac{1}{2.5 \cdot 1.1} e^{-\frac{(2.2-3.1)^2}{2 \cdot 1.2}} = 0.36 \cdot 2.6 = 0.95.$$

For atmospheric formation: $m_{S_{1AU}} = 1.85$; $\sigma_{S_{1AU}} = 0.66$; $\sigma^2_{S_{1AU}} = 0.44$; $S_{1AU} = 1.75$.

$$W(S_1/AU) = \frac{1}{\sqrt{2\pi}\sigma_{AU}} e^{-\frac{(S_1 - m_{AU})^2}{2\sigma^2_{AU}}} = \frac{1}{2.5 \cdot 0.66} e^{-\frac{(1.75-1.85)^2}{2 \cdot 0.44}} = 0.61 \cdot 1.1 = 0.67.$$

$$\frac{W(S_1/HO)}{W(S_1/AU)} = \frac{0.95}{0.67} = 1.4.$$

As a result of solving equation (3) using the first Stokes parameter as a predictor, the maximum likelihood rule (1) is fulfilled.

Thus the method proposed for polarized selection of navigation objects in challenging atmospheric conditions through shipboard radar polarization complexes (SRPC) involves a sophisticated approach. Leveraging the statistical properties of Stokes parameters from partially polarized electromagnetic waves derived from lunar signals of complex objects, the technique integrates a priori information to enhance radar observation system capabilities.

A pivotal aspect is the utilization of an omnipolarized antenna with controlled polarization, emphasizing the optimization of SRPC's energy capabilities by representing polarization through real energy Stokes parameters. The analysis of echo signals involves the simultaneous recording of all four Stokes parameters, enabling the polarization selection of navigation objects amidst atmospheric formations.

The application of the maximum likelihood rule is highlighted as a key component, demonstrating its practical justification due to its efficiency, particularly when a limited amount of a priori radar information is available. The derivation of criterion values for Stokes parameters is integral to the process, contributing to informed decision-making.

Radar measurements, considering precipitation intensity and utilizing data from a meteorological radar, provide a real-world context for the study. The statistical processing of radar observations for navigation objects and atmospheric formations involves the calculation of coefficients and verification of statistical parameters.

Accurate identification and tracking of navigation objects even in challenging atmospheric conditions provides ship operators with critical information to make informed decisions during maneuvers. This not only helps avoid collisions, but also enhances the safety of maritime activities. The integration of advanced polarimetric techniques is in line with the broader goal of improving navigation safety in a variety of environmental conditions.

CONCLUSIONS

The presented study outlines a systematic and data-driven methodology for polarimetric extraction of navigational objects in complex atmospheric conditions using shipboard radar polarization systems (SRPC). Using statistical properties of the Stokes parameters of partially polarized electromagnetic waves derived from lunar signals of complex objects, this approach integrates a priori information to improve the performance of radar surveillance systems.

The use of an omnipolarized antenna with controlled polarization is a critical element that emphasizes the optimization of SRPC energy capabilities by representing polarization through real Stokes energy parameters. The analysis of the echo signals includes simultaneous registration of all four Stokes parameters, enabling polarization selection of navigation objects in complex atmospheric formations.

The key point is the application of the maximum likelihood rule, which demonstrates its practical justification due to its effectiveness, especially in the conditions of limited amount of a priori radar information. The derivation of criterion values for the Stokes parameters is an integral part of the process, allowing informed decisions to be made in the context of polarization selection. The meth-

odology offers a comprehensive and rigorous approach to polarization selection problems, demonstrating the integration of statistical analysis, radar measurements, and practical considerations. The results contribute to the understanding of polarimetric selection methods in navigation systems and may find applications in improving the performance of ship radar systems in various atmospheric conditions.

REFERENCES

1. A.P. Ben, I.V. Palamarchuk, "Features of the construction of modern high-precision intelligent control systems for the movement of sea vessels," *Scientific Bulletin of the Kherson State Maritime Academy*, 1(14), pp. 4–10, 2016.
2. V.I. Bogomya, V.S. Davidov, V.V. Doronin, D.P. Pashkov, and I.V. Tikhonov, *Navigation support for vessel traffic management*. Kyiv: DVVP "Kompas", 2012.
3. A.A. Musorin, Y.E. Shapran, and I.V. Trofimenko, "Analysis of forecasting methods for determining the technical parameters of ship equipment," *Proceedings of Azerbaijan State Marine Academy*, 2, pp. 115–119, 2017.
4. O.O. Musorin, Y.E. Shapran, and I.V. Trofimenko, "Features of analytical support for vessel operation in modern conditions," *Scientific Notes of the Ukrainian Research Institute of Communications*, 1(45), pp. 117–121, 2017.
5. V.V. Mal'tsev, I.V. Sisigin, and K.O. Kolesnikov, "Approach to modeling radar signals reflected from objects of complex spatial configuration," *Radiopromyshlennost*, 1, pp. 42–49, 2018.
6. S.V. Nechitaylo, V.M. Orlenko, O.I. Sukharevsky, and V.A. Vasilets, *Electromagnetic Wave Scattering by Aerial and Ground Radar Objects*. Boca Raton, USA: SRC Press Taylor & Francis Group, 2014, 334 p.
7. D.V. Korban, "Selection of radar signals from navigation objects located in the zone of atmospheric formations," in *Proceedings of the Scientific and Technical Conference "Marine and River Fleet: Operation and Repair," March 24–25, 2022, Odessa: NU "OMA",* pp. 25–28.
8. G.S. Zalevsky, A.V. Muzychenko, and O.I. Sukharevsky, "Method of Radar Detection and Identification of Metal and Dielectric Objects with Resonant Sizes Located in Dielectric Medium," *Radioelectronics and Communications Systems*, vol. 55, no. 9, pp. 393–404, 2012.
9. O.L. Kuznetsov, O.B. Tantsyura, and O.L. Melnyk, "Constraints on the quality of spatial measurements in phased-array radar due to the influence of atmospheric inhomogeneities and the Earth's surface," *Systems of Navigation and Communication*, 1 (21), vol. 2, pp. 49–52, 2012.
10. V.D. Karlov, N.N. Petrushenko, V.V. Chelpanov, and K.P. Kvitkin, "The influence of the propagation medium on the maritime direction when measuring the angular coordinates of radar targets," *Collection of scientific works of the Kharkiv University of the Air Force*, 3 (25), pp. 51–53, 2010.
11. V.D. Karlov, D.B. Kucher, O.V. Strutsynsky, and O.V. Lukashuk, "On the issue of measuring the range of a low-altitude target during its radar tracking within the tropospheric waveguide over the sea," *Science and Technology of the Air Force of the Armed Forces of Ukraine*, 3 (24), pp. 98–101, 2016.
12. V.D. Karlov, A.P. Kondratenko, A.K. Sheigas, and Y.B. Sitnik, "On the issue of measuring the Doppler frequency of a signal reflected from a target located beyond the radio horizon over the sea," *Science and Technology of the Air Force of the Armed Forces of Ukraine*, 1(14), pp. 115–117, 2014.
13. V.D. Karlov, A.O. Rodyukov, and I.M. Pichugin, "Statistical characteristics of radar signals reflected from local objects under conditions of anomalous refraction," *Science and Technology of the Air Force of the Armed Forces of Ukraine*, 4(21), pp. 71–74, 2015.

14. A.P. Gorobtsov, A.N. Marinich, and Y.M. Ustinov, "Comparison of ship radars operating in S-, X-, K-bands," *Bulletin of the State University of the Maritime and River Fleet named after Admiral S.O. Makarov*, 5(51), pp. 1087–1093, 2018. doi: https://doi.org/10.21821/2309-5180-2018-10-5-1087-1093
15. A.G. Bole, A.D. Wall, and A. Norris, *Radar and ARPA Manual: Radar, AIS and Target Tracking for Marine Radar Users (3rd edition).* Oxford, United Kingdom: Elsevier Science & Technology, 2014.
16. F. Heymann, T. Noack, P. Banyś, and E. Engler, "Is ARPA Suitable for Automatic Assessment of AIS Targets?," *Marine Navigation and Safety of Sea Transportation*, pp. 223–232, 2013. doi: https://doi.org/10.1201/b14961-40
17. V.I. Konoverts, N.B. Smyrinska, "Ways to enhance awareness of the maritime situation through the integration of shipborne radars into the surface situation display system," *Collection of Scientific Works of Kharkiv National University of the Air Force*, 4(66), pp. 71–78, 2020. doi: https://doi.org/10.30748/zhups.2020.66.10
18. A.M. Ponsford, "Radars for Maritime Domain Awareness," *Conference "Military Radar Summit," Arlington (VA), Virginia, 2015*. doi: http://dx.doi.org/10.13140/RG.2.1.3961.7687
19. A. Bole, A. Wall, and A. Norris, *Radar and ARPA Manual*; 3rd ed. Oxford, UK: Elsevier Butterworth-Heineman, 2014.
20. D. Luchenko, I. Georgiievskyi, and M. Bielikova, "Challenges and Developments in the Public Administration of Autonomous Shipping," *Lex Portus*, 9 (1), pp. 20–36, 2023. doi: 10.26886/2524-101X.9.1.2023.2
21. T. Plachkova, O. Avdieiev, "Public administration of safety of navigation: Multilevel challenges and answers," *Lex Portus*, 5 (25), pp. 34–62, 2020. doi: 10.26886/2524-101X.5.2020.2
22. O. Melnyk, Y. Bychkovsky, and A. Voloshyn, "Maritime situational awareness as a key measure for safe ship operation," *Scientific Journal of Silesian University of Technology. Series Transport*, 114, pp. 91–101, 2022. doi: https://doi.org/10.20858/sjsutst.2022.114.8
23. O. Melnyk, M. Malaksiano, "Effectiveness assessment of non-specialized vessel acquisition and operation projects, considering their suitability for oversized cargo transportation," *Transactions on Maritime Science*, 9 (1), pp. 23–34, 2020. doi: 10.7225/toms.v09.n01.00223
24. O. Melnyk et al., "Autonomous Ships Concept and Mathematical Models Application in their Steering Process Control," *TransNav*, 16 (3), pp. 553–559, 2022. doi: 10.12716/1001.16.03.18
25. O. Melnyk, S. Onyshchenko, "Navigational safety assessment based on Markov-model approach," *Scientific Journal of Maritime Research*, 36 (2), pp. 328–337, 2022. doi: https://doi.org/10.31217/p.36.2.16
26. O. Melnyk, S. Onyshchenko, O. Onishchenko, O. Lohinov, and V. Ocheretna, Integral approach to vulnerability assessment of ship's critical equipment and systems, *Transactions on Maritime Science*, vol. 12, no. 1, 2023. doi: https://doi.org/10.7225/toms.v12.n01.002
27. O. Melnyk et al., "Review of Ship Information Security Risks and Safety of Maritime Transportation Issues," *TransNav*, 16 (4), pp. 717–722, 2022. doi: 10.12716/1001.16.04.13
28. O. Melnyk et al., "Study of Environmental Efficiency of Ship Operation in Terms of Freight Transportation Effectiveness Provision," *TransNav*, vol. 16, no. 4, pp. 723–722, 2022. doi: 10.12716/1001.16.04.14
29. O. Melnyk et al., "Application of Fuzzy Controllers in Automatic Ship Motion Control Systems," *International Journal of Electrical and Computer Engineering*, vol. 13, no.4, pp. 3948–3957, 2023. doi: 10.11591/ijece.v13i4.pp3948-3957

30. Y. Volyanskaya, S. Volyanskiy, A. Volkov, and O. Onishchenko, "Determining energy-efficient operation modes of the propulsion electrical motor of an autonomous swimming apparatus," *Eastern-European Journal of Enterprise Technologies*, 6 (8-90), pp. 11–16, 2017. doi: 10.15587/1729-4061.2017.118984
31. V.A. Golikov, V.V. Golikov, Y. Volyanskaya, O. Mazur, and O. Onishchenko, "A simple technique for identifying vessel model parameters," *IOP Conference Series*: *Earth and Environmental Science*, 172 (1), art. no. 012010, 2018. doi: 10.1088/1755-1315/172/1/012010
32. V. Budashko, V. Nikolskyi, O. Onishchenko, and S. Khniunin, "Decision support system's concept for design of combined propulsion complexes," *Eastern-European Journal of Enterprise Technologies*, 3 (8-81), pp. 10–21, 2016. doi: 10.15587/1729-4061.2016.72543
33. V. Budashko, T. Obniavko, O. Onishchenko, Y. Dovidenko, and D. Ungarov, "Main Problems of Creating Energy-efficient Positioning Systems for Multipurpose Sea Vessels," *2020 IEEE 6th International Conference on Methods and Systems of Navigation and Motion Control, MSNMC 2020 - Proceedings*, art. no. 9255514, pp. 106–109. doi: 10.1109/MSNMC50359.2020.9255514
34. G.K. Lavrenchenko, A.G. Slinko, A.S. Boychuk, S.V. Kozlovskyi, and V.M. Halkin, "Conversion of liquid to steam. How and why?," *Journal of Chemistry and Technologies*, 31 (3), pp. 678–684, 2023. doi: 10.15421/jchemtech.v31i3.285771
35. L.A. Frolova, T.V. Hrydnieva, "Influence of various factors on the ferric α-oxyhydroxide synthesis," *Journal of Chemistry and Technologies*, 28 (1), pp. 61–67, 2020. doi: 10.15421/082008
36. A. Bondar, N. Bushuyeva, S.D. Bushuyev, and S. Onyshchenko, "Modelling of Creation Organisational Energy-Entropy," *International Scientific and Technical Conference on Computer Sciences and Information Technologies*, 2, art. no. 9321997, pp. 141–145, 2020. doi: 10.1109/CSIT49958.2020.9321997
37. S. Onyshchenko, A. Bondar, V. Andrievska, N. Sudnyk, and O. Lohinov, "Constructing and exploring the model to form the road map of enterprise development," *Eastern-European Journal of Enterprise Technologies*, 5 (3-101), pp. 33–42, 2019. 10.15587/1729-4061.2019.179185
38. S. Bushuyev, V. Bushuieva, S. Onyshchenko, and A. Bondar, "Modeling the dynamics of information panic in society. COVID-19 case," *CEUR Workshop Proceedings*, 2864, pp. 400–408, 2021.
39. A. Bondar, S. Onyshchenko, O. Vishnevska, D. Vishnevskyi, S. Glovatska, and A. Zelenskyi, "Constructing and investigating a model of the energy entropy dynamics of organizations," *Eastern-European Journal of Enterprise Technologies*, 3 (3-105), pp. 50–56, 2020. doi: 10.15587/1729-4061.2020.206254
40. O. Scherbina, O. Drozhzhyn, O. Yatsenko, and O. Shybaev, "Cooperation forms between participants of the inland waterways cargo delivery: a case study of the Dnieper region," *Scientific Journal of Silesian University of Technology. Series Transport*, 103, pp. 155–166, 2019. doi: 10.20858/sjsutst.2019.103.12
41. A. Shibaev, S. Borovyk, and I. Mykhailova, "Developing a strategy for modernizing passenger ships by the optimal distribution of funds," *Eastern-European Journal of Enterprise Technologies*, 6 (3-108), pp. 33–41, 2020. doi: 10.15587/1729-4061.2020.219293
42. D.S. Minchev et al., "Prediction of centrifugal compressor instabilities for internal combustion engines operating cycle simulation," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 237 (2-3), pp. 572–584, 2023. doi: 10.1177/09544070221075419
43. R. Varbanets et al., "Concept of Vibroacoustic Diagnostics of the Fuel Injection and Electronic Cylinder Lubrication Systems of Marine Diesel Engines," *Polish Maritime Research*, 29 (4), pp. 88–96, 2022. doi: 10.2478/pomr-2022-0046

44. M. Stetsenko et al., "Improving Navigation Safety by Utilizing Statistical Method of Target Detection on the Background of Atmospheric Precipitation," *Trends in Sustainable Computing and Machine Intelligence - Proceedings of ICTSM 2023*. doi: https://doi.org/10.1007/978-981-99-9436-6_8
45. D. Korban, O. Melnyk, O. Onishchenko, S. Kurdiuk, V. Shevchenko, and T. Obniavko, "Radar-based detection and recognition methodology of autonomous surface vehicles in challenging marine environment," *Scientific Journal of Silesian University of Technology. Series Transport*, 122, pp. 111–127, 2024. doi: 10.20858/sjsutst.2024.122.7

Received 30.01.2024

INFORMATION ON THE ARTICLE

Dmytro V. Korban, ORCID: 0000-0002-6798-2526 , National University "Odesa Maritime Academy", Ukraine, e-mail: korbandmv@gmail.com

Oleksiy M. Melnyk, ORCID: 0000-0001-9228-8459, Odesa National Maritime University, Ukraine, e-mail: m.onmu@ukr.net

Serhii V. Kurdiuk, ORCID: 0000-0002-3165-4571, National University "Odesa Maritime Academy", Ukraine, e-mail: serega15507@ukr.net

Oleg A. Onishchenko, ORCID: 0000-0002-3766-3188, National University "Odesa Maritime Academy", Ukraine, e-mail: oleganaton@gmail.com

Valentyna V. Ocheretna, ORCID: 0000-0003-4077-6711, Odesa National Maritime University, Ukraine, e-mail: v.ocheretna@ukr.net

Olha V. Shcherbina, ORCID: 0000-0002-9247-5972, Odesa National Maritime University, Ukraine, e-mail: olshcherbina@i.ua

Oleg V. Kotenko, ORCID: 0009-0007-5294-474X, Odesa National Maritime University, Ukraine, e-mail: kot_ov@ukr.net

МЕТОД ПОЛЯРИЗАЦІЙНОЇ СЕЛЕКЦІЇ НАВІГАЦІЙНИХ ОБ'ЄКТІВ У СКЛАДНИХ МЕТЕОРОЛОГІЧНИХ УМОВАХ З ВИКОРИСТАННЯМ СТАТИСТИЧНИХ ВЛАСТИВОСТЕЙ РАДІОСИГНАЛІВ / Д.В. Корбан, О.М. Мельник, С.В. Курдюк, О.А. Онищенко, В.В. Очеретна, О.В. Щербина, О.В. Котенко

Анотація. Присвячено дослідженню та застосуванню поляризаційної селекції для навігаційних об'єктів у складних атмосферних умовах. Основний акцент зроблено на використанні статистичних властивостей поляризаційних параметрів частково поляризованих ехо-сигналів. Детально розглянуто статистичні властивості поляризаційних параметрів частково поляризованих ехо-сигналів, які можуть бути використані для підвищення точності суднових радіолокаційних систем. Дослідження ґрунтується на аналізі експериментальних даних, зібраних у різних атмосферних умовах. Отримані результати свідчать про ефективність поляризаційної селекції для підвищення стійкості та точності навігаційних радіолокаційних систем у різних атмосферних умовах. Використання статистичних методів дозволяє радіолокаційній системі адаптуватися до мінливих умов, забезпечуючи надійність у різних сценаріях. Поляризаційна селекція на основі статистичних властивостей поляризаційних параметрів є перспективним методом покращення навігації в умовах підвищеної атмосферної вологості, туману та інших складних атмосферних умов і може бути використана у розробленні сучасних систем навігації.

Ключові слова: безпека судноплавства, атмосферні умови, статистичні властивості, часткова поляризація, ехо-сигнали, радіолокаційні системи, навігаційне обладнання, ресурси навігаційного містка, морський транспорт, радіолокація, керованість і маневрування суден.

UDC 004.032.26+519.2+621.315.614.7
DOI: 10.20535/SRIT.2308-8893.2025.1.07

OPTIMAL SELECTION OF COTTON WARP SIZING PARAMETERS UNDER SYSTEM RESEARCH LIMITATION

H.S. TKACHUK, V.V. ROMANUKE, A.V. TKACHUK

Abstract. Warp sizing is the process of applying the sizing agents to the warp yarn to improve its weavability along with improving the economic performance of weaving. We consider a finite set of sizing agents or parameters mapped into a finite set of sizing quality indicators. Due to various limitations of material and time resources, exhaustive system research and constructing an information technology to interpret and optimize sizing data is impossible. Therefore, we suggest an algorithm for controlling warp sizing quality under system research limitation, where optimal selection of cotton warp sizing parameters is exemplified. The algorithm utilizes a set of basis vectors of sizing parameters corresponding to a set of respective vectors of quality indicators. The method of radial basis functions is used to determine the probabilistically appropriate vector of quality indicators for any given vector of sizing parameters. The uncountably infinite space of sizing vectors is uniformly sampled into a finite space. The finite space may be refined by excluding sizing vectors corresponding to inadmissible values of one or more quality indicators. A set of Pareto-efficient sizing vectors is determined within the finite (refined) space, and an optimal, efficient sizing vector is determined as one being the closest to the unachievable sizing vector. The suggested algorithm serves as a method of optimal selection of warp sizing parameters, resulting in improved performance of warp yarns that can withstand repeated friction, stretching, and bending on the loom without causing a lot of fluffing or breaking. The algorithm is not limited to cotton, and it can be applied to any yarn material by an experimentally adjusted radial basis function spread.

Keywords: warp sizing, sizing agents, colloidal systems, inorganic compounds, sizing quality indicators, radial basis function, Pareto efficiency.

INTRODUCTION

Manufacture of high-quality fabric is a very important industrial branch whose impact cannot be overestimated. The basis of high-quality fabric is high-quality thread. Spinning high-quality thread is not that much complicated process, but it relies on technologically correct and efficient sizing applied to thread [1; 2]. The purpose of the sizing is to improve the breakage characteristics of the yarn and increase its resistance to friction and multi-cycle loads on the loom during fabric production. Efficient sizing is required for efficient textile manufacturing. The latter is a major industry largely based on the conversion of fiber into yarn, then yarn into fabric which is subsequently dyed and fabricated into cloth [3; 4].

The sizing is applied to single-threaded yarn. An adhesive substance is applied to the surface of the threads, which covers the threads after drying with a smooth elastic film. This reduces the breakage of the threads, protects them from rubbing against the parts of the loom, improves abrasion resistance of the yarn, and decreases hairiness of the yarn. In the process of sizing, the warp threads must be glued evenly along their entire length and the width of the fill [1; 2; 5].

© *Publisher IASA at the Igor Sikorsky Kyiv Polytechnic Institute*, 2025

The protective film of the sizing should have approximately the same elongation indicators as the warp threads, and also provide the threads with great uniformity, wear resistance, and durability under repeated loads [5; 6]. The sizing film should not fall off, and the thread impregnated with it should not be brittle. The sizing should have a good affinity for the fibrous material, not spoil the yarn and weaving equipment, be easy to get desized (washed), and be relatively cheap [7; 8].

The sizing recipe and its parameters are determined by the type of yarn material (e. g., cotton, polyester, linen), the thickness of the yarn, the type of weaving machinery, and conditions under which the fabric weaved from the yarn is assumed to be used [2; 9; 10]. For sizing of cotton yarn (warp), starches are used as the main component of the sizing compositions [11; 12]. Starches are relatively cheap, are characterized by a reliable raw material base, and are completely biodegradable without harming the environment. Nevertheless, the films formed by starch have an unsatisfactory set of physical and mechanical indicators [4; 5; 13]. Another issue is the cost of the sizing [2; 8]. Depending on the type of adhesive, the sizing material can be from 23% to 78% of the sizing process cost. Moreover, the cost of energy consumption accounts for from 9% to 24% of the sizing process cost. Therefore, developing new sizing technologies is aimed at improving the economic performance of weaving [8; 14]. Thus, a physico-chemical rationale is provided in [15] for the technology of cotton warp sizing by using starches with hygroscopic additions of kaolin or potassium alum. However, the parameters of the cotton warp sizing are basically taken from rule of thumb, rather than from an exhaustive system research and constructing an information technology to interpret and optimize sizing data [5; 7; 9; 16]. This is so due to the physico-chemical research of colloidal systems with inorganic compounds is resource-intensive in both time and materials [1; 4; 5; 9; 15].

PROBLEM STATEMENT

Conducting an exhaustive system research would give a sufficient amount of sizing data which subsequently could be optimized to further improve sizing quality. However, it is impossible due to various limitations of material and time resources. Therefore, the goal of our research is to suggest an algorithm of optimally selecting cotton warp sizing parameters under system research limitation, i.e. when a limited amount of data is available. In general terms, this is about to control sizing quality. To achieve the goal, we have to accomplish the following four tasks:

1. To report and describe main characteristics of materials and sizing agents used for cotton warp sizing during real experiments. This is done for repeatability of the sizing research.
2. To suggest a consistent algorithm to control sizing quality. Apart from the verifiability, the algorithm consistency implies also its independence on the number of sizing parameters and the number of quality-controlled factors.
3. To apply the suggested algorithm to the results of the factually conducted experiments.
4. To discuss and conclude on the significance, practical applicability, and contribution of the suggested algorithm as a method of optimal selection of cotton warp sizing parameters under system research limitation.

CHARACTERISTICS OF MATERIALS AND SIZING AGENTS

The quality of sizing is assessed through experimenting with various parameters of the cotton thread sizing and measuring characteristics of the sized thread. The following sizing agents and chemical materials are used for the experiments [14; 15]:

1. Water H_2O for household and drinking purposes.
2. Kaolin $Al_2Si_2O_5(OH)_4$ or, in oxide notation, $Al_2O_3 \cdot 2SiO_2 \cdot 2H_2O$ — a white mass (a shade of another color is possible), soft to the touch, insoluble in water.
3. Potassium alum $Al_2(SO_4)_3 \cdot K_2SO_4 \cdot 24H_2O$ — colorless cubic crystals soluble in water.
4. Soft paraffins — mixtures of hydrocarbons of the methane series with a normal structure in the range $C_{19}H_{40}$ — $C_{35}H_{72}$. They are derived from petroleum, their molecular mass is 300 to 400, melting point is between 50 and 54 ºC, and the oil content does not exceed 2.3%.
5. Corn starch of a general composition $(C_6H_{10}O_5)_n$.
6. Polyvinyl alcohol $[CH_2CH(OH)]_n$ — a colorless, odorless, weakly hygroscopic, water-soluble powder. For sizing, it is used in the form of granules with a size of 0.3 — 1.7 mm.
7. Syntanol DS-10 — a mixture of polyethylene glycol ethers of synthetic fatty alcohols $C_nH_{2n+1}O(C_2H_4O)_mH$, where $n \in \overline{\{10, 18\}}$, $m \in \overline{\{8, 10\}}$. It is a nonionic material in the form of a soft white or light yellow paste, biodegradable, well soluble in water at 30 — 40 ºC.
8. Caustic sodium NaOH — white rhombic blurring crystals, a caustic substance.
9. Hydrogen peroxide H_2O_2 — colorless liquid.
10. Iodine I_2 — purple-black rhombic crystals with a metallic luster with a density of 4.933 g/cm^3.
11. Potassium iodide KI — colorless cubic crystals, soluble in water.
12. Phenolphthalein — is a polyfunctional polynuclear aromatic compound whose crystals are soluble in alcohol. It is an acid-base indicator with a pH range 8.3 — 10.
13. Sodium bromide dihydrate $NaBr \cdot 2H_2O$ — colorless monoclinic crystals, well soluble in water.
14. Zinc sulfate heptahydrate $ZnSO_4 \cdot 7H_2O$ — colorless crystals soluble in water.
15. Copper sulfate $CuSO_4 \cdot 5H_2O$ — blue triclinic crystals, well soluble in water, losing water of crystallization at 110 ºC.
16. Ammonium chloride NH_4Cl — colorless cubic crystals, well soluble in water.
17. Potassium dichromate $K_2Cr_2O_7$ — orange monoclinic or triclinic crystals, well soluble in water.

The research of the technological parameters of the sizing process and quality indicators of the sized warp is carried out with the use of the cotton yarn of class 1, number 34. It is characterized by the following indicators:

1. Nominal linear density is 29 Tex.
2. Specific breaking load (tenacity) is 12.1 cN/Tex.
3. Coefficient of variation by breaking load is 10.4%.
4. Quality indicator is 0.859.

The above-mentioned characteristics and properties are intentionally reported for repeatability of the sizing research. More specificities about the sizing experiments can be found in [15].

CONTROL OF SIZING QUALITY

Consider a warp with F sizing parameters or features compiled into a numerical vector $\mathbf{U}=[u_i]_{1\times F}$ of positive values $\{u_i\}_{i=1}^{F}$. Denote a space of all feasible combinations of F sizing parameters by U, where $\mathbf{U}\in U$. During practical experiments with a definite set of F sizing parameters

$$\mathbf{U}_c=[u_i^{(c)}]_{1\times F}\in U \tag{1}$$

numbered by c, we measure a quality-controlled factor (numbered by j) and denote the average of its measured values by $\widetilde{y}_c^{(j)}$, $c=\overline{1,C}$, $j=\overline{1,J}$, where C is the number of distinct sets of sizing parameters, and J is the number of distinct quality-controlled factors. Vector (1) should be standardized to provide comparability:

$$v_i^{(c)}=\frac{u_i^{(c)}}{\max\limits_{k=\overline{1,C}} u_i^{(k)}},\quad c=\overline{1,C},\quad i=\overline{1,F}. \tag{2}$$

Thus, values (2) are compiled into a vector

$$\mathbf{V}_c=[v_i^{(c)}]_{1\times F}\in V, \tag{3}$$

where every $v_i^{(c)}\in(0;1]$ and V is a standardized space U.

For a set of F sizing parameters $\mathbf{V}=[v_i]_{1\times F}\in V$, we can use the method of radial basis functions [17] to ascertain the topological location of vector $\mathbf{V}$ within set $\{\mathbf{V}_c\}_{c=1}^{C}\subset V$. A value proportional to the probability of a similarity between vectors $\mathbf{V}$ and $\mathbf{V}_c$ is [18]

$$p_c(\mathbf{V})=e^{-d_c},\quad c=\overline{1,C}, \tag{4}$$

by

$$d_c=\frac{\sum\limits_{i=1}^{F}[v_i-v_i^{(c)}]^2}{2\sigma^2}, \tag{5}$$

where σ is a radial basis function spread [17; 18]. Then the probability of a similarity between vectors $\mathbf{V}$ and $\mathbf{V}_c$ is

$$p_c^*(\mathbf{V})=\frac{p_c(\mathbf{V})}{\sum\limits_{k=1}^{C} p_k(\mathbf{V})},\quad c=\overline{1,C}. \tag{6}$$

Therefore, a weighted value of the j-th quality-controlled factor is

$$\widetilde{y}^{(j)}(\mathbf{V})=\sum_{c=1}^{C}\widetilde{y}_c^{(j)}p_c^*(\mathbf{V}),\quad j=\overline{1,J}. \tag{7}$$

The sizing quality is commonly based on quality-controlled factors (quality indicators) which should be maximized. Thus, we have to solve J maximization problems

$$\mathbf{V}^{*(j)} \in \arg\max_{\mathbf{V}\in V} \widetilde{y}^{(j)}(\mathbf{V}) \subset V\ ,\ j=\overline{1,J}\ . \tag{8}$$

However, it is highly probable that solutions $\{\mathbf{V}^{*(j)}\}_{j=1}^{J}$ to J maximization problems (8) are different. This means that a J-criterion problem

$$V^{*(j)} = \arg\max_{\mathbf{V}\in V} \widetilde{y}^{(j)}(\mathbf{V}) \subset V\ ,\ j=\overline{1,J}\ , \tag{9}$$

does not have an exact solution, i.e.

$$\bigcap_{j=1}^{J} V^{*(j)} = \varnothing\ .$$

An approximate solution to this problem can be found as follows [19; 20]. First, we have to find a set of Pareto-efficient points. So, we have to find every Pareto-efficient point $\mathbf{V}^{**}$, at which inequalities

$$\widetilde{y}^{(j)}(\mathbf{V}) \geqslant \widetilde{y}^{(j)}(\mathbf{V}^{**}) \quad \forall\, j=\overline{1,J} \tag{10}$$

are impossible for any $\mathbf{V}\in V$ unless $\exists\mathbf{V}_0 \in V$ such that

$$\widetilde{y}^{(j)}(\mathbf{V}_0) = \widetilde{y}^{(j)}(\mathbf{V}^{**}) \quad \forall\, j=\overline{1,J}\ . \tag{11}$$

Suppose that set U is sampled (uniformly or close to that) into M sample vectors

$$\{\mathbf{U}^{(m)}\}_{m=1}^{M} \subset U\ ,$$

which are subsequently standardized to a set V_M of M sample vectors

$$\{\mathbf{V}^{(m)}\}_{m=1}^{M} = V_M \subset V\ , \tag{12}$$

whose entries are within half-interval $(0;1]$. A set $\mathbf{V}$ of H Pareto-efficient points is then determined for vectors (12) by using (10), (11), where

$$\mathbf{V} = \{\mathbf{V}^{**(h)}\}_{h=1}^{H} \subset V_M \tag{13}$$

and $\mathbf{V}^{**(h)}$ is an h-th Pareto-efficient point, $h=\overline{1,H}$ and H is a number of efficient sizing configurations. Weighted values

$$\{\{\widetilde{y}^{(j)}(\mathbf{V}^{**(h)})\}_{h=1}^{H}\}_{j=1}^{J} \tag{14}$$

of the quality-controlled factors calculated by (7) are further standardized as

$$\widetilde{y}_1^{(j)}(\mathbf{V}^{**(h)}) = \frac{\widetilde{y}^{(j)}(\mathbf{V}^{**(h)})}{\max\limits_{l=\overline{1,H}} \widetilde{y}^{(j)}(\mathbf{V}^{**(l)})}\ ,\ h=\overline{1,H}\ ,\ j=\overline{1,J}\ . \tag{15}$$

Then the distance to the (most likely, unachievable) unit point in R^J is calculated for every Pareto-efficient point:

$$\rho_h = \sum_{j=1}^{J} [1-\widetilde{y}_1^{(j)}(\mathbf{V}^{**(h)})]^2\ ,\ h=\overline{1,H}\ . \tag{16}$$

Finally, the best Pareto-efficient point is the closest to the unit point, and thus

$$\mathbf{V}^{***}=[v_i^{***}]_{1\times F}\in\arg\min_{\{\mathbf{V}^{**(h)}\}_{h=1}^{H}}\rho_h=$$

$$=\arg\min_{\{\mathbf{V}^{**(h)}\}_{h=1}^{H}}\sum_{j=1}^{J}[1-\widetilde{y}_1^{(j)}(\mathbf{V}^{**(h)})]^2\,. \qquad (17)$$

Hence, $\mathbf{V}^{***}$ by (17) is the optimal configuration of the sizing parameters standardized according to ratio (2).

To get back to real values of sizing parameters, we unstandardize entries of vector $\mathbf{V}^{***}$ by using ratio (2):

$$u_i^{***}=v_i^{***}\max_{k=\overline{1,C}}u_i^{(k)}\,,\quad i=\overline{1,F}. \qquad (18)$$

Thus, vector $\mathbf{U}^{***}=[u_i^{***}]_{1\times F}$ contains the optimal values of the sizing parameters. Real values of the quality-controlled factors are calculated similarly by unstandardizing entries of vector

$$\widetilde{\mathbf{Y}}_1(\mathbf{V}^{***})=[\widetilde{y}_1^{(j)}(\mathbf{V}^{***})]_{1\times J} \qquad (19)$$

by using ratio (15):

$$\widetilde{y}^{(j)}(\mathbf{V}^{***})=\widetilde{y}_1^{(j)}(\mathbf{V}^{***})\cdot\max_{l=\overline{1,H}}\widetilde{y}^{(j)}(\mathbf{V}^{**(l)}),\quad j=\overline{1,J}\,, \qquad (20)$$

where upon vector

$$\widetilde{\mathbf{Y}}(\mathbf{V}^{***})=[\widetilde{y}^{(j)}(\mathbf{V}^{***})]_{1\times J} \qquad (21)$$

contains the best values of the quality-controlled factors.

COTTON WARP SIZING

During the real-time experimental research of the cotton yarn of class 1, three sizing parameters were studied:

1. The amount of starch, g/liter ($i=1$).
2. The amount of hydrophilic component of kaolin or potassium alum as a percentage of the starch mass ($i=2$).
3. The amount of soft paraffin plasticizer as a percentage of the starch mass ($i=3$).

An exhaustive experimental research is impossible due to the research of every feasible combination of these three sizing agents spans up to 36 hours, let alone spending other material resources. Thus, only marginal values of the sizing agents were used to control the sizing quality (Table 1).

Table 1. Combinations of the three sizing agents for the cotton yarn of class 1

Number of the distinct combination, c	1	2	3	4	5	6	7	8
Amount of starch $u_1^{(c)}$, g/liter	40	40	40	40	60	60	60	60
Amount of hydrophilic component of kaolin or potassium alum $u_2^{(c)}$, %	0.1	0.1	1	1	0.1	0.1	1	1
Amount of soft paraffin plasticizer $u_3^{(c)}$, %	0.8	1.5	0.8	1.5	0.8	1.5	0.8	1.5

In fact, Table 1 shows up a matrix of eight vectors (1) for this study case. The sizing quality here is studied for three quality indicators:

4. The tenacity or relative breaking strength ($j = 1$), cN/Tex.
5. The percentage of breaking elongation ($j = 2$).
6. The percentage of adhesion strength ($j = 3$).

The averages of the quality indicators are presented in Table 2.

T a b l e 2 . The averaged values of quality indicators for combinations in Table 1

Number of the distinct combination, c	**Averages of quality indicators**		
	Relative breaking strength $\tilde{y}_c^{(1)}$	Percentage of breaking elongation $\tilde{y}_c^{(2)}$	Percentage of adhesion strength $\tilde{y}_c^{(3)}$
1	13.675	5.05	4.425
2	13.25	5.15	4.125
3	13.275	5.175	4.4
4	13.25	5.225	4.4
5	16.825	4.8	7.1
6	16.1	4.85	6.625
7	16.65	4	7.25
8	16.75	4.775	6.85

Consequently, by the method of controlling the sizing quality in accordance with formulae (1)–(21) with an experimentally adjusted spread of $\sigma = 0.1$, we have eight sets of sizing parameters

$$\{\mathbf{U}_c\}_{c=1}^{8} = \{[u_i^{(c)}]_{1\times 3}\}_{c=1}^{8} \in U \tag{22}$$

which are standardized into eight sets

$$\{\mathbf{V}_c\}_{c=1}^{8} = \{[v_i^{(c)}]_{1\times 3}\}_{c=1}^{8} \in V$$

by (2) as

$$v_i^{(c)} = \frac{u_i^{(c)}}{\max\limits_{k=\overline{1,8}} u_i^{(k)}}, \quad c = \overline{1,8}, \quad i = \overline{1,3}.$$

We sample set (22) uniformly into 1000 to 343000 sample vectors and determine the number of Pareto-efficient points according to (10), (11). In order to prevent an excessive adhesion strength of over 6%, we exclude from set (13) all Pareto-efficient points such, for which

$$\tilde{y}_1^{(3)}(\mathbf{V}^{**(h)}) > 6, \quad h = \overline{1,H}. \tag{23}$$

Thus, set (13) is refined with a fewer number H of Pareto-efficient points. Formulae (14)–(17) are subsequently applied and the optimal values of the three sizing parameters by (18) are

$$u_i^{***} = v_i^{***} \max_{k=\overline{1,8}} u_i^{(k)}, \quad i = \overline{1,3}. \tag{24}$$

The best values of the relative breaking strength, breaking elongation percentage, and adhesion strength percentage are

$$\widetilde{y}^{(j)}(\mathbf{V}^{***}) = \widetilde{y}_1^{(j)}(\mathbf{V}^{***}) \cdot \max_{l=\overline{1,H}} \widetilde{y}^{(j)}(\mathbf{V}^{**(l)}), \quad j=\overline{1,3}, \tag{25}$$

respectively. The best values (25) are obtained by the optimal amounts of starch, hydrophilic component of kaolin or potassium alum, and soft paraffin plasticizer by (24) or, in standardized units, by (17). The results of solving the problem are presented in Table 3, where $M^{(6)}$ is a number of sample vectors after refinement by (23) prior to determining set $\mathbf{V}$ of H Pareto-efficient points. It is clearly seen that the optimal values of the three sizing parameters (24) and the best values of the quality indicators (25) depend on the sampling. In particular, the amount of hydrophilic component of kaolin or potassium alum badly depends on the sampling, ranging from it minimum to maximum possible percentages. The amount of soft paraffin plasticizer varies much less, but its efficient value is mostly at the upper bound (i.e., $u_3^{***}=1.5$). Meanwhile, the amount of starch varies the least, and its relative deviation is just a bit greater than 1 g/liter. The quality indicators obtained by the optimal values of the three sizing parameters vary also, but their variation decreases as the sampling becomes denser.

Starting from $M=15625$ up to $M=343000$, the variation of the best Pareto-efficient point entries (24) significantly decreases. The minimum, maximum, and average for $M \geqslant 15625$ are presented in Table 3 also. The variation of the starch amount does not exceed 0.66 g/liter, the amount of hydrophilic component of kaolin or potassium alum ranges from 0.55 to 1, whereas the amount of soft paraffin plasticizer remains constantly at the upper bound. Fig. 1 showing the variation of u_1^{***} confirms its trend to decreasing (here and in the plots below the average value is shown with a horizontal line). To the contrary, Fig. 2 shows that the amount of hydrophilic component of kaolin or potassium alum varies at denser sampling as mush as it varies at sparser sampling down to at $M=15625$.

Table 3. Solutions (24), (25) of the three-criterion problem for the cotton yarn of class 1

M	$M^{(6)}$	H	u_1^{***}	u_2^{***}	u_3^{***}	$\widetilde{y}^{(1)}(\mathbf{V}^{***})$	$\widetilde{y}^{(2)}(\mathbf{V}^{***})$	$\widetilde{y}^{(3)}(\mathbf{V}^{***})$
1000	545	112	51.1111	0.6	1.5	15.519	4.9329	5.9889
1331	726	120	50	1	1.5	15	5	5.625
1728	942	139	50.9091	0.9182	1.5	15.4328	4.9444	5.9279
2197	1219	134	51.6667	0.1	1.2667	15.3078	4.9335	5.9266
2744	1477	183	50.7692	0.9308	1.5	15.3683	4.9526	5.8828
3375	1855	167	51.4286	0.55	1.5	15.4364	4.9293	5.9668
4096	2199	204	50.6667	1	1.5	15.3204	4.9588	5.8493
4913	2673	219	51.25	0.55	1.5	15.3676	4.9374	5.9132
5832	3239	337	51.7647	0.5235	1.5	15.3624	4.929	5.9631
6859	3772	322	51.1111	0.6	1.5	15.519	4.9329	5.9889
8000	4410	410	51.5789	0.1	1.2053	15.3596	4.9283	5.955
9261	5061	374	51	0.595	1.5	15.4669	4.9394	5.9526
10648	5831	495	51.4286	0.1	1.2	15.3226	4.932	5.9198
12167	6723	539	50.9091	0.9182	1.5	15.4328	4.9444	5.9279
13824	7632	538	51.3043	0.1391	1.1652	15.4105	4.9225	5.9695
15625	8582	640	50.8333	0.925	1.5	15.398	4.9488	5.9036
17576	9643	534	51.2	0.568	1.5	15.4917	4.9316	5.9789

Continued Tabl. 3

M	$M^{(6)}$	H	u_1^{***}	u_2^{***}	u_3^{***}	$\widetilde{y}^{(1)}(\mathbf{V}^{***})$	$\widetilde{y}^{(2)}(\mathbf{V}^{***})$	$\widetilde{y}^{(3)}(\mathbf{V}^{***})$
19683	10762	571	50.7692	0.9654	1.5	15.3683	4.9526	5.8828
21952	12254	629	51.1111	0.6	1.5	15.519	4.9329	5.9889
24389	13301	608	51.4286	0.55	1.5	15.4364	4.9293	5.9668
27000	15016	678	51.0345	0.6276	1.5	15.4891	4.9371	5.9674
29791	16186	693	51.3333	0.55	1.5	15.4	4.9336	5.9385
32768	18128	757	50.9677	0.9129	1.5	15.4594	4.9409	5.9466
35937	19825	1019	51.25	0.55	1.5	15.3676	4.9374	5.9132
39304	21674	870	50.9091	0.9182	1.5	15.4328	4.9444	5.9279
42875	23579	895	51.1765	0.5765	1.5	15.5162	4.931	5.9912
46656	25596	920	50.8571	1	1.5	15.409	4.9474	5.9113
50653	27964	1101	51.1111	0.575	1.5	15.4833	4.9348	5.9684
54872	30236	1198	50.8108	1	1.5	15.3876	4.9502	5.8963
59319	32700	1139	51.0526	0.5974	1.5	15.4918	4.9363	5.9699
64000	35198	1284	50.7692	0.9308	1.5	15.3683	4.9526	5.8828
68921	37878	1276	51	0.9775	1.5	15.474	4.939	5.9568
74088	40625	1332	51.2195	0.561	1.5	15.4542	4.9332	5.9592
79507	43855	1723	50.9524	0.9357	1.5	15.4525	4.9418	5.9417
85184	46622	1332	51.1628	0.5814	1.5	15.5225	4.9311	5.9939
91125	50238	1820	50.9091	0.9795	1.5	15.4328	4.9444	5.9279
97336	53455	1495	51.1111	0.58	1.5	15.497	4.9341	5.9763
103823	57382	1954	50.8696	0.9413	1.5	15.4147	4.9467	5.9153
110592	61306	2236	51.0638	0.5979	1.5	15.497	4.9357	5.9736
117649	64815	2062	50.8333	0.9063	1.5	15.398	4.9488	5.9036
125000	69187	2315	51.0204	0.9449	1.5	15.4832	4.9379	5.9632
132651	72897	1972	50.8	0.91	1.5	15.3826	4.9508	5.8928
140608	77641	2206	50.9804	1	1.5	15.4651	4.9402	5.9506
148877	82037	2037	51.1538	0.5673	1.5	15.4684	4.9343	5.9626
157464	86698	2372	50.9434	0.983	1.5	15.4484	4.9423	5.9389
166375	91996	2412	51.1111	0.5833	1.5	15.5036	4.9337	5.98
175616	96508	2555	50.9091	0.9182	1.5	15.4328	4.9444	5.9279
185193	102226	2618	51.0714	0.5982	1.5	15.5006	4.9352	5.9761
195112	107485	2807	50.8772	0.9526	1.5	15.4182	4.9462	5.9177
205379	113130	2768	51.0345	0.8914	1.5	15.4895	4.9371	5.9676
216000	118864	2557	50.8475	0.9847	1.5	15.4045	4.948	5.9082
226981	124826	3006	51	0.925	1.5	15.474	4.939	5.9568
238328	130884	2625	50.8197	0.9852	1.5	15.3917	4.9496	5.8992
250047	138396	3358	50.9677	0.9129	1.5	15.4594	4.9409	5.9466
262144	144298	3222	51.1111	0.5857	1.5	15.5073	4.9335	5.9822
274625	151787	3814	50.9375	0.9438	1.5	15.4457	4.9427	5.937
287496	158087	3464	51.0769	0.5846	1.5	15.4906	4.9355	5.9706
300763	165932	4032	50.9091	0.9182	1.5	15.4328	4.9444	5.9279
314432	173480	4270	51.0448	0.6239	1.5	15.4935	4.9365	5.9705
328509	180889	3533	50.8824	0.9338	1.5	15.4205	4.9459	5.9194
343000	189565	4689	51.0145	0.987	1.5	15.4805	4.9382	5.9614
Minimum			50	0.1	1.1652	15	4.9225	5.625
Maximum			51.7647	1	1.5	15.5225	5	5.9939
Average			51.0352	0.7421	1.4809	15.4325	4.9405	5.9377
Minimum for $M \geqslant 15625$			50.7692	0.55	1.5	15.3676	4.9293	5.8828
Maximum for $M \geqslant 15625$			51.4286	1	1.5	15.5225	4.9526	5.9939
Average for $M \geqslant 15625$			**51.0054**	**0.7965**	**1.5**	**15.4512**	**4.9403**	**5.9444**

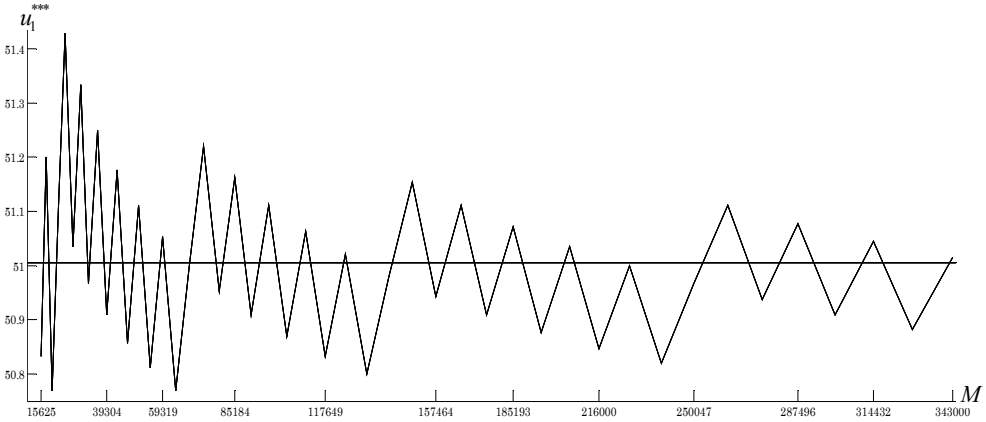


Fig. 1. The variation of the starch amount in the best efficient sizing

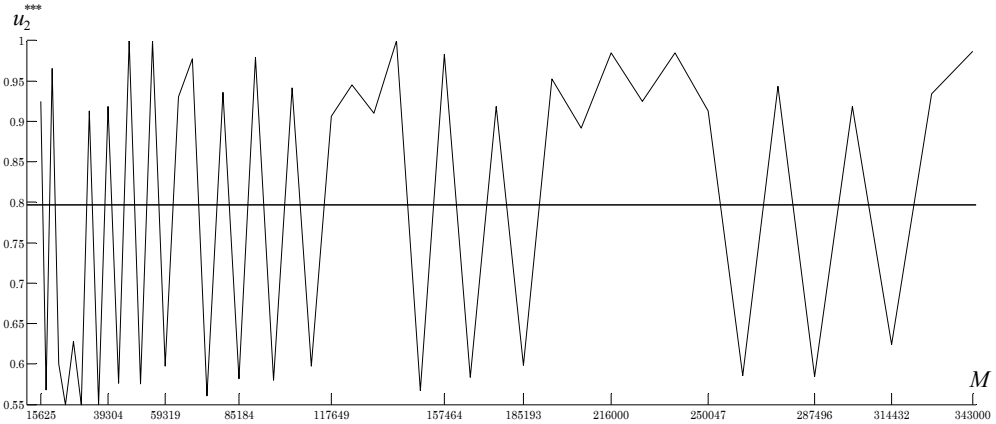


Fig. 2. The variation of the amount of hydrophilic component of kaolin or potassium alum

The relative breaking strength obtained at the optimal selection of cotton warp sizing parameters varies by 1% at most (Fig. 3). The percentage of breaking elongation varies by about 0.473% at most (Fig. 4). The variation of the percentage of adhesion strength is a little bit more significant — it varies by 1.889% at most (Fig. 5). Nevertheless, the above-mentioned decreasing trends of the variations are quite apparent in Figs. 3–5.

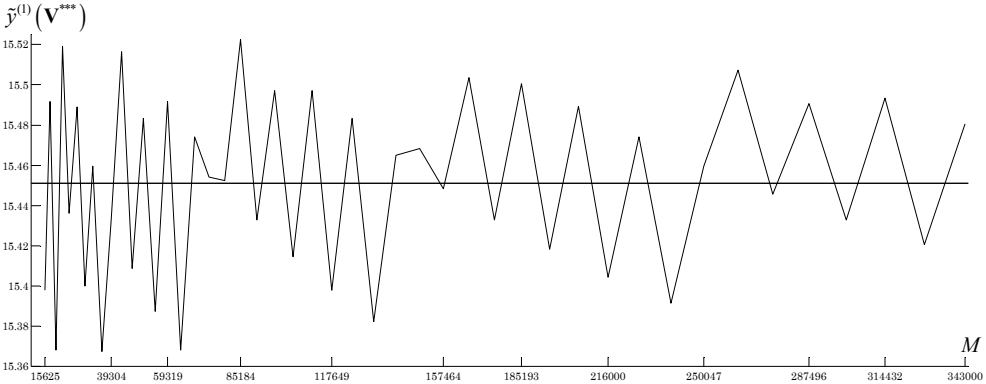


Fig. 3. The variation of the relative breaking strength

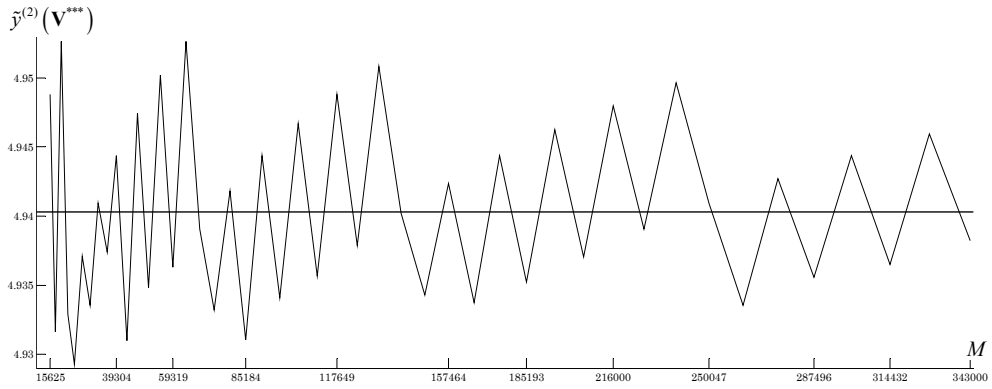


Fig. 4. The variation of the percentage of breaking elongation

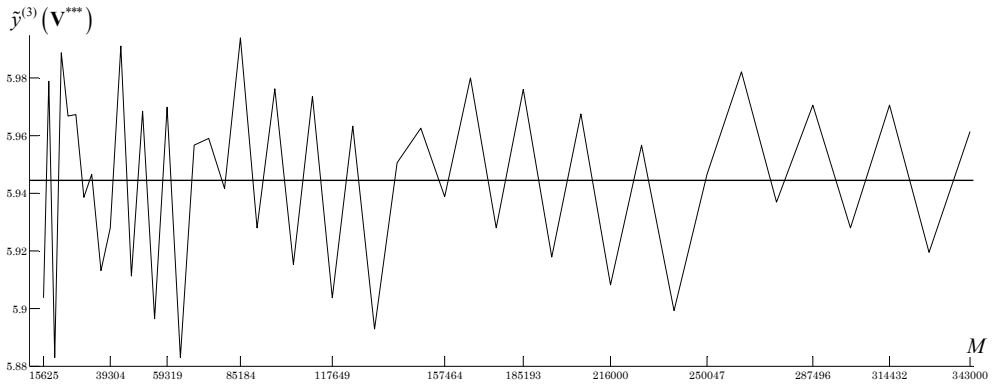


Fig. 5. The variation of the percentage of adhesion strength

It is worth mentioning that the variations of the optimal values for sizing agents and of the best quality indicators are explained with the probabilistic nature of the quality-controlled factors calculated by (7). The variations are further decreased by increasing the radial basis function spread σ, but then accuracy of the respective probabilistic classifier significantly drops [21; 22] and values of the quality-controlled factors calculated by (7) become irrelevant.

Therefore, the averages highlighted bold in Table 3 are considered as the final solutions (24), (25) of the three-criterion problem for the cotton yarn of class 1, where some rounding is still admissible, though. Thus, the optimal amount of starch is 51.0054 g/liter (however, by weighing it accurately to milligrams, it becomes 51.005 g/liter). The optimal amount of hydrophilic component of kaolin or potassium alum is 0.7965% (depending on weighing accuracy and technology, it can be rounded to 0.8%). The optimal amount of soft paraffin plasticizer is 1.5%. The obtained relative breaking strength is 15.4512 cN/Tex, which is 1.0828 times better than that obtained in [14; 15] by conducting a factorial experiment based on data in Tables 1 and 2. The breaking elongation percentage is less optimistic — it is 4.9403% versus the range from 5.12% to 5.4% reported in [15]. On average, $\tilde{y}^{(2)}(\mathbf{V}^{***})$ is 1.0647 times worse, but Table 3 shows that a 5% breaking elongation is hardly achievable for the given characteristics of materials and sizing agents. The percentage of adhesion strength is quite satisfactory — it is 5.9444% versus the range from 4.9% to 5.8% reported in [15].

DISCUSSION OF THE CONTRIBUTION

The method of radial basis functions renders ascertaining a topological location of any vector $\mathbf{V}$ within set $\{\mathbf{V}_c\}_{c=1}^{C} \subset V \subset \mathrm{R}^C$ into an approximation problem [23; 24]. This approximation problem can be considered as an interpolation approach [23; 25]. Thus, given a basis of C vectors $\{\mathbf{V}_c\}_{c=1}^{C} \subset V$, each of which corresponds to a distinct vector of quality indicators, the task is to determine a vector of quality indicators $\mathbf{Y} \in Y \subset \mathrm{R}^J$ for any vector $\mathbf{V} \in \mathrm{R}^C$. This task is solved by (4)–(7).

It is impossible to ascertain an analytical bond between uncountably infinite vector spaces $V \subset \mathrm{R}^C$ and $Y \subset \mathrm{R}^J$, but space V is sampled so that each element of the finite sampled space (12) can be mapped into a vector in R^J by using (4)–(7). The single variable of the mapping is the radial basis function spread σ. This parameter could be optimized during training the respective probabilistic neural network whose pattern matrix [22] would consist of vectors $\{\mathbf{V}_c\}_{c=1}^{C}$ concatenated into an $F \times C$ matrix of these vectors transposed into columns.

Upon the mapping, a set of Pareto-efficient vectors within subset (12) is determined. This relieves from considering useless vectors of quality-controlled factors whose values are below the already achievable quality level. More specifically, the efficiency selection saves memory and computational resources. It is seen from Table 3 that, in the case of the experimental research of the cotton yarn of class 1, the percentage of the number of Pareto-efficient vectors H with respect to the number of sampled vectors M does not exceed 11.2%.

Formally, optimization problem (17) is always solvable, but its practically consistent solvability depends on selecting a reasonably dense sampling of space U and spread σ. The approximation error has an upper bound [25; 26], which is exemplarily seen in Figs. 1–5, but its estimates would heavily depend on M and σ, as well as on the experimental sizing itself. Figs. 1–5 also show that practical convergence is possible by a moderate number of sample vectors.

Refining set (13), where vectors with inadmissible values of one or more quality indicators are excluded, is optional. In the particularly conducted experiments, overly adhered cotton warp leads to brittleness of the cotton yarn, and so Pareto-efficient vectors of sizing parameters producing an excessive adhesion strength of over 6% are not considered further. Constraints similar to (23) can be imposed on any other sizing quality indicators [27; 28].

The suggested algorithm of controlling sizing quality is consistent, verifiable, and scalable. It does not depend on the number of sizing parameters, nor depends it on the number of quality-controlled factors. Applied to the results of the factually conducted experiments, the algorithm has allowed to increase two of three sizing quality indicators, by acceptably decreasing the third one. Thus, this is a method to improve warp yarn weavability along with improving the economic performance of weaving. This is a practically significant and easy applicable contribution to the theory and real-time practicing of optimal selection of cotton warp sizing parameters, when the number of factual experiments is limited due to material and time resources limitations. Moreover, the algorithm is not limited to cotton, and it can be applied to any yarn material by following formulae (1)–(21) with an experimentally adjusted spread.

CONCLUSION

An algorithm has been suggested to control warp sizing quality under system research limitation, where optimal selection of cotton warp sizing parameters is exemplified. The algorithm has been successfully applied to the results of the factually conducted experiments with the cotton yarn of class 1, yielding the improved quality indicators of the sized warp. The algorithm utilizes a set of basis vectors of sizing parameters that corresponds to a set of respective vectors of quality indicators. Next, the method of radial basis functions is used to determine the probabilistically appropriate vector of quality indicators for any given vector of sizing parameters. Having sampled the uncountably infinite space of sizing vectors, it may then be refined by excluding sizing vectors corresponding to inadmissible values of one or more quality indicators. A set of Pareto-efficient sizing vectors is determined within the finite (refined) space of sizing vectors, and an optimal efficient sizing vector is determined as one being the closest to the best-ever sizing vector, which is usually unachievable.

The suggested algorithm serving as a method of optimal selection of warp sizing parameters under system research limitation depends on both the radial basis function spread and the number of basis vectors of sizing parameters. The latter, however, may have little significance due to only marginal values of the sizing parameters are commonly used. The research is possible to supplement with studying an impact of optimizing the radial basis function spread. While the relationship between the variation of sizing parameter optimum and the radial basis function spread is known to be inverse, an optimized spread may not change much the variation by insignificantly increasing one or a few quality indicators.

REFERENCES

1. K.L. Gandhi, "4 – Yarn preparation for weaving: Sizing," in *K.L. Gandhi (Ed.), The Textile Institute Book Series. Woven Textiles* (Second Edition). Woodhead Publishing, 2020, pp. 119–166. doi: https://doi.org/10.1016/B978-0-08-102497-3.00004-0
2. M.K. Singh, "4 – Yarn sizing," in *M.K. Singh (Ed.), Industrial Practices in Weaving Preparatory*. Woodhead Publishing, 2014, pp. 134–266. doi: https://doi.org/10.1016/B978-93-80308-29-6.50004-9
3. R. Chattopadhyay, S.K. Sinha, and M.L. Regar, "1 – Introduction: textile manufacturing process," in *R. Chattopadhyay, S.K. Sinha, and M.L. Regar (Eds.), The Textile Institute Book Series. Textile Calculation*. Woodhead Publishing, 2023, pp. 1–12. doi: https://doi.org/10.1016/B978-0-323-99041-7.00008-4
4. S.S. Saha, "2 – Basic principles of control systems in textile manufacturing," in *A. Majumdar, A. Das, R. Alagirusamy, and V.K. Kothari (Eds.), Woodhead Publishing Series in Textiles. Process Control in Textile Manufacturing*. Woodhead Publishing, 2013, pp. 14–40. doi: https://doi.org/10.1533/9780857095633.1.14
5. V. Goud, A. Das, and A. Ramasamy, "14 – Yarn testing," in *R. Chattopadhyay, S.K. Sinha, and M.L. Regar (Eds.), The Textile Institute Book Series. Textile Calculation*. Woodhead Publishing, 2023, pp. 325–348. doi: https://doi.org/10.1016/B978-0-323-99041-7.00010-2
6. S.C. Ray, M. Blaga, "5 – Yarns for knitting and their selection," in *S. Maity, S. Rana, P. Pandit, and K. Singha (Eds.), The Textile Institute Book Series. Advanced Knitting Technology*. Woodhead Publishing, 2022, pp. 141–159. doi: https://doi.org/10.1016/B978-0-323-85534-1.00010-6
7. M.S.I. Sarker, I. Bartok, "Global trends of green manufacturing research in the textile industry using bibliometric analysis," *Case Studies in Chemical and Environmental Engineering*, vol. 9, Article ID 100578, 2024. doi: https://doi.org/10.1016/j.cscee.2023.100578

8. F. Puig, A. Debón, S. Cantarero, and H. Marques, "Location, profitability, and international trade liberalization in European textile-clothing firms," *Economic Modelling*, vol. 129, Article ID 106563, 2023. doi: https://doi.org/10.1016/j.econmod.2023.106563
9. W. Fung, M. Hardcastle, "4 – Yarn and fabric processing," in *W. Fung and M. Hardcastle (Eds.), Woodhead Publishing Series in Textiles. Textiles in Automotive Engineering*. Woodhead Publishing, 2001, pp. 110–157. doi: https://doi.org/10.1533/9781855738973.110
10. K. Slater, "5 – Yarn production," in *K. Slater (Ed.), Woodhead Publishing Series in Textiles. Environmental Impact of Textiles*. Woodhead Publishing, 2003, pp. 40–60. doi: https://doi.org/10.1533/9781855738645.40
11. K.M. Mostafa, "Evaluation of nitrogen containing starch and hydrolyzed starch derivatives as a size base materials for cotton yarns," *Carbohydrate Polymers*, vol. 51, issue 1, pp. 63–68, 2003. doi: https://doi.org/10.1016/S0144-8617(02)00106-6
12. A.P.S. Immich, P.H. Hermes de Araújo, L.H. Catalani, S.M.A.G.U. Souza, C.R. Oliveria, and A.A.U. Souza, "Temporary tensile strength for cotton yarn via polymeric coating and crosslinking," *Progress in Organic Coatings*, vol. 159, Article ID 106397, 2021. doi: https://doi.org/10.1016/j.porgcoat.2021.106397
13. W. Li, Z. Zhang, L. Wu, Z. Zhu, and Z. Xu, "Improving the adhesion-to-fibers and film properties of corn starch by starch sulfo-itaconation for a better application in warp sizing," *Polymer Testing*, vol. 98, Article ID 107194, 2021. doi: https://doi.org/10.1016/j.polymertesting.2021.107194
14. A. Tkachuk, "Sizing Compositions for Cotton Warps," in *Abstracts of International Conference "AUTEX 2013", Dresden, Germany, 22–24 May 2013*. Available: https://katalog.slub-dresden.de/id/0-755719387
15. A. Tkachuk, "Adhesive properties of size compositions for cotton warps," in *G. Paraska and J. Kowal (Eds.), Engineering and Methodology of Modern Technology: Monograph*. Khmelnitskiy, 2012, pp. 146–156. Available: https://elar.khmnu.edu.ua:8080/bitstream/ 123456789/1264/1/Tkachuk.pdf
16. P. Ouagne et al., "5 – Use of bast fibres including flax fibres for high challenge technical textile applications. Extraction, preparation and requirements for the manufacturing of composite reinforcement fabrics and for geotextiles," in *R.M. Kozłowski and M. Mackiewicz-Talarczyk (Eds.), The Textile Institute Book Series. Handbook of Natural Fibres* (Second Edition). Woodhead Publishing, 2020, pp. 169–204. doi: https://doi.org/10.1016/B978-0-12-818782-1.00005-5
17. M.D. Buhmann, *Radial Basis Functions: Theory and Implementations*. Cambridge University Press, 2003. doi: https://doi.org/10.1017/CBO9780511543241
18. M.E. Biancolini, *Fast Radial Basis Functions for Engineering Applications*. Springer International Publishing, 2018. doi: https://doi.org/10.1007/978-3-319-75011-8
19. V.V. Romanuke, "A couple of collective utility and minimum payoff parity loss rules for refining Nash equilibria in bimatrix games with asymmetric payoffs," *Visnyk of Kremenchuk National University of Mykhaylo Ostrogradskyy*, issue 1 (108), pp. 38–43, 2018. doi: https://doi.org/10.30929/1995-0519.2018.1.38-43
20. V.V. Romanuke, "Maritime data transmission coverage optimization under power and distance constraints," *Pomorstvo. Scientific Journal of Maritime Research*, vol. 37, pp. 255–270, 2023. doi: https://doi.org/10.31217/p.37.2.8
21. B. Mohebali, A. Tahmassebi, A. Meyer-Baese, and A.H. Gandomi, "Chapter 14 – Probabilistic neural networks: a brief overview of theory, implementation, and application," in *P. Samui, D.T. Bui, S. Chakraborty, and R.C. Deo (Eds.), Handbook of Probabilistic Models*. Butterworth-Heinemann, 2020, pp. 347–367. doi: https://doi.org/10.1016/B978-0-12-816514-0.00014-X
22. V.V. Romanuke, G.A. Yegoshyna, and S.M. Voronoy, "Training probabilistic neural networks on the single class pattern matrix and on concatenation of pattern matrices," *Scientific Papers of O.S. Popov Odessa National Academy of Telecommunications*, no. 2, pp. 86–97, 2019.
23. R. Franke, "Scattered data interpolation: tests of some methods," *Mathematics of Computation*, vol. 38, issue 157, pp. 181–200, 1982. doi: https://doi.org/10.1090/S0025-5718-1982-0637296-4

24. G. Fasshauer, *Meshfree Approximation Methods with MATLAB*. World Scientific Publishing, 2007.
25. F. Pooladi, E. Larsson, "Stabilized interpolation using radial basis functions augmented with selected radial polynomials," *Journal of Computational and Applied Mathematics*, vol. 437, Article ID 115482, 2024. doi: https://doi.org/10.1016/j.cam.2023.115482
26. K. Segeth, "Spherical radial basis function approximation of some physical quantities measured," *Journal of Computational and Applied Mathematics*, vol. 427, Article ID 115128, 2023. doi: https://doi.org/10.1016/j.cam.2023.115128
27. L. Hunter, "12 – Testing cotton yarns and fabrics," in *S. Gordon and Y.-L. Hsieh (Eds.), Woodhead Publishing Series in Textiles. Cotton*. Woodhead Publishing, 2007, pp. 381–424. doi: https://doi.org/10.1533/9781845692483.3.381
28. T. Ahmed et al., "Evaluation of sizing parameters on cotton using the modified sizing agent," *Cleaner Engineering and Technology*, vol. 5, Article ID 100320, 2021. doi: https://doi.org/10.1016/j.clet.2021.100320

Received 25.12.2023

INFORMATION ON THE ARTICLE

Hanna S. Tkachuk, ORCID: 0000-0003-3502-0557, Khmelnytskyi National University, Ukraine, e-mail: tkachukha@khmnu.edu.ua

Vadim V. Romanuke, ORCID: 0000-0001-9638-9572, Vinnytsia Institute of Trade and Economics of State University of Trade and Economics, Ukraine, e-mail: romanukevadimv@gmail.com

Andriy V. Tkachuk, ORCID: 0000-0003-0865-9603, Khmelnytskyi National University, Ukraine, e-mail: tkachukan@khmnu.edu.ua

ОПТИМАЛЬНИЙ ВИБІР ПАРАМЕТРІВ ШЛІХТУВАННЯ БАВОВНЯНОЇ ПРЯЖІ ЗА ОБМЕЖЕНОСТІ СИСТЕМНИХ ДОСЛІДЖЕНЬ / Г.С. Ткачук, В.В. Романюк, А.В. Ткачук

Анотація. Шліхтування основи тканини полягає у нанесенні матеріалів шліхтування на основу пряжі для покращення її властивостей при ткацтві разом з підвищенням економічної ефективності технологічного процесу ткацтва. Розглянуто скінченну множину агентів або параметрів шліхтування, котра відображається у скінченну множину показників якості шліхтування. Оскільки існують різні обмеження на матеріальні та часові ресурси, вичерпне системне дослідження і побудова інформаційної технології для інтерпретації та оптимізації даних шліхтування неможливі. Тому запропоновано алгоритм контролю якості шліхтування основи тканини за обмежень системного дослідження, наведено приклад оптимального вибору параметрів шліхтування бавовняної основи. Алгоритм використовує множину базисних векторів параметрів шліхтування, яку зіставлено з множиною відповідних векторів показників якості. Використано метод радіальних базисних функцій для визначення ймовірнісно прийнятного вектора показників якості для довільного вектора параметрів шліхти. Незліченно нескінченний простір векторів шліхти рівномірно дискретизується у скінченний простір. Цей скінченний простір можна також поліпшити вилученням векторів шліхти, котрі відповідають недопустимим значенням одного або декількох показників якості. У межах даного скінченного (поліпшеного) простору визначається множина Парето-ефективних векторів шліхти, й оптимальний ефективний вектор шліхти визначається як той, який є найближчим до недосяжного вектора шліхти. Запропонований алгоритм слугує методом оптимального відбору параметрів шліхтування основи тканини, результатом застосування якого є покращені властивості основ пряж, що можуть витримувати циклічні тертя, розтягування та згинання на ткацькому верстаті без наслідків ворсування чи іншого псування. Розроблений алгоритм не обмежується використанням бавовни і може бути застосований до довільного матеріалу пряжі за експериментально допасованого значення розтягу радіальної базисної функції.

Ключові слова: шліхтування основи тканини, агенти шліхтування, колоїдні системи, неорганічні складники, показники якості шліхтування, радіальна базисна функція, ефективність за Парето.

UDC 004.89:004
DOI: 10.20535/SRIT.2308-8893.2025.1.08

AGENT-BASED APPROACH TO IMPLEMENTING ARTIFICIAL INTELLIGENCE (AI) IN SERVICE-ORIENTED ARCHITECTURE (SOA)

A.I. PETRENKO

Abstract. Artificial Intelligence (AI) is becoming a general-purpose technology and is gaining a universal character for engineering, science, and society that today is only inherent in mathematics and computer technology. The agent-based approach to implementing artificial intelligence (AI) within the service-oriented architecture of an application is a fascinating and highly synergistic concept. Combining these paradigms leads to robust, scalable, and intelligent systems well suited for dynamic and distributed environments. This paper presents the results of a comparative analysis of three possible approaches to integrating AI into business processes, namely, connecting AI agents to service-oriented architecture (SOA), connecting AI agents to software (SaaS), and building AI as a service (AIaaS). The paper provides some insights into the potential benefits, challenges, examples, and considerations when adopting each of these approaches.

Keywords: AI (Artificial intelligence), agentic AI, AI-agent, SOA (Service oriented architecture), SaaS (Software-as-a-Service), RAG (Retrieval-Augmented Generation), large language models (LLM), single-agent and multi-agent systems, AI agent development platforms, AI agent integration with SaaS, AI agents and SOA.

AGENTIC AI

As artificial intelligence becomes an increasingly integral part of how we live and work, it's important to understand the differences between agent-based AI and generative AI [1; 2; 5; 10; 11; 12; 18].

Generative AI is a type of AI that focuses on creating new content, such as text, images, music, or even video. It works by learning from large amounts of data to understand patterns, styles, or structures, and then generating original content based on what it has learned. For example, generative AI such as *ChatGPT* can generate unique text answers to questions, while image generation models such as *DALL-E* can create images from text descriptions. In essence, generative AI is like a digital artist or writer, creating creative works based on what it has learned.

Agentic AI, on the other hand, is a step forward. Unlike generative AI, it can take initiative, set goals, and learn from its own experience (Fig. 1). It is proactive, able to adjust its actions over time, and can handle more complex tasks that

© *Publisher IASA at the Igor Sikorsky Kyiv Polytechnic Institute*, 2025

require constant problem solving and decision making. This transition from reactive to proactive AI opens up new possibilities for technology in many fields, allowing machines to operate with near-human understanding and creating a seismic technological shift. Machines now understand us better than ever before. They can learn, predict, intuit, and reason. They can take on uncertain tasks, manage complex processes, and make subtle decisions that only a year or two ago could only be made by humans. Imagine a robot that operates without a human controller, determining what to do next based on its environment, or a self-driving car that strives to get you to your destination safely, with every action, from steering to braking, serving that goal.

In short, at its core, agent-based AI is a type of AI that prioritizes autonomy. Agents have true autonomy, making decisions and taking actions independently with minimal human supervision. The level of autonomy is determined by the number of iterations an AI agent can go through to reach a conclusion, as well as the number of tools at its disposal.

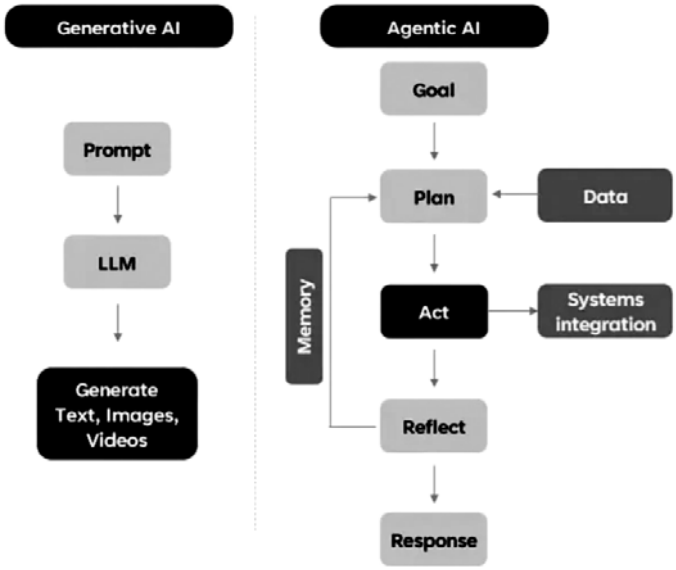


Fig. 1. Difference between Generative AI and Agentic AI [5]

An AI agent is an interactive computer program with pre-defined goals that can perform a variety of tasks on behalf of a user or another program. AI agents have the potential to understand and learn from their environment, make decisions, act, and even continuously improve with minimal human intervention. This is achieved by integrating several key components that allow it to interact with data, interpret its environment, choose appropriate responses, and communicate meaningfully with users. In addition, AI agents can benefit from human feedback, which enhances their adaptability and performance.

In particular, AI agents are used to automate the process of developing web applications without coding. They can analyze user needs, generate code, test applications, and even deploy them. This makes application development faster and more cost-effective. However, the use of AI agents requires the availability of LLMs (large language models such as *GPT-4, Claude, Gemini*, etc.), which can be expensive because LLMs require a lot of resources:

- Significant computational resources for training and operation.
- Huge amounts of data for training.

- Highly qualified machine learning and artificial intelligence specialists to develop and maintain LLMs.
- Significant energy for training and running LLMs.

But there are alternatives and ways to reduce costs:

- Cloud providers such as *Google Cloud, Amazon Web Services,* and *Microsoft Azure* offer access to LLMs on a pay-as-you-go basis.
- For some tasks, smaller SLMs can be used that require fewer computing resources (*Microsoft phi-4, DistilBERT, TinyBERT, Albert*).
- There are open-source LLMs that can be used for free (*Llama 2* and *OPT* from Meta, *GPT-Neo, GPT-J, GPT-3* from EleutherAI). However, they may be inferior in quality to commercial models.

The choice depends on the specific needs of your project. If you need maximum performance and functionality, commercial LLMs may be a better option. If efficiency, speed and resource savings are important, smaller SLMs or open source LLMs may be a better choice. In general, open source LLMs and small SLMs play an important role in the development and dissemination of artificial intelligence technologies, making them more accessible, efficient and versatile.

It should be noted that the boundaries between generative and agent-based AI are not always clear. Many modern AI systems include elements of both, creating hybrid models that can generate content and make autonomous decisions.

AI AGENTS WITH RETRIEVAL-AUGMENTED GENERATION (RAG)

To make an agent contextually relevant, it is connected to an external knowledge base or data source that supports its responses with accurate, domain-specific information. A common approach to such integration is the Retrieval-Augmented Generation (RAG) pattern, which combines external data retrieval with generative capabilities. In addition to basic understanding, the agent is equipped with a toolkit - specialized skills and abilities that allow it to autonomously perform actions, initiate workflows, or solve tasks according to set goals. An orchestrator coordinates all these components and ties the agent's functionality together. The orchestrator processes user input, manages internal operations, and delivers consistent results either directly to the user or to other agents in multi-agent interaction systems.

LLMs are trained on static data sets, which can lead to outdated information. RAG allows agents to access up-to-date information from dynamic sources such as web pages and news feeds (Fig. 2). RAG allows agents to tailor responses to

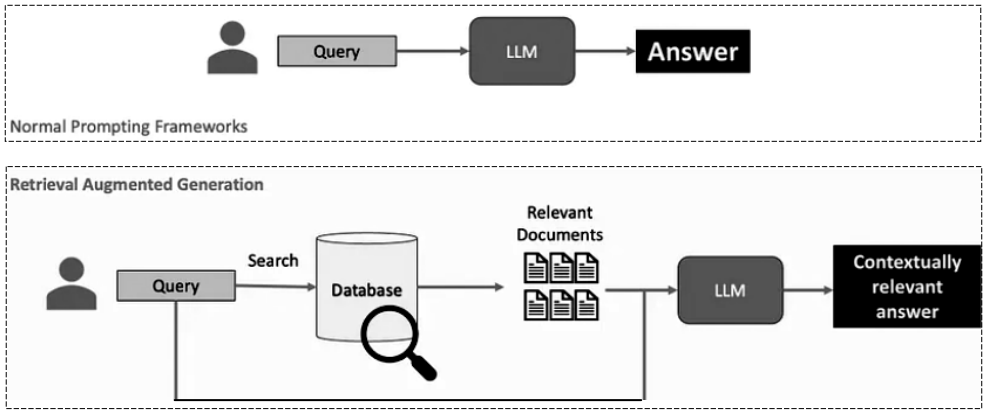


Fig. 2. Comparing standard LLM calls with RAG [9]

the specific context and needs of the user by retrieving relevant information from personalized sources such as search history or user profiles. RAG also enables agents to provide explanations and sources of information, increasing transparency and trust in their responses. RAG works in AI agents as follows [7–9]:

- *Request:* The user asks a question or makes a request to the AI agent.
- *Retrieval:* The agent uses information from the request to search for relevant documents or data in external sources.
- *Extraction:* The agent extracts the most important information from the sources found.
- *Generation:* The agent uses an LLM to generate a response, using the extracted information as context.

Overall, RAG is an important tool for creating more intelligent, accurate and useful AI agents that can effectively interact with the real world and meet user needs. However, an alternative and more modern approach has recently emerged — Table-Augmented Generation (TAG). While RAG has proven effective in integrating AI with external data retrieval systems, TAG offers a paradigm shift by allowing large language models (LLMs) to interact directly with structured databases. Table-Augmented Generation (TAG) provides a more direct and structured approach, allowing LLMs to query databases using SQL or other database-specific query languages [7].

TYPES OF AI AGENTS AND TOOLS FOR CREATING THEM

The following four rules define the functionality of an AI agent: autonomy, perception, decision making and adaptability [3; 6].

- *Autonomy:* This rule means that the AI agent must function independently to perform tasks without constant user intervention.
- *Perception*: The AI agent can interpret data from the environment, obtained through sensors, cameras or other sources.
- *Decision-making*: This involves the AI agent's ability to choose appropriate actions to achieve its goals.
- *Adaptability:* Adaptability is the ability of an AI agent to learn from new information or experience and improve its responses over time.

These principles are the foundation for the design of AI agents and are widely accepted in the AI community to describe the core capabilities that enable intelligent, agent-like behaviour.

Table 1 below provides examples of different types of AI agents. Each type reflects the functionality, adaptability, and level of autonomy of the agent, as well as the specific ways in which AI agents interact with the environment, make decisions, and process information [13; 14; 15; 17].

Since the end of 2024, there has been a significant shift from single-agent AI solutions to multi-agent systems [16; 18].

- *Single-agent systems:* These are focused artificial intelligence models geared towards specific tasks, such as smart chatbots. While effective in isolated scenarios, they have limitations in managing complex, interconnected workflows. Single-agent systems typically require human involvement to provide ongoing feedback.

- *Multi-agent systems:* These involve a network of AI agents that collaborate to solve problems or achieve goals that require diverse knowledge. Imagine a team collaborating, communicating internally, critiquing each other, and improving each other's results to solve a given task, as opposed to a single agent receiving feedback only from the human interacting with it.

Table 1. Types of AI Agents

№	Agent Type	Description	Examples
1	Simple Reflex Agent	These agents act solely based on the current state of the environment, without considering the history or planning for the future. They follow a set of predefined rules (condition-action rules), e.g., "if sensor detects X, then perform Y"	Thermostats, basic robots, or systems like spam filters that react to input based on simple criteria
2	Model-Based Reflex Agent	These agents maintain a model of the world that helps them keep track of the state of the environment. They use the model to make better decisions by considering both the current state and the history of interactions	Robotic vacuum cleaners (like Roombas) that map their environment to navigate efficiently
3	Goal-Based Agent	These agents not only track the state of the environment but also act to achieve specific goals. They use planning and search algorithms to determine the best course of action to achieve their objectives	Autonomous vehicles deciding the optimal route to a destination
4	Utility-Based Agent	These agents evaluate different states or actions based on a utility function, which measures how desirable a state is. They aim to maximize the utility (or "happiness") by choosing actions that lead to the best possible outcome	AI in recommendation systems (e.g., Netflix or Spotify), where the goal is to maximize user satisfaction
5	Learning Agent	These agents improve their performance over time by learning from data or experiences. They can adapt to new tasks or environments without explicit programming	Chatbots (like ChatGPT), recommendation systems, autonomous vehicles, and game-playing AIs (like AlphaGo)
6	Collaborative Agent	Works with other agents or humans	Coordinating the work of multiple robots
7	Mobile Agent	Moves between networks to perform tasks	Network management scenario
8	Multi-Agent System	Involves multiple agents working together or competing to solve complex problems that cannot be handled by a single agent. Agents in a MAS can communicate, negotiate, and collaborate to achieve shared or individual goals	Traffic management systems, distributed supply chain systems, and swarm robotics
9	Belief-Desire-Intention (BDI) Agent	Balances between beliefs, desires, and intentions	Autonomous bots for customer service
10	Interface Agent	Helps users by learning preferences	Personalized email sorting
11	Reactive Agent	Reacts quickly without internal models	Real-time game characters
12	Hybrid Agents	These agents combine multiple types of agent architectures (e.g., reflex and goal-based) to leverage the strengths of each	Autonomous drones that use reflex actions for obstacle avoidance and goal-based planning for navigation

AI Agent Development Platforms [16; 19; 20] are specialized tools that streamline the process of building, training, and deploying AI agents. These platforms aim to abstract away much of the complexity involved in AI development, allowing developers and even non-developers to create AI agents more efficiently. They simplify AI Agent Development by:

- *Rapid Prototyping*: Developers can quickly build prototypes without deep dives into complex algorithms or infrastructure setup.
- *Reduced Technical Debt:* By using standardized platforms, there's less custom code to maintain, reducing long-term technical debt.
- *Access to Cutting-edge Technology*: Platforms often incorporate the latest AI advancements, making them available to users without the need for extensive research or development.
- *Focus on Business Logic:* Developers can focus more on defining the business logic of the AI agent rather than the underlying technology.

Some such platforms are listed in Table 2 although there are many other platforms (*H2O.ai, DataRobot, DOMO, etc.)*.

T a b l e 2 . Examples of applications of Agent-Based AI in SOA

№	**Platform name**	**Description of features**
1	Google Vertex AI Agent Builder	This platform integrates Google's foundation models, search functionalities, and conversational AI technologies into a unified development environment. Utilising Vertex AI Agent Builder, developers are empowered to construct AI agents through a no-code interface or by employing more sophisticated frameworks
2	Microsoft Azure Autonomous Systems Platform	A plethora of instruments and facilities are available for the development of artificial intelligence agents, encompassing Azure Bot Service, Azure Cognitive Services and Azure Machine Learning
3	Amazon SageMaker Agents	AWS offers a variety of services for developing AI agents, such as Amazon Lex, Amazon Polly, and Amazon Rekognition
4	Hugging Face Transformers	An open-source platform that provides free access to a large number of pre-trained AI models and tools for developing AI agents
5	Microsoft AutoGen	Microsoft's multi-agent conversational platform is designed to facilitate the development of Large Language Model (LLM) workflows, with the objective of enabling the utilisation of diverse applications across multiple industry sectors. In addition, AutoGen provides "AutoGen Studio," a tool that facilitates the creation of multi-agent systems without the requirement for extensive coding
6	LangChain	This is an open-source platform for AI agents that features a low-code drag-and-drop interface. It is available for download at no cost on GitHub, but a paid OpenAI API key is required for operation
7	AgentGPT	A web platform for deploying AI agents directly from your web browser using GPT 3.5. Supports tasks such as AI image creation, Google search, and code writing
8	Salesforce Einstein Agent Builder	A platform for deploying AI agents from Salesforce, designed to automate complex tasks. Includes Agent Builder for creating AI agents with natural language instructions
9	Godmode	Web solution for AI agents powered by OpenAI and Microsoft. Supports the creation of multiple AI agents simultaneously using GPT-3.5 and GPT-4 based on user-defined goals
10	OpenAI Agents	With GPT-4o, o1, and subsequent versions, developers can create increasingly complex agents and deploy them in their applications or on the ChatGPT platform

Continued Table 2

11	Fetch.ai	This provides cryptographic and blockchain capabilities to AI agents, thus allowing various types of agents (i.e. reflex agents, goal agents and utility agents) to access blockchain features such as crypto wallets and on-chain interaction
12	LlamaIndex	A popular framework for developing AI agents, which provides tools for working with large language models, such as GPT-3
13	CrewAI	CrewAI is an open-source framework in Python designed to support the development and management of multi-agent artificial intelligence systems. The enhancement of these AI systems is achieved by the allocation of specific roles, the enablement of autonomous decision-making, and the facilitation of communication between agents. This collaborative approach enables the effective resolution of complex problems, surpassing the capabilities of individual agents operating in isolation
14	PhiData	PhiData provides a comprehensive framework for the creation of complex agents that possess enhanced memory and knowledge management capabilities, utilising GPT-4 technology. This platform is particularly well-suited for applications that necessitate profound contextual understanding and long-term learning capabilities. Potential use cases include long-term projects, knowledge-intensive tasks and personalised user interaction
15	Atomic Agents	Atomic Agent is a versatile, open-source framework created by BrainBlend AI designed for developing multi-agent systems and AI applications. It emphasizes modularity and atomicity, allowing developers to construct complex AI solutions by combining simple, interchangeable components. By breaking down AI systems into smaller, self-contained, reusable components, Atomic Agents promises a future where AI development is both modular and predictable

Key Benefits of AI Agent Development Platforms are:

Abstraction of Complexity:

- *No-Code/Low-Code Interfaces*: Platforms like Microsoft Azure AI, and Google Cloud AI offer drag-and-drop interfaces or visual programming tools, reducing the need for extensive coding.
- *Pre-built Components:* Many platforms provide pre-built AI models, templates, or modules for common tasks like NLP, image recognition, etc., which can be easily integrated.

Integration and Deployment:

- *Seamless Integration:* These platforms often come with built-in tools for integrating AI agents with other systems or services, using APIs or direct connectors.
- *Deployment Automation:* They handle the deployment process, including scaling, which can be particularly useful for cloud-based solutions.

Data Management:

- *Data Pipelines*: Platforms like DataRobot or H2O.ai offer tools to manage data flows, from ingestion to preprocessing, making it easier to feed data into AI models.
- *Model Training:* Automated or semi-automated model training features, which can optimize hyperparameters and select the best model architecture.

User-Friendly Interfaces:

- *Dashboarding:* Visual dashboards for monitoring model performance, data quality, and agent interactions.
- *Collaboration:* Features for team collaboration, version control, and sharing of AI assets.

Scalability and Performance:

- *Cloud Resources:* Leveraging cloud infrastructure for scalability, which is crucial for AI applications that might require significant computational resources.

Security and Compliance:

- *Built-in Security Measures:* Many platforms include security protocols to protect data and models.
- *Compliance*: Some platforms help in adhering to industry standards and regulations, which is vital for deploying AI in regulated sectors.

In conclusion, AI agent development platforms significantly lower the entry barrier for building AI applications. They democratize AI technology by making it accessible to a broader audience, including those without deep technical expertise in AI. However, for highly specialized or performance-critical applications, a more custom approach might still be required. Many platforms operate on a subscription or usage-based model, which can be costly for large-scale or long-term projects. There's also a risk of becoming dependent on the platform's ecosystem, which might complicate migration or integration with other systems.

CONNECTING AI AGENTS TO SOA

The agent-based approach to implementing artificial intelligence (AI) within a service-oriented architecture (SOA) is a fascinating and highly synergistic concept. Remind, that SOA is an architectural style where software components are designed as independent, reusable services that communicate with each other over a network. The combination of agent-based AI and SOA brings together the best of both worlds—autonomous decision-making from agents and the modular, distributed nature of SOA [20; 21].

Potential Benefits of Connecting AI Agents to SOA:

- **Enhanced Intelligence in Services:** AI agents can bring intelligence and autonomy to SOA by enabling services to dynamically adapt, learn, and make intelligent decisions based on real-time data.
- **Improved Automation:** AI agents can automate complex tasks and workflows within an SOA, leading to increased efficiency and reduced human intervention.
- **Personalized Experiences:** AI agents can analyze user data and preferences to personalize the delivery of services within an SOA, creating more tailored and engaging experiences.
- **Dynamic Optimization:** AI agents can continuously monitor and optimize the performance of services within an SOA, ensuring optimal resource utilization and responsiveness.
- **Dynamic and Adaptive Systems:** Agents can adapt to changes in the environment or user demands in real-time. When embedded in SOA, this adaptability allows services to be reconfigured dynamically based on the context, improving system responsiveness and robustness.
- **Decentralization**: SOA is inherently decentralized, and the agent-based approach aligns well with this philosophy. Each agent can act independently while still interacting with other services, reducing bottlenecks and single points of failure.

- **Scalability and Modularity:** SOA is designed for modularity, and adding intelligent agents to individual services allows for a scalable way to introduce AI capabilities. New agents can be introduced or updated without disrupting the entire system.
- **Interoperability:** Agents can act as intermediaries or orchestrators between services in SOA, enabling better integration of heterogeneous systems or legacy services.
- **Enhanced Decision-Making:** Agents bring reasoning and decision-making capabilities to SOA, enabling services to not just respond to requests but also predict, optimize, and proactively act to improve outcomes.
- **Support for Complex, Multi-Agent Systems**: In cases where multiple agents are deployed (e.g., for supply chain management or IoT systems), SOA provides a framework for communication and collaboration among agents, ensuring interoperability and coordination.

Connecting AI agents to a Service-Oriented Architecture (SOA) involves **several technologies and methodologies** to ensure seamless integration, interoperability, and effective communication between AI components and other services. Here's a breakdown of the key technologies and approaches:

API Integration

- *RESTful APIs:* AI agents can expose their functionalities through RESTful services, allowing them to be consumed by other services within the SOA. This method is stateless, making it scalable and easy to integrate.
- *GraphQL:* For more flexible data fetching, GraphQL can be used where clients can request exactly what data they need from the AI agent, reducing over-fetching and under-fetching.

Messaging Systems

- *Message Brokers (e.g., RabbitMQ, Apache Kafka):* These facilitate asynchronous communication. AI agents can publish results or receive tasks through messages, which is particularly useful for handling high volumes of data or when real-time processing isn't necessary.
- *Event-Driven Architecture (EDA):* AI agents can react to events triggered by other services, allowing for dynamic, responsive systems where AI capabilities are invoked based on specific business events.

Microservices Architecture

- *Containerization (Docker, Kubernetes):* AI services can be containerized, making them portable and scalable. This approach fits well with microservices where each AI function might be its own microservice.
- *Service Mesh (e.g., Istio):* Enhances how services communicate, manage, and secure inter-service communication, which is crucial when integrating AI agents that might need specific network policies or security measures.

Data Handling and Integration

- *Data APIs:* For AI agents that require or produce data, data APIs can be used to integrate with data services or databases within the SOA.
- *Data Streaming Technologies:* Tools like Apache Kafka or AWS Kinesis can be used for real-time data streaming to and from AI agents, ensuring they have the latest data for processing.

Orchestration and Workflow Management

- *Workflow Engines (e.g., Camunda, Apache Airflow):* These can orchestrate complex workflows where AI agents are just one part of a larger process, ensuring that AI tasks are executed in the right sequence and context.

- *Serverless Computing:* Using platforms like AWS Lambda or Azure Functions, AI tasks can be executed in response to events without managing the underlying infrastructure.

Security and Governance

- *OAuth, JWT (JSON Web Tokens):* For secure communication and authentication between services.
- *API Gateways:* To manage access to AI services, ensuring that only authorized services can interact with AI agents.

Monitoring and Management

- *Service Monitoring Tools (e.g., Prometheus, Grafana):* To monitor the health, performance, and usage of AI services within the SOA.
- *Centralized Logging:* Tools like ELK stack (Elasticsearch, Logstash, Kibana) for logging and analyzing interactions and performance of AI agents.

AI-Specific Technologies

- *Model Serving Platforms (e.g., TensorFlow Serving, Seldon Core*)**:** These platforms allow for the deployment of machine learning models as services, making it easier to integrate AI models into SOA.
- *Feature Stores:* Centralized repositories for managing and serving features used by machine learning models, ensuring consistency and reducing data duplication.

Integration Patterns

- *Proxy Pattern:* AI agents can act as proxies or facades, simplifying the interface for other services.
- *Adapter Pattern:* Used to convert the interface of an AI agent into another interface clients expect, improving reusability.

The technology for connecting AI agents to SOA involves a blend of modern software architecture practices, cloud technologies, and AI-specific tools. The goal is to create a flexible, scalable, and maintainable system where AI capabilities are seamlessly integrated into business processes, enhancing overall system intelligence without disrupting existing services. This integration requires careful planning, especially around data flows, security, and performance, to ensure that the AI components work harmoniously within the broader service ecosystem.

While the agent-based AI approach within SOA is powerful, there are ***some challenges*** and factors you need to consider:

- ***Complexity*****:** Introducing agents into SOA can increase system complexity, especially when managing interactions between autonomous agents and services.
- ***Communication Overhead*****:** Agents and services need to communicate frequently, which can introduce latency or bottlenecks in distributed systems if not carefully designed.
- ***Security*****:** Autonomous agents might make unauthorized decisions or interact with malicious services. Ensuring secure communication and decision-making is critical.
- ***Standardization*****:** SOA relies on standard protocols (e.g., SOAP, REST), while agents may require additional protocols for negotiation, collaboration, or reasoning. Aligning these standards can be challenging.
- ***Scalability of Decision-Making*****:** As the number of agents and services grows, ensuring that agents can make decisions in a timely manner without overwhelming the system is essential.

- ***Interoperability of AI Models*:** Agents might use different AI models, which could lead to compatibility issues. A common framework or ontology may be needed for agents to collaborate effectively.
- ***Monitoring and Debugging*:** Debugging agent behaviors in a distributed SOA environment can be complex, especially when agents are making autonomous decisions based on incomplete or uncertain information.

It should be emphasized that this approach differs significantly from previous developments, where a software agent-based service-oriented integration architecture for collaborative intelligent systems was proposed. A unique feature of mentioned approach was that the order planning process was organized online through negotiations between agent-based web services. Of course, the software agents used were not present AI agents [22].

Agent-based AI within a Service-Oriented Architecture (SOA) is a powerful combination. Examples of applications of Agent-Based AI in SOA are presented in Table 3.

Table 3. Examples of applications of Agent-Based AI in SOA

№	Sectors	Use cases	Agents' role
1	Resource Management and Optimization	Smart Grids	AI agents represent energy producers (solar panels, wind turbines), consumers (homes, businesses), and storage units. They interact and negotiate in real-time to balance energy supply and demand, optimize grid stability, and reduce costs
		Supply Chain Logistics:	Agents model suppliers, manufacturers, distributors, and customers. They autonomously manage inventory, predict disruptions, and optimize delivery routes to improve efficiency and responsiveness
2	Personalized Customer Service	E-commerce	AI agents act as virtual shopping assistants, learning customer preferences and providing personalized product recommendations, deals, and support, enhancing the shopping experience
		Financial Services	Agents offer personalized financial advice, analyze market trends, and manage investment portfolios based on individual client goals and risk tolerance
3	Complex System Modeling and Simulation	Healthcare	Agents simulate patients, doctors, hospitals, and other healthcare providers to model disease spread, evaluate treatment strategies, and optimize healthcare resource allocation
		Traffic Management	Agents represent vehicles, pedestrians, and traffic signals. They interact to optimize traffic flow, reduce congestion, and improve road safety
4	Autonomous Systems	Robotics	Agents control robots in manufacturing, warehouse automation, and exploration. They can adapt to changing environments, collaborate with other robots, and learn from experience
		Self-Driving Cars	Agents perceive the environment, make driving decisions, and coordinate with other vehicles to ensure safe and efficient navigation

Examples of Real-World Applications:

- **IBM Watson:** Used in healthcare to provide personalized cancer treatment recommendations.
- **Amazon Alexa:** Employs AI agents for natural language understanding and task automation.

- **Tesla Autopilot:** Utilizes agent-based AI for autonomous driving features.

Agent-based AI in SOA is transforming how we design and build intelligent systems. By combining the strengths of both approaches, we can create more flexible, scalable, and responsive solutions to complex real-world problems.

CONNECTING AI AGENTS TO SAAS

Applying an AI agent to a Software as a Service (SaaS) platform can indeed be highly beneficial, offering numerous advantages that can enhance the value proposition of the SaaS product. It is important to recall the definition of SaaS, which is a software delivery service in which a provider hosts a software service and makes it available to customers over the Internet. Customers can access the software through a web browser, eliminating the need to purchase, install and maintain software on their own servers. The SaaS provider assumes responsibility for a wide range of tasks, including maintaining servers and databases, providing updates and implementing security measures. There are a number of platforms for creating sophisticated web applications without writing code [23]:

- Customer Relationship Management (CRM): Salesforce, HubSpot, Zoho Creator.
- Enterprise Resource Planning (ERP): SAP, Oracle NetSuite.
- Project Management: Trello, Asana, Webflow.
- Document Collaboration: Google Workspace, Microsoft 365.
- E-commerce Platforms: Shopify, Magento, Airtable.
- Marketing Automation: Mailchimp, Marketo, Bubble.
- Human Resources: BambooHR, Workday.

Key Characteristics of SaaS Platforms:

- *Cloud-Based Delivery:* SaaS applications are hosted on servers in the cloud, accessible via web browsers or lightweight client applications. This eliminates the need for users to install software on their local machines.
- *Subscription-Based Pricing:* Users typically pay for SaaS on a subscription basis, which can be monthly, annually, or based on usage. This model contrasts with traditional software where you buy a license outright.
- *Multi-Tenancy:* The software is designed to serve multiple customers (tenants) from a single instance of the application. Each customer's data is isolated and secure, but the codebase and infrastructure are shared.
- *Scalability:* SaaS platforms are built to scale easily, allowing them to accommodate growth in users, data, or functionality without significant additional setup or cost.
- *Automatic Updates:* Updates, including new features, security patches, and bug fixes, are automatically rolled out to all users, ensuring everyone has the latest version without manual updates.
- *Data Management:* The provider manages data storage, backup, and recovery, which includes handling data security and compliance with relevant regulations.
- *Accessibility:* Users can access the software from any device with an internet connection, often requiring only a web browser, which enhances mobility and remote work capabilities.

Benefits of SaaS Platforms:

- *Cost Efficiency*: Reduces the need for capital expenditure on hardware and software licenses.

- *Ease of Use:* Generally easier to deploy and use compared to on-premises software.
- *Flexibility:* Allows for flexible scaling of resources according to business needs.
- *Innovation:* SaaS providers often update their services with new features, keeping the software current with market trends.

Challenges:

- *Dependency on Internet:* Requires a reliable internet connection for access.
- *Data Security:* Concerns about data privacy and security since data is stored on external servers.
- *Customization*: Some SaaS solutions might not offer the level of customization that on-premises software can provide.
- *Vendor Lock-in:* Potential for dependency on the SaaS provider, making it difficult to switch.

Key Benefits and Potential of AI-SaaS Integration:

- **Enhanced Automation:** AI agents automate routine business processes in SaaS applications, minimizing human intervention and can automate routine tasks, customer inquiries, or even complex processes like data analysis, freeing up human resources for more strategic tasks.
- **Improved Customer Experience:** AI agents use SaaS platforms to deliver fast and personalized services to customers and to tailor the user interface, content, and recommendations based on user behavior and preferences, making the service more intuitive and engaging.
- **Data Analytics and Insights:** AI agents analyze data within SaaS applications to deliver actionable insights and to automate routine tasks, customer inquiries, or even complex processes like data analysis, freeing up human resources for more strategic tasks.
- **Collaboration Efficiency:** AI optimizes collaboration processes within SaaS applications and optimize the use of computational resources, ensuring the SaaS platform scales efficiently with demand.
- **Cost and Time Savings:** AI-SaaS integration reduces operational costs by minimizing the need for manual labour and saves time**.**
- **Security and Compliance:** AI agents enhance data security and compliance within SaaS applications by identifying unusual patterns that might indicate a security breach or unauthorized access.
- **Innovation and Competitive Edge:** AI agents can be used to develop new features or enhance existing ones, providing a competitive edge by offering capabilities that are difficult for competitors to replicate quickly.

Market Differentiation: A SaaS with integrated AI can differentiate itself in a crowded market by offering smart, proactive services.

- ***Customer Engagement*:** Chatbots and Virtual Assistants can provide 24/7 customer support, handle FAQs, and guide users through the platform, improving customer satisfaction and engagement.

AI agents integrated with SaaS (Software as a Service) applications offer a wide range of opportunities for businesses, including process optimization, customer satisfaction, cost savings, and strategic decision support. When integrating AI into a SaaS platform, the AI functionalities are often provided as part of the service, enhancing the platform's capabilities (Table 4).

Table 4. Examples of applications of Agent-Based AI in SaaS

№	Sectors	Use cases	Agents' role
1	Personalized User Experience	Learning Platforms	AI agents act as virtual tutors, adapting to individual learning styles, recommending relevant content, and providing personalized feedback to optimize learning outcomes
		Content Streaming Service	Agents analyze user preferences and viewing history to suggest personalized recommendations, discover new content, and create custom playlists
2	Intelligent Automation	CRM Systems	AI agents automate repetitive tasks like data entry, lead qualification, and customer segmentation, freeing up human agents to focus on more strategic activities
		Project Management Software	Agents can monitor project progress, identify potential risks, and automatically assign tasks to team members based on their skills and availability
3	Proactive Customer Support	Help Desk Software	AI agents provide instant answers to common questions, troubleshoot issues, and escalate complex problems to human agents, improving response times and customer satisfaction
		Chatbots	Agents engage in natural language conversations with customers, providing support, answering questions, and guiding them through processes
4	Data Analysis and Insights	Marketing Automation Platforms	Agents analyze customer data to identify patterns, predict behavior, and personalize marketing campaigns for better engagement and conversion rates
		Business Intelligence Tools	Agents can sift through large datasets, identify trends, and generate reports to provide valuable insights for decision-making
5	Enhanced Security	Cybersecurity Platforms	AI agents monitor network traffic, detect anomalies, and respond to security threats in real-time, protecting sensitive data and systems
		Identity and Access Management	Agents can analyze user behavior, identify suspicious activities, and prevent unauthorized access to critical resources
6	Automated Task Management	AgentForce	AgentForce is Salesforce's innovative AI agent designed to deliver speed, efficiency, and personalized solutions in areas like customer service, sales, and marketing. Its core strength lies in placing artificial intelligence at the heart of business processes, helping companies become smarter and more competitive

Examples in Action:

- **Salesforce Einstein:** Uses AI agents to provide sales predictions, automate tasks, and personalize customer interactions.
- **HubSpot:** Leverages AI for content optimization, lead scoring, and chatbot interactions.
- **Grammarly:** Employs AI agents to provide grammar and writing suggestions.

By incorporating agent-based AI into SaaS applications, businesses can unlock new levels of efficiency, personalization, and intelligence, ultimately delivering greater value to their customers. In summary, a SaaS platform delivers software applications over the internet, managed by third-party providers, offering users a convenient, scalable, and often cost-effective way to access software. The integration of AI into these platforms leverages cloud computing's scalability and data processing capabilities to provide advanced, data-driven features that can significantly enhance the functionality and user experience of the software.

BUILDING AI AS A SERVICE (AIAAS)

With AIaaS, businesses of all sizes can access natural language processing (NLP), machine learning (ML) algorithms, predictive analytics, and more to automate tasks, analyze data, or improve business strategies and customer experience. They can use and benefit from these AI tools, even without a large team of developers or a huge budget, making it a lower-risk way to integrate AI into their business. As a cloud computing service, AIaaS is flexible and can easily scale as its needs grow without updating your hardware or infrastructure [24; 25].

Advantages:

- *Cost-effectiveness:* Businesses can access AI capabilities without the need for significant upfront investment in hardware and software.
- *Scalability and Flexibility*: AIaaS solutions can be easily scaled up or down to meet changing business needs.
- *Faster Deployment:* Pre-built AI agents can be quickly integrated into existing systems and workflows.
- *Access to Expertise:* AIaaS providers offer expertise and support to help businesses effectively leverage AI technology.
- *Business Model Innovation:* AIaaS can open new revenue streams by offering AI capabilities as a subscription or pay-per-use model, potentially attracting a broader customer base.
- *Scalability:* Easier to scale AI services as they are managed centrally by the service provider, who can optimize resources across multiple clients.
- *Focus on Core Business:* Allows companies to focus on their core competencies while outsourcing AI development and maintenance.

Challenges:

- *Dependency:* Clients become dependent on the service provider for AI capabilities, which could pose risks if the provider faces issues or if there are changes in service terms.
- *Customization:* Generic AI services might not fully meet the specific needs of every business, potentially requiring additional customization which could negate some cost benefits.
- *Data Privacy and Security*: Outsourcing AI means sensitive data might be processed outside the organization, raising concerns about data privacy and compliance with regulations like GDPR or CCPA.
- *Market Saturation:* The AIaaS market could become saturated, leading to price wars and reduced profitability.

The “AI Agent as a Service” model involves selecting of an AI agent type and tools for AI agent Development [25; 27]. The type of AI agent which is building depends on the complexity of the task and the environment in which it will operate. From simple reflex agents to learning agents and multi-agent systems, each type has strengths and applications. Similarly, the tools and frameworks which are choosing will depend on project requirements, such as scalability, ease of use, and the specific domain (e.g., robotics, gaming, or conversational AI). By combining the right type of agent with suitable tools, it is possible to create powerful AI systems tailored to your needs. Some existing use cases where this model has been implemented: are summarizing in Table 5.

Table 5. Examples of use cases for which AIaaS was developed

№	Sectors	AIaaS	Agents' role
1	Specialized AI Assistants	Customer Service Agents	Companies like Ada and Intercom offer AI-powered chatbots that can be integrated into websites and apps to handle customer inquiries, provide support, and even process transactions
		Sales and Marketing Agents	Tools like Drift and Conversica provide AI agents that can qualify leads, schedule appointments, and nurture prospects through personalized email campaigns
		HR and Recruiting Agents	Services like Ideal and Pymetrics use AI agents to screen resumes, conduct initial interviews, and match candidates with the best-fit jobs
2	AI-Powered Automation Platforms	UiPath and Automation Anywhere	These platforms offer AI agents that can automate repetitive tasks, such as data entry, invoice processing, and report generation, across various business applications
		Zapier and IFTTT	These services use AI agents to connect different apps and automate workflows, such as sending notifications, creating tasks, and updating spreadsheets
3	Healthcare Data Analysis	Google Cloud Healthcare API with AI capabilities	This AI agent allows healthcare providers to analyze patient data for better treatment outcomes, operational efficiency, and compliance with healthcare regulations without needing to develop AI models in-house
4	Content Generation and Editing	Grammarly for Business	This AI agent not only checks for grammar and spelling but also provides suggestions for clarity, tone, and engagement, ensuring high-quality content production at scale
5	Personalized Learning	Knewton's Alta for adaptive learning	This AI agent personalizes the learning experience for each student, adapting content and difficulty based on individual performance, thereby improving learning outcomes
6	Energy Sector Optimization	Siemens' MindSphere	This AI agent helps in scheduling maintenance efficiently, reducing energy costs, and extending the lifespan of infrastructure
7	Real-time Traffic Management	INRIX Traffic AI	This AI agent helps in reducing traffic jams, improving emergency response times, and enhancing overall city mobility

These examples demonstrate how "AI Agent as a Service" can be applied across various industries to provide specialized AI functionalities without the need for clients to invest heavily in AI infrastructure or expertise. This model allows businesses to leverage cutting-edge AI technologies for specific tasks, enhancing their operations, customer service, and decision-making processes.

COMPARISON OF DIFFERENT WAYS OF AI AGENTS' USAGE

There are three basic approaches: connecting AI agents to SOA, connecting AI agents to SaaS and building a new business model "AI as a Service". Which is better?

Key Differences:

- **SOA:** Emphasizes modularity, reusability, and interoperability across different systems and applications within an organization. It's ideal for complex enterprise-level solutions where integration and flexibility are crucial.
- **SaaS:** Focuses on enhancing specific applications with AI capabilities to improve user experience, automate tasks, and provide intelligent insights. It's often delivered as part of a cloud-based software subscription.

- **AIaaS:** Provides access to pre-built AI agents or tools for specific tasks, such as natural language processing, image recognition, or data analysis. It allows businesses to leverage AI without the need for extensive development or infrastructure.

The choice depends on the organization's strategic goals, existing technological landscape, risk tolerance, and market positioning [26] The detailed analysis of important factors, including features of integration, scalability, customization, maintenance, deployment and cost, are summarized in Table 6.

Table 6. Comparison of different ways of AI agents' usage

Feature	Agent-Based AI in SOA	Agent-Based AI in SOA	AI Agents as a Service
Architecture	Decentralized, service-oriented	Centralized, application-specific	Typically centralized, API-driven
Integration Method	AI agents are integrated as services within an existing or new SOA framework	AI agents are integrated into or alongside SaaS applications, often through APIs or as plugins	AI capabilities are offered as standalone services, accessible via APIs, which can be integrated into any platform or application
Deployment	On-premise or cloud	Cloud-based	Cloud-based
Focus	Complex system integration, enterprise-level solutions	Specific application functionality, user experience	Specialized AI tasks, pre-built agents
Customization	High	Moderate	Limited, but increasing with API options
Scalability	Highly scalable through adding/removing agents	Agent-Based AI in SaaS	Highly scalable, managed by the provider
Data Handling	Full control over data flow and processing	Less control over data, what simplifies data management	Users have control over what data they send to the AI service, but the data processing happens on the provider's infrastructure
Maintenance	Can be complex due to the need to maintain both AI services and the underlying SOA infrastructure	Lower maintenance burden on the user side as SaaS providers handle much of the backend maintenance	Low for users, as the service provider manages the AI infrastructure and updates
Cost	Higher upfront investment	Subscription-based, predictable costs	Pay-as-you-go, variable costs
Examples	Smart grids, supply chain optimization, healthcare systems	Personalized learning platforms, CRM systems, help desk soft	Chatbots, virtual assistants, automation tools

Choosing the Right Approach:

The best approach depends on your specific needs and goals:

- ***SOA:*** Best for large organizations with complex systems and integration requirements.
- ***SaaS:*** Suitable for businesses seeking AI-powered features within specific applications.
- ***AIaaS***: Ideal for those wanting to quickly integrate AI capabilities into their existing systems or build custom AI solutions without heavy investment.

Comparison Summary:

- *Integration Complexity*: **SOA > SaaS > AIaaS (AIaaS** being the simplest for external integration).

- *Control Over Data and Infrastructure:* **SOA > SaaS > AIaaS** (SOA offers the most control).
- *Scalability and Flexibility:* **AIaaS > SaaS > SOA** (AIaaS and SaaS excel in cloud-native environments).

Ultimately, these approaches are not mutually exclusive. They can be combined and integrated to create comprehensive AI solutions that address a wide range of business challenges.

CONCLUSIONS

Businesses are increasingly leveraging AI to enhance their operations and customer experiences. There are three primary ways to integrate AI:

AI with SOA: This involves incorporating AI agents into existing Service-Oriented Architectures (SOA). SOAs are built on interconnected services, and adding AI can automate tasks, analyze data within these services, and improve overall system efficiency. This approach is useful for businesses with established SOAs looking to enhance functionality.

AI with SaaS: This integrates AI agents into Software as a Service application. This can personalize user experiences, automate tasks within the application, and provide valuable insights from user data. This approach is beneficial for businesses utilizing SaaS solutions and wanting to improve their capabilities.

AI as a Service: This is a business model where AI capabilities are offered as a standalone service. Companies can access sophisticated AI tools without investing heavily in infrastructure or expertise. This is ideal for businesses wanting to experiment with AI or needing specific AI functions without building them from scratch.

Each approach has its own merits and the best choice depends on a business's specific needs, existing infrastructure, and AI goals. However, all three approaches have the potential to help businesses improve their operations and gain a competitive advantage. Careful attention to design, communication, security, and scalability is required to fully realize the benefits of these approaches.

AI agents powered by advanced generative AI (GenAI) technologies will be the most disruptive force in technology in 2025 [29]. These autonomous systems, capable of performing complex tasks with minimal human intervention, are poised to revolutionize industries, rethink workflows, and increase productivity. Using the results of recent research, including Deloitte's 2025 Forecast Report [30], we can predict the emergence, application, and future of AI agents in most industries. Retrieval-augmented generation (RAG) extends the ability of AI agents to work with unstructured data. Such agents can quickly find the information they need and generate accurate answers, making them invaluable for knowledge management and improving the efficiency of managing interdependent workflows. AI agent architectures are becoming more modular, allowing for greater customization and scalability. This evolution ensures that organizations can deploy solutions tailored to their specific needs without significant AI agent overhauls.

The surge in popularity of AI agents in late 2024 mirrors how ChatGPT and other LLMs transformed the AI market in 2022. Now, vendors and developers are massively shifting from creating cutting-edge LLMs and AI chatbots to developing AI agents and exploring ways to implement them. Recently, a new approach to language modelling (called Large Concept Models or LCMs) has been announced that operates at a higher semantic level, dealing with concepts that often correspond to a sentence in text or an equivalent speech utterance [31].

It is predicted that starting in 2025, organizations that embrace this transformation will thrive, while those that cling to traditional SOA and SaaS paradigms will struggle to remain relevant. Ethical considerations are at the forefront of AI agent development. In 2025, the focus will be on explainability so that users can understand and trust the decisions made by AI agents.

REFERENCES

1. *SOAIS. Agentic AI: The Future of Autonomous, Action-Driven Automation.* Available: https://soais.com/agentic-ai-the-future-of-autonomous-action-driven-automation/
2. Lorena Nessi, *What are AI Agents: How To Create a Based AI Agent*. Available: https://www.ccn.com/education/crypto/ai-agents-how-to-create-based-ai-agent/
3. Olesia, *AI Agents: Types, Functions, Advantages & Challenges - Exploring the World of Autonomous Intelligence*. Available: https://lablab.ai/blog/the-best-ai-agents-in-2023
4. *Atomicwork. 12 AI agent frameworks for businesses to consider in 2025.* Available: https://www.atomicwork.com/itsm/best-ai-agent-frameworks
5. Haricharaun Jayakumar, *Agent AI and Generative AI: understanding the difference*. Available: https://www.solix.com/uk/blog/agentic-ai-and-generative-ai-understanding-the-difference
6. Ilias Ism, *10 Real-World AI Agent Examples in 2024*. Available: https://www.chatbase.co/blog/ai-agent-examples
7. G. Sreedevi, *From RAG to TAG: Leveraging the Power of Table-Augmented Generation (TAG): A Leap Beyond Retrieval-Augmented Generation (RAG)*. Available: https://www.linkedin.com/pulse/from-rag-tag-leveraging-power-table-augmented-leap-beyond-gogusetty-nh8hf/
8. Hiren Dhaduk, *6 Types of AI Agents: Exploring the Future of Intelligent Machines*. Available: https://www.simform.com/blog/types-of-ai-agents/
9. Avishek Biswas, *The Ultimate Guide to RAGs — Each Component Dissected*. Available: https://towardsdatascience.com/the-ultimate-guide-to-rags-each-component-dissected-3cd51c4c0212
10. Emrecan Dogan, *What are AI agents, and how do they work?* Available: https://www.glean.com/blog/ai-agents-how-they-work
11. *Kanerika Inc. AI Agent Examples: The Future of Business and Technology*. Available: https://www.linkedin.com/pulse/ai-agent-examples-future-business-technology-kanerika-tdbcc/
12. *Agents in Artificial Intelligence.* Available: https://www.javatpoint.com/agents-in-ai
13. *Fuzen. AI Agents Examples and Types.* Available: https://fuzen.io/ai-agents-examples-and-types/
14. Samanyou Garg, *7 Types of AI Agents to Streamline Your Workflow in 2025*. Available: https://writesonic.com/blog/types-of-ai-agents
15. *Intelligent Agent Examples: Transforming Industries with AI.* Available: https://salescloser.ai/intelligent-agent-examples-transforming-industries-with-ai/
16. Sarfraz Nawaz, *Top 24 Use Cases of AI Agents in Business*. Available: https://www.ampcome.com/post/24-use-cases-of-ai-agents-in-business
17. Safa Bouguezzi, *Artificial Intelligence Agents Explained*. Available: https://www.baeldung. com/cs/artificial-intelligence-agents
18. *The 4 biggest AI stories of 2024*. Available: https://www.oksim.ua/2024/12/26/4-najbilshi-istori%D1%97-pro-shtuchnij-intelekt-za-2024-rik/
19. *OpenAI will launch an AI agent*. Available: https://babel.ua/news/112657-openai-zapustit-shi-agenta-vin-vikoristovuvatime-komp-yuter-vid-imeni-lyudini
20. Madhu Kumar, *Paradigms and Transformations in Software Development*. Available: https://dev.to/aws-heroes/the-future-of-ai-agent-driven-paradigms-and-transformations-in-software-development-55l
21. Jonghong Jeon, *SOA, Agentic AI, AI as a Service, and Future Autonomous AI*. Available: https://medium.com/@favorable_eminence_oyster_546/soa-agentic-ai-ai-as-a-service-and-future-autonomous-ai-132fdd1765fc

22. M. Alessandrini, W.M. Lippe, and W. Nuesser, “Intelligent Service System: An Agent-Based Approach for integrating Artifical Intelligence Components in SOA Landscapes,” *2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, Sydney, NSW, Australia, 2008*, pp. 496–499. doi: 10.1109/WIIAT.2008.139
23. *KOFANA. Digital. How Do AI Agents Integrate with SaaS Applications?* Available: https://www.kofana.com/en/trends/reports-and-blog/how-do-ai-agents-integrate-with-saas-applications
24. *DOMO. Guide to AI as a Service (AIaaS): Benefits, Types, Companies.* Available: https://www.domo.com/glossary/ai-as-a-service
25. *How To Build LangChain Custom Agents (Tools).* YouTube. 2024. Retrieved August 4, 2023.
26. Oleksandra, *A Brief Guide on AI Agents: Benefits and Use Cases*. Available: https://www.codica.com/blog/brief-guide-on-ai-agents/
27. G. Sreedevi, *Building Multi-Agent LLM Systems with PydanticAI Framework: A Step-by-Step Guide To Create AI Agents With Ease*. Available: https://www.linkedin.com/pulse/building-multi-agent-llm-systems-pydanticai-framework-gogusetty-w4b5f/
28. Ori Ziv, *How AI Agents Will Disrupt SaaS in 2025*. Available: https://medium.com/@oriziv4/how-ai-agents-will-disrupt-saas-in-2025-7567d793ca68
29. *Deloitte Global’s 2025 Predictions Report: Generative AI: Paving the Way for a transformative future in Technology, Media, and Telecommunication*. Available: https://www. deloitte.com/global/en/about/press-room/deloitte-globals-2025-predictions-report.html
30. Edwin Lisowski, *A List of AI Agents Set to Dominate in 2025*. Available: https://medium.com/@elisowski/a-list-of-ai-agents-set-to-dominate-in-2025-028f975c5b99
31. Mehul Gupta, *Meta Large Concept Models (LCM): End of LLMs?* Available: https://medium.com/data-science-in-your-pocket/meta-large-concept-models-lcm-end-of-llms-68cb0c5cd5cf

Received 16.01.2025

INFORMATION ON THE ARTICLE

Anatolii I. Petrenko, ORCID: 0000-0001-6712-7792, Educational and Research Institute for Applied System Analysis of the National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”, Ukraine, e-mail: tolja.petrenko@gmail.com

АГЕНТНИЙ ПІДХІД ДО ВПРОВАДЖЕННЯ ШТУЧНОГО ІНТЕЛЕКТУ (AI) В МЕЖАХ СЕРВІС-ОРІЄНТОВАНОЇ АРХІТЕКТУРИ (SOA) / А.І. Петренко

Анотація. Штучний інтелект (ШІ) стає технологією загального призначення і набуває універсального характеру для техніки, науки і суспільства, який сьогодні притаманний лише математиці та комп’ютерним технологіям. Агентний підхід до реалізації ШІ в межах сервіс-орієнтованої архітектури додатків є захопливою і дуже синергетичною концепцією. Поєднання цих парадигм призводить до створення надійних, масштабованих та інтелектуальних систем, які добре підходять для динамічних і розподілених середовищ. Подано результати порівняльного аналізу трьох можливих підходів до інтеграції ШІ в бізнес-процеси, а саме: підключення агентів ШІ до сервіс-орієнтованої архітектури (SOA), підключення агентів ШІ до програмного забезпечення (SaaS) та побудова ШІ як послуги (AIaaS). Розглянуто потенційні переваги, виклики, приклади та міркування із застосуванням кожного з цих підходів.

Ключові слова: AI (Artificial Intelligence), агентний AI, AI-agent, SOA (Service oriented architecture), SaaS (Software-as-a-Service), RAG (Retrieval-Augmented Generation), великі мовні моделі (LLM), одноагентні і багатоагентні системи, платформи розроблення AI агентів, інтеграція агентів AI з SaaS, агенти AI і SOA.

UDC 519.925.51
DOI: 10.20535/SRIT.2308-8893.2025.1.09

INVESTIGATION OF THE EFFECTIVENESS OF ARTIFICIAL NEURAL NETWORKS OF DIFFERENT GENERATIONS IN THE TASK OF FORECASTING IN THE FINANCIAL SPHERE

Ye. BODYANSKIY, Yu. ZAYCHENKO, He. ZAICHENKO, O. KUZMENKO

Abstract. This paper discusses ANNs of different generations. The efficiency of using computational intelligence in the task of short- and medium-term forecasting in the financial sphere is investigated. For the investigation, a fully connected feed-forward network (Back Propagation), a recurrent network (LSTM), a hybrid deep learning network based on self-organization (GMDH neo fuzzy), and a hybrid system of computational intelligence based on bagging and group method of data handling (HSCI bagging) were chosen. The experimental parameters chosen are the prediction interval, the number of inputs, the percentage of validation data in the training set, and the number of fuzzifiers (for GMDH neo-fuzzy). Experiments were conducted, and the best results for different prediction intervals were compared. The optimal parameters of the networks and the feasibility of their use in the task of forecasting at different intervals are determined.

Keywords: generations of ANNs, Back Propagation, LSTM, GMDH neo fuzzy, HSCI bagging.

INTRODUCTION

Generations of artificial neural networks represent different stages in the evolution of artificial intelligence (AI) and machine learning technologies. Each generation introduces new approaches, architectures, and improvements that make neural networks more powerful and efficient.

The first generation of artificial neural networks encompasses early developments in artificial intelligence and machine learning. The basic model of this generation is the perceptron, developed in 1957 by Frank Rosenblatt [1]. It was the simplest type of artificial neural network and consisted of three main components. The S-System (Sensory System) was represented by a set of points in a TV raster, or a set of photocells. The A-System (Association System) performed the switching functions between input and output. The R-System (Response System) consisted of a typically of a relatively small number of units. Such a model allowed solving only linearly separable problems, i.e. problems where a straight line can be drawn to separate data classes. However, it was not possible to solve more complex problems, such as the XOR-problem, where classes cannot be separated by a straight line. Although the perceptron was an important step forward, its capabilities were limited. After researchers realized that it could not solve nonlinear problems, the development of neural networks slowed down for a while, but the first generation of neural networks laid the foundation for future research. It demonstrated the ability of machines to learn from experience, albeit with limited capabilities. This led to the further development of more complex models in the following generations.

© *Publisher IASA at the Igor Sikorsky Kyiv Polytechnic Institute*, 2025

The second generation of artificial neural networks has advanced significantly compared to the first, introducing multi-layer architectures and advanced learning methods such as the back-propagation algorithm. This generation opened up new possibilities for solving more complex problems that could not be solved by simple first-generation models. The main feature of this generation was the introduction of Multilayer Perceptrons (MLPs). Such networks consisted of several layers of neurons: an input layer, one or more hidden layers, and an output layer. The second generation will include the Back Propagation neural network, which was proposed by Rumelhart, Hinton, and Williams in 1986 [2]. The authors of this paper first showed how to train such a network with an arbitrary number of layers and proposed a recurrent gradient-type algorithm for training it for a network with an arbitrary structure. This network belongs to the class of feed-forward neural networks. The Back Propagation Network has been widely used in numerous tasks of function approximation, forecasting, and pattern recognition. Its versatility is defined by the Universal Approximation Theorem [3]. With its multilayer architecture and backpropagation algorithm, the second generation of neural networks was able to solve problems that cannot be separated linearly, such as the XOR-problem. This significantly expanded the scope of neural networks.

The third generation of artificial neural networks was marked by the emergence of Deep Learning [4], an approach that has led to significant breakthroughs in many areas of artificial intelligence. Deep neural networks consist of a large number of layers (deep architecture), which allows modeling more complex and abstract data representations. Deep Learning requires large amounts of data for training and powerful computing resources, in particular graphics processing units (GPUs). This has become possible due to the development of data storage and processing technologies, as well as improvements in hardware. It is worth noting that Jurgen Schmidhuber refers to the Group Method of Data Handling (GMDH) [5; 6] as the earliest deep learning method, noting that it was used to train a neural network consisting of eight layers back in 1971 [7]. This method was proposed in the late 60s and early 70s by acad. A.G. Ivakhnenko and his colleagues. This method is based on the selective selection of models on the basis of which more complex models are built. The modeling accuracy at each subsequent step increases due to the model's complexity. Solving complex problems has led to the emergence of new types of neural networks. A special type of deep neural network, convolutional neural networks (CNNs) [8], has been developed for working with images. They use special layers (convolutional layers) that can detect various image features, such as contours, textures, and objects, at different levels of abstraction. Recurrent neural networks (RNNs) [9–11] are designed to work with sequences of data, such as text or audio. They store information about previous processing steps, which allows them to consider context in natural language processing and speech recognition tasks. In addition to error backpropagation, the third generation uses advanced optimization, regularization, and normalization techniques to train deep networks more efficiently, reducing the likelihood of overtraining. The third generation of neural networks has become a real breakthrough in the field of artificial intelligence. This generation, artificial neural networks have become a key tool in the development of intelligent systems that are now used in everyday life.

The fourth generation of artificial neural networks is characterized by the emergence of new architectures and methods that have further expanded the capabilities of artificial intelligence. The main feature of this generation is the ability to perform several different tasks using a single architecture. Quite often, new approaches to training are used, such as pre-training methods on large data sets followed by fine-tuning on specific tasks, which allows creating models that can easily adapt to different contexts and tasks. This hybrid approach to building neural networks allows them to be used in such industries as medical diagnostics, virtual assistants, business process automation, autonomous vehicles, personalized advertising, and much more. Their versatility and adaptability are also achieved through self-organization. The fourth generation is represented by self-organizing deep learning networks [12–15]. The key feature of this type of network is that it builds its structure in the process of learning. Transformers have become an important innovation of the fourth generation. Transformers use a self-attention mechanism that allows the model to efficiently process sequential data, such as text, and consider the dependencies between sequence elements regardless of their location. Large Language Models (LLMs) are one of the most striking achievements of the fourth generation, such as GPT (Generative Pre-trained Transformer), BERT (Bidirectional Encoder Representations from Transformers), and others. These models are capable of generating, analyzing, and understanding text at a level that was previously considered unattainable. They can perform translation, text generation, question answering, and other tasks. In addition to language models, the fourth generation includes powerful generative models such as DALL-E, Stable Diffusion, and others. These models can create images based on textual descriptions, opening up new possibilities in creativity, design, and many other industries. The fourth generation of artificial neural networks has significantly expanded the boundaries of what is possible in the field of artificial intelligence. The use of transformers and large language models has allowed for new heights in understanding and generating natural language, which has become the foundation for many innovations in various industries. This generation has also emphasized the importance of multitasking and versatility, as models have become capable of performing a variety of tasks using a single architecture. Such advances continue to change the technological landscape and drive the further development of artificial intelligence, making it even more powerful and useful in various aspects of life.

What will the next generation of artificial neural networks look like? What opportunities will we have? What tasks will we be able to solve? To answer these questions, it is worth taking a closer look at the evolution of the structure of artificial neural network training algorithms and practically exploring their capabilities.

EVOLUTION OF ANNS

The perceptron [1] is the basis for more complex neural networks, where hidden layers are added to solve nonlinear problems and work with more complex data. The Back Propagation neural network has a multilayer fully connected architecture. During training, neuron weights are adjusted to reduce the error between the predicted result and the actual value [2]. This process ensures efficient training of the neural network based on a large amount of data [3].

The third-generation LSTM neural network is more complex (Fig. 1).

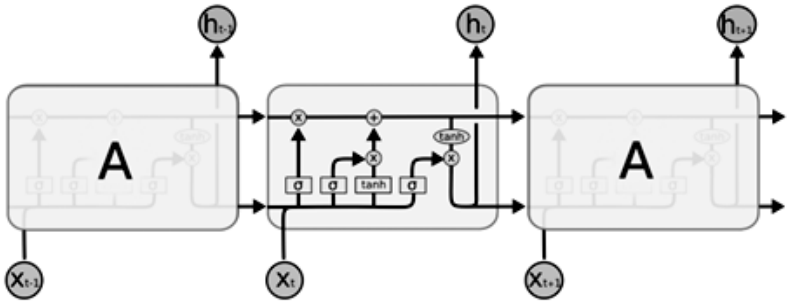


Fig. 1. LSTM recurrent network architecture

A forward pass through a single LSTM block consists of several main steps that are supposed to interact with the internal network state [4; 10; 11]. The first step is a forget gate unit that decides which information should be erased in the internal state. The internal state saves all information from all previous steps. The process of "forgetting" information from previous steps can be expressed with the usage of sigmoid function and weights matrices:

$$f_i^{(t)} = \sigma\left(b_i^f + \sum_j U_{i,j}^f x_j^{(t)} + \sum_j W_{i,j}^f h_j^{(t-1)} \right), \tag{1}$$

where $x^{(t)}$ is a current input vector at time step t; $h^{(t)}$ is a hidden unit vector at the current time step that contains information from LSTM block outputs in the previous time steps; b_i^f is forget gate bias vector; U^f is a matrix of input weights for forget gate; W^f is a matrix of recurrent weights for forget gate.

The next step for LSTM block consists of several intermediate steps. First, the input gate decides which information in the internal state should be updated with new data. Then, the network creates a list of new elements that reflect new information that should be added to the internal state. Finally, the network combines all information from previous steps and updates the internal state $s_i^{(t)}$. All these operations are described with the following equation:

$$s_i^{(i)} = f_i^{(t)} s_i^{(t-1)} + g_i^{(t)} \sigma\left(b_i + \sum_j U_{i,j} x_j^{(t)} + \sum_j W_{i,j} h_j^{(t-1)} \right), \tag{2}$$

$$g_i^{(t)} = \sigma\left(b_i^g + \sum_j U_{i,j}^g x_j^{(t)} + \sum_j W_{i,j}^g h_j^{(t-1)} \right), \tag{3}$$

where b is a bias vector into LSTM block; U — input weights in the LSTM block; W is recurrent weights into LSTM block; $g_i^{(t)}$ is an external input gate function.

The last step of LSTM block decides which information should be returned as output. Output value calculated using output gate mechanism:

$$h_i^{(t)} = \tanh\left(s_i^{(t)} q_i^{(t)}\right), \tag{4}$$

$$q_i^{(t)} = \sigma\left(b_i^o + \sum_i U_{i,j}^o x_j^{(t)} + \sum_j W_{i,j}^o h_j^{(t-1)} \right), \tag{5}$$

where b^o, U^o, W^o is respectively bias vector, input, and recurrent weights matrices of output gate.

For training LSTM stochastic gradient method and its modern modifications are used. LSTM architecture has been successful on real-world tasks in different domains and shows that it works much better with long-term dependencies than poor RNNs.

A hybrid deep learning network based on self-organization — GMDH-neo-fuzzy – is a fourth-generation network (Fig. 2) [12; 13].

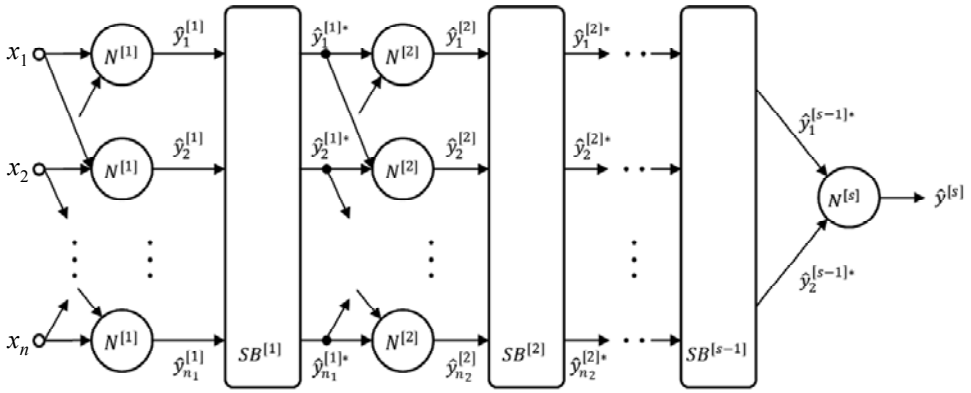


Fig. 2. The process of synthesis of the GMDH-neo-fuzzy network structure

To the system's input layer a $(n\times 1)$-dimensional vector of input signals is fed. After that this signal is transferred to the first hidden layer. This layer contains $n_1 = c_n^2$ nodes, and each of these neurons has only two inputs.

At the outputs $N^{[1]}$ of the first hidden layer the output signals are formed. Then these signals are fed to the selection block of the first hidden layer.

It selects among the output signals $\hat{y}_l^{[1]} n_1 *$ ($n_1 * = F$ is so called freedom of choice) most precise signals by some chosen criterion (mostly by the mean squared error $\sigma^2_{y_l^{[1]}}$). Among these n_1^* best outputs of the first hidden layer $\hat{y}_l^{[1]} * n_2$ pairwise combinations $\hat{y}_l^{[1]} *$, $\hat{y}_p^{[1]} *$ are formed. These signals are fed to the second hidden layer, that is formed by neurons $N^{[2]}$. After training these neurons output signals of this layer $\hat{y}_l^{[2]}$ are transferred to the selection block $SB^{[2]}$ which choses F best neurons by accuracy (e.g. by the value of $\sigma^2_{y_l^{[2]}}$) if the best signal of the second layer is better than the best signal of the first hidden layer $\hat{y}_1^{[1][\ ]} *$. Other hidden layers forms signals similarly. The system evolution process continues until the best signal of the selection block $SB^{[s+1]}$ appears to be worse than the best signal of the previous s th layer. Then we return to the previous layer and choose its best node neuron $N^{[s]}$ with output signal $\hat{y}^{[s]}$. And moving from this neuron (node) along its connections backwards and sequentially passing all previous layers we finally get the structure of the GMDH-neo-fuzzy network.

It should be noted that in such a way not only the optimal structure of the network may be constructed but also well-trained network due to the GMDH algorithm [5; 6]. Besides, since the training is performed sequentially layer by layer the problems of high dimensionality as well as vanishing or exploding gradient are avoided.

Let's introduce into consideration the architecture of the node that is suggested as a neuron of the GMDH-system. As a node of this structure a neo-fuzzy neuron (NFN) by Takeshi Yamakawa and co-authors in is used [14]. The neo-fuzzy neuron is a nonlinear multi-input single-output system shown in Fig. 3. The main difference of this node from the general neo-fuzzy neuron structure is that each node uses only two inputs.

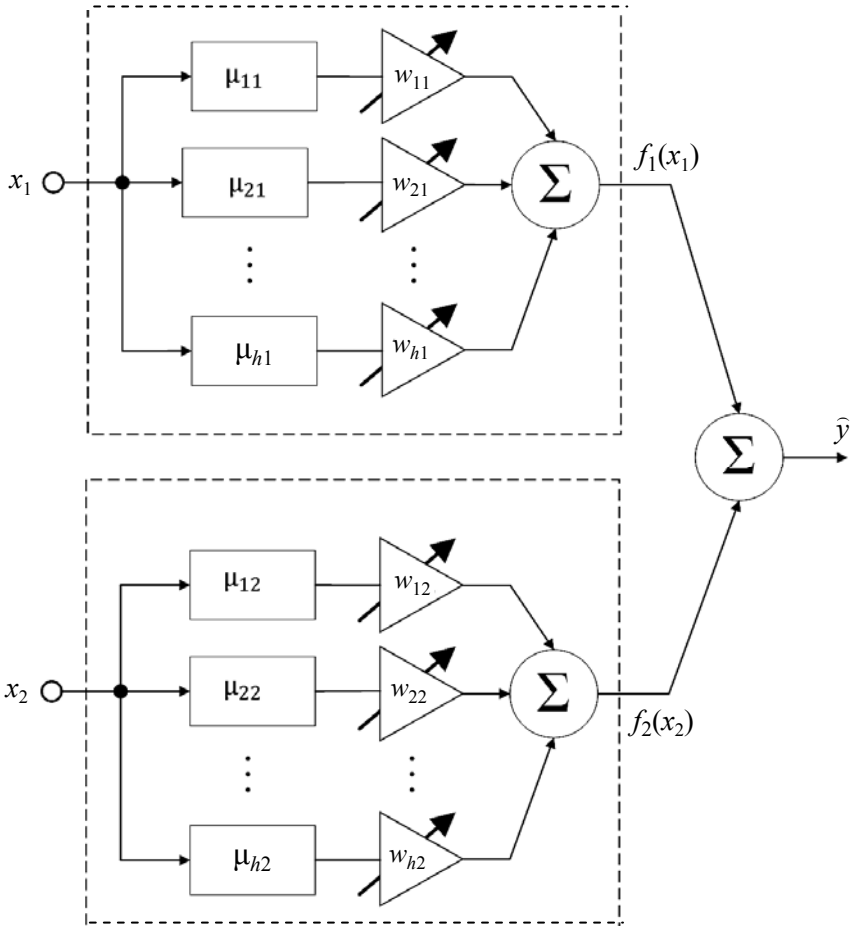


Fig. 3. Neo-fuzzy neuron with two inputs

It realizes the following mapping:

$$\hat{y} = \sum_{i=1}^{2} f_i(x_i), \tag{6}$$

where x_i is the input $i\,(i = 1,2,\ldots,n)$, $\hat{y}$ is a system output. Structural blocks of neo-fuzzy neuron are nonlinear synapses NS_i which perform transformation of input signal in the form

$$f_i(x_i) = \sum_{j=1}^{h} w_{ji} \mu_{ji}(x_i) \tag{7}$$

and realize fuzzy inference: if x_i is x_{ji} then the output is w_{ji}, where x_{ji} is a fuzzy set which membership function is μ_{ji}, w_{ji} is a synaptic weight in consequent.

The learning criterion (goal function) is the standard local quadratic error function:

$$E(k) = \frac{1}{2}(y(k) - \hat{y}(k))^2 = \frac{1}{2}e(k)^2 = \frac{1}{2}\left(y(k) - \sum_{i=1}^{2}\sum_{j=1}^{h} w_{ji}\mu_{ji}(x_i(k)) \right)^2. \tag{8}$$

It is minimized via the conventional stochastic gradient descent algorithm.

In case we have priori defined data set the training process can be performed in a batch mode for one epoch using conventional least squares method

$$w^{[1]}(N) == \left(\sum_{k=1}^{N} \mu^{[1]}(k)\mu^{[1]T}(k) \right)^{+} \sum_{k=1}^{N} \mu^{[1]}(k) y(k) = P^{[1]}(N) \sum_{k=1}^{N} \mu^{[1]}(k) y(k), \tag{9}$$

where $(\bullet)^+$ means pseudo inverse of Moore–Penrose (here $y(k)$ denotes external reference signal (real value).

If training observations are fed sequentially in on-line mode, the recurrent form of the LSM can be used in the form

$$w_l^{ij}(k) = w_l^{ij}(k-1) + \frac{\mathrm{P}^{ij}(k-1)(y(k) - (w_l^{ij}(k-1))^T \varphi^{ij}(x(k)))\varphi^{ij}(x(k))}{1 + (\varphi^{ij}(x(k)))^T \mathrm{P}^{ij}(k-1)\varphi^{ij}(x(k))},$$

$$\mathrm{P}^{ij}(k) = \mathrm{P}^{ij}(k-1) - -\frac{\mathrm{P}^{ij}(k-1)\varphi^{ij}(x(k))(\varphi^{ij}(x(k)))^T \mathrm{P}^{ij}(k-1)}{1 + (\varphi^{ij}(x(k)))^T \mathrm{P}^{ij}(k-1)\varphi^{ij}(x(k))}. \tag{10}$$

Consider the HSCI-bagging network (Fig. 4). It is a hybrid system of computational intelligence (HSCI) built on the basis of the ensemble approach and batching, which builds its architecture in the process of learning based on the ideas of GMDH [15].

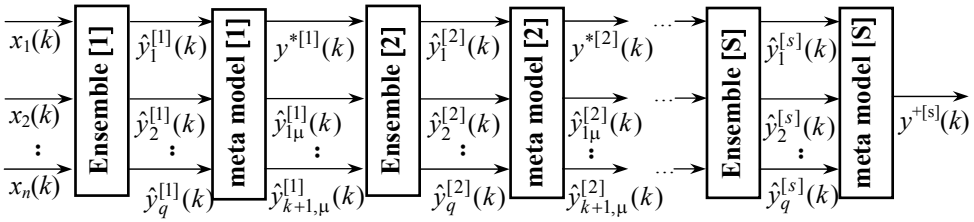


Fig. 4. Hybrid system of computational intelligence based on bagging and GMDH

The architecture of the system contains 2S sequentially connected stacks, while odd stacks are formed by ensembles of parallel-connected subsystems that solve the same problem (recognition, prediction, etc.) and even ones are essentially learning metamodels that generalize the output signals of ensembles and form optimal results in the sense of the accepted criterion. The output signal of the first metamodel is the generalized optimal signal $y^{*1}(k)$ and $(n-1)$ output signals $\hat{y}_{i\mu}^{[1]}(k)$, $i = 1,2...,n-1(k)$ "best members of the ensemble". At their core,

metamodels function as selection units in traditional GMDH systems, but not only select the best results from the previous stack, but also form the optimal solution based on these results.

Further, the output signals of the first metamodel are fed to the inputs of the second ensemble, which is completely similar to the first. The outputs of the second ensemble $\hat{y}_1^{[2]}(k)$, $\hat{y}_2^{[2]}(k),...,\hat{y}_q^{[2]}(k)$ come to the second metamodel, which calculates the optimal signal $y^{*[2]}(k)$ and $(n-1)\hat{y}_{i\mu}^{[2]}(k)$ "closest" to it. The last *S*-th ensemble is similar to the first two, and the output of the last *S*-th metamodel is $y^{*[s]}(k)$, which exactly corresponds to a priori established requirements for the quality of solving the problem under consideration.

Each of the ensembles contains q different computational intelligence systems that solve the same problem. There may still be simple neural networks such as a single-layer perceptron, radial- basis neural network (RBFN), counterpropagating neural network, etc., which do not use error backpropagation procedure for training, neuro-fuzzy systems such as ANFIS, Wang–Mendel or Takagi–Sugeno–Kang type, wavelet-neuro systems, neo-fuzzy neurons and others, the output signal of which linearly depends on the adapted parameters, which allows to use optimal speed learning algorithms.

The input information, on the basis of which the system is configured, is a training selection of input signals:

$$x(1), x(2),\ldots, x(k),\ldots, x(N),$$

$$x(k) = (x_1(k),\ldots, x_i(k),\ldots, x_n(k))^{\mathrm{T}} \in R^n, \tag{11}$$

and its corresponding scalar refence signals $y(1),...,y(k),...,y(N)$. On the basis of these observations, the elements of the first ensemble are tuned independently of each other, at the outputs of which q scalar signals $\hat{y}_p^{[2]}(k)$, $p = 1,2,...,q$, are formed, which are conveniently represented in the form of a vector $\hat{y}^{[1]}(k) = (\hat{y}_1^{[1]}(k),\ldots,\hat{y}_p^{[1]}(k),\ldots,\hat{y}_q^{[1]}(k))^{\mathrm{T}}$. These signals are sent to the inputs of the first metamodel, at the outputs of which n sequences $\hat{y}^{*[1]}(k)$, $\hat{y}_{1\mu}^{[1]}(k),\ldots,\hat{y}_{i\mu}^{[1]}(k),\ldots,\hat{y}_{n-1,\mu}^{[1]}(k)$ the main of which is $y^{*[1]}(k)$ while others are auxiliary. The main signal of the metamodel $y^{*[1]}(k)$ is the union of the outputs of all members of the ensemble in the form of:

$$y^{*[1]}(k) = \sum_{p=1}^{q} w_p^{*[1]}\, \hat{y}_p^{[1]}(k) = \hat{y}^{[1]T}(k) w^{*[1]}, \tag{12}$$

where $w^{*[1]} = (w^{*[1]},...,w_p^{*[1]}...w_q^{*[1]})^{\mathrm{T}}$ is a vector of adapted parameters-synaptic weights on which additionally restrictions are set on unbiasedness:

$$\sum_{p=1}^{q} w_p^{*[1]} = I_q^T\, w^{*[1]} = 1, \tag{13}$$

where I_q — $(q \times 1)$ is the vector of unities.

The problem of teaching the first metamodel is reduced to minimizing the standard quadratic criterion in the presence of additional constraints.

Thus, the problem of training the first metamodel can be solved using the standard method of penalty functions, which in this case reduces to minimizing the expression:

$$J\ (w^{*[1]},\delta) = (Y(N) - \overline{Y}^{[1]}(N)w^{*[1]})^{\mathrm{T}}(Y(N) - \\ - \overline{Y}^{[1]}(N)w^{*[1]}) + \delta^{-2}(I_q^T\ w^{*[1]} - 1), \qquad (14)$$

where $Y(N) = (y(1),\ldots,y(k),\ldots,y(N))^{\mathrm{T}}$ — is a vector, $\overline{Y}^{1}(N) = (\hat{y}^{[1]}(1),\ldots$ $\ldots,\hat{y}^{[1]}(k),\ldots,\hat{y}^{[1]}(N))^{\mathrm{T}}$ — $(N\times q)$ is a matrix, δ is the penalty coefficient.

As a result of learning the first metamodel, the optimal signal $y^{*[1]}(k)$ is formed at its output, as well as q signals $\hat{y}^{[1]}_{p,\mu}(k)$ from which we choose $n-1$ (if $q \geq n$) with the highest levels of fuzzy membership $\mu_p^{[1]}$, which subsequently in the form of $(n\times 1)$ — vector are fed to the input of the second ensemble, the outputs of which go to the inputs of the second metamodel, and so on. The process of increasing the number of ensembles and metamodels continues until the required accuracy of the last metamodel with the output $y^{*[s]}(k)$ is achieved, or the value of the criterion minimized for the bagging model begins to increase, i.e. $\overline{\varepsilon}^2\,(y^{*[s+1]}(k)) \geq \overline{\varepsilon}^2\,(y^{*[s]}(k))$.

DATA SET

Data on the dynamics of changes in the Dow Jones Industrial Average (DJIA) index from August 2, 2022 to July 31, 2024 were used for forecasting [16]. The dynamics of DJIA Close values is shown in the Fig. 5.

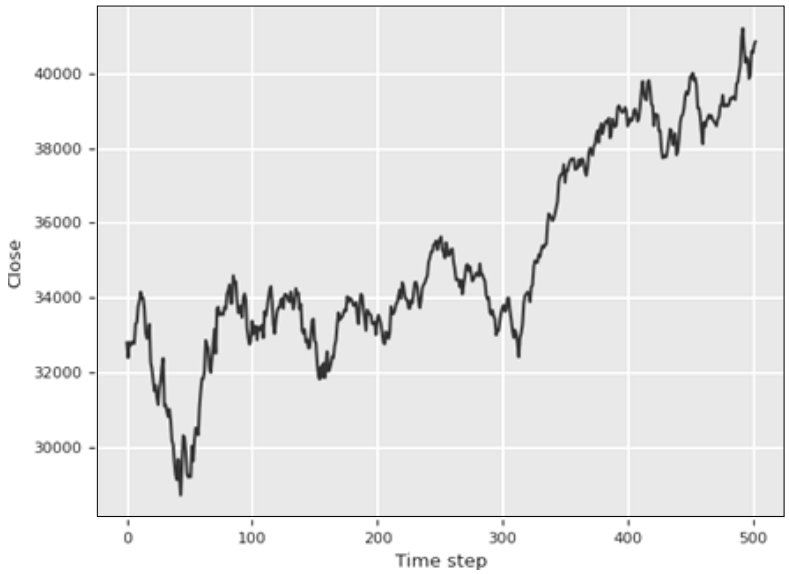


Fig. 5. Dynamics of the index Close DJIA

The correlogram of DJIA Close vales is presented in the Fig. 6.

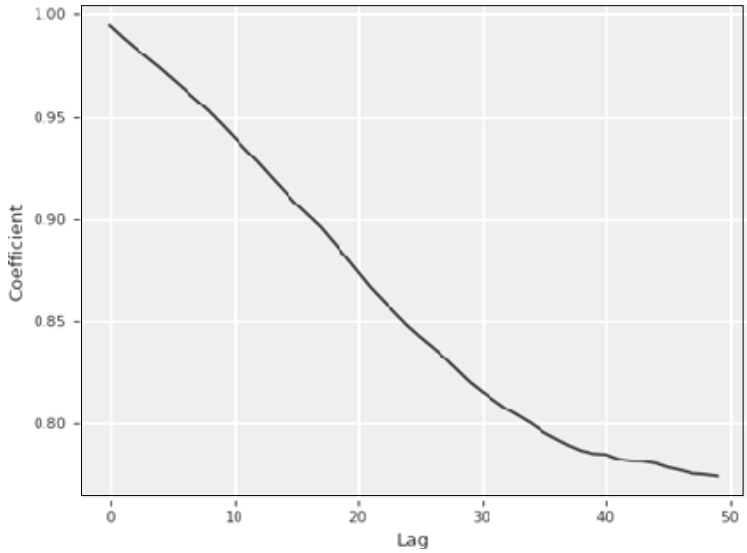


Fig. 6. Correlogram of the index Close DJIA

Analyzing the presented curve, one may conclude that there is strong correlation between preceding and conceding values and even for lag 50 days the correlation is more than 0.7.

EXPERIMENTAL INVESTIGATION AND DISCUSSION

Experimental studies of the accuracy of index forecasting using networks were conducted: Back Propagation, LSTM, HMDH-neo-fuzzy, and HSCI-bagging. During the experiments, the values of the parameters presented in Table 1 were changed. The data set was split into three subsets: training, validation, and test. The test subset of data for all experiments had a fixed size (30 last points from the dataset) and was not used for training and validation.

T a b l e 1. Experimental Parameters

Parameter	Values
Interval	1; 3; 5; 7; 15; 20; 30
Number of inputs	3; 4; 5
Number of fuzzifiers	2; 3; 4
Validation split	0.4; 0.3; 0.2

After training, the accuracy of the models was checked on a test sample. For each network, the best results of the forecast accuracy according to the MAPE criterion were determined.

The first set of experiments was conducted with a back propagation network (2nd generation). It had three hidden layers: the first layer of 7 neurons, the second layer of 5 neurons, and the third layer of 3 neurons. The output was a single signal. The best prediction results of this network for all intervals are shown in Table 2.

Table 2. The best results of the Back Propagation network

Interval	Number of inputs	Validation split	MSE	MAPE
1	3	0.2	1644597.0402	2.9792
3	3	0.2	1824141.7299	3.0973
5	3	0.2	1820227.1563	3.1028
7	3	0.2	1847014.9194	3.1448
15	4	0.2	1926517.248	3.175
20	4	0.2	4176396.3173	4.8037
30	3	0.2	24150324.9138	12.1998

The second set of experiments investigated the prediction accuracy of the 3rd generation network — LSTM. It had the following structure: input signals determined by an experimental parameter, three hidden layers (32 neurons, 16 neurons, and 8 neurons), and one neuron with an output signal. For all forecasting intervals, the best results were determined and are shown in Table 3.

For the Back Propagation and LSTM networks, the structure was selected using Cross-Validation and Grid Search methods. As a result, optimal structures were obtained and used that do not have too many hidden layers. This approach made it possible to use minimal computational costs to obtain sufficiently high accuracy of the results.

Table 3. The best results of the LSTM network

Interval	Number of inputs	Validation split	MSE	MAPE
1	5	0.2	96500.7026	0.582
3	4	0.2	258966.107	0.979
5	5	0.2	474929.5015	1.3487
7	5	0.2	537666.5732	1.4243
15	3	0.2	1159702.2855	2.0442
20	4	0.2	2847643.2154	3.7911
30	4	0.3	4233239.0843	4.6857

The third set of experiments was conducted to determine the forecasting accuracy of the 4th generation network — GMDH-neo-fuzzy. The structure of the network was synthesized during training and in most cases had two hidden layers and one layer with the output signal. After comparing the obtained forecasting results on the test subsample, Table 4 was created with the best results and optimal network parameters for all intervals.

Table 4. The best results of the GMDH-neo-fuzzy

Int.	Number of inputs	Number of fuzzifiers	Validation split	MSE	MAPE
1	3	2	0.3	155027.4886	0.7004
3	3	3	0.2	332615.1181	1.1576
5	5	3	0.3	417632.8307	1.198
7	5	4	0.3	394795.8022	1.2579
15	5	3	0.4	622501.8958	1.6785
20	4	3	0.3	1086960.4794	2.204
30	4	3	0.3	1138011.8375	2.353

The last series of experiments was conducted to evaluate the prediction accuracy of the HSCI-bagging network, which also belongs to the 4th generation. The previous networks were used for its implementation. The best forecasting results are presented in Table 5.

Table 5. The best results of the HSCI-bagging

Interval	Number of inputs	Validation split	MSE	MAPE
1	5	0.2	80602.0465	0.5054
3	4	0.2	248445.23	0.9685
5	4	0.2	384022.2476	1.1568
7	4	0.2	409905.6681	1.2263
15	5	0.2	595730.7206	1.6062
20	5	0.2	827577.2912	2.011
30	4	0.2	978388.1249	2.1217

Based on the data from Tables 2–5, Table 6 was created to compare the forecasting results according to the MAPE criterion.

Table 6. Comparative table of the best forecasting results

Interval	Back Propagation	LSTM	GMDH-neo-fuzzy	HSCI-bagging
1	2.9792	0.582	0.7004	0.5054
3	3.0973	0.979	1.1576	0.9685
5	3.1028	1.3487	1.198	1.1568
7	3.1448	1.4243	1.2579	1.2263
15	3.175	2.0442	1.6785	1.6062
20	4.8037	3.7911	2.204	2.011
30	12.1998	4.6857	2.353	2.1217

For the convenience of analyzing the results, a comparative graph of the average forecast accuracy according to the MAPE criterion for all the investigated networks at each of the intervals was also constructed. This graph is shown in Fig. 7.

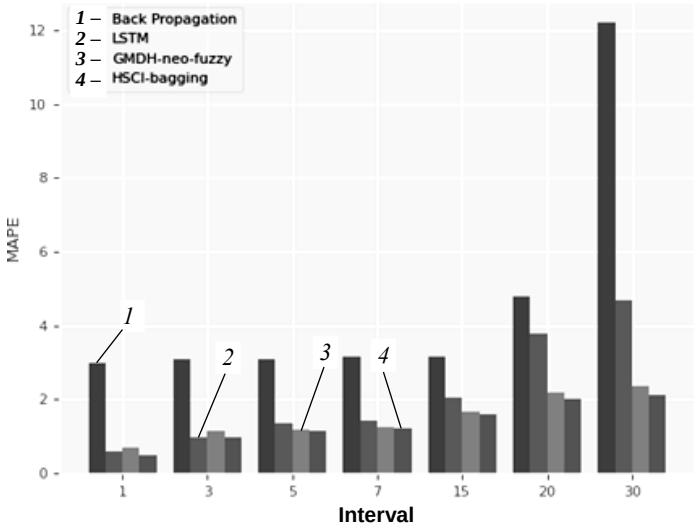


Fig. 7. Comparative diagram of forecast accuracy according to the MAPE criterion

Thus, the results of the conducted studies show that the Back Propagation network showed the worst results at all intervals. The best forecasting accuracy according to the MAPE criterion was obtained using the 4th generation HSCI-bagging network, its slightly better than hybrid GMDH-neo-fuzzy network. The LSTM recurrent network showed good results on short-term intervals, but starting from interval 5, it is inferior to the GMDH-neo-fuzzy network.

CONCLUSION

This article considers the problem of short- and middle-term forecasting in the financial sector using the Dow Jones Industrial Averagse (DJIA) dataset.

Experimental investigations of the forecasting accuracy of neural networks of different generations were conducted: Back Propagation (2nd generation), LSTM (3rd generation), GMDH-neo-fuzzy (4th generation) and HSCI-bagging (4th generation).

During the experiments, at each of the short- and medium-term intervals, the optimal parameters for each of the networks were determined, at which it demonstrated the best forecasting results.

The accuracy of forecasts by the MAPE criterion of all networks at short and medium-term intervals was compared. The best forecasting results were obtained using HSCI-bagging, and the GMDH-neo-fuzzy hybrid network showed slightly worse results, but better than other studied networks of previous generations.

The results of the investigation show that, in general, the forecasting accuracy increases with the generation of neural networks. In addition, the latest generations of artificial neural networks have shown better results on medium-term intervals.

REFERENCES

1. Frank Rosenblatt, “The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain,” *Psychological Review*, vol. 65, no. 6, pp. 386–408, 1958.
2. David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams, “Learning representations by back-propagating errors,” *Nature*, vol. 323, pp. 533–536, 1986. doi: 10.1038/323533a0
3. G.V. Cybenko, “Approximation by Superpositions of a Sigmoidal function,” *Mathematics of Control, Signals and Systems*, vol. 2, no. 4, pp. 303–314, 1989, doi: 10.1007/BF02551274
4. I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016. Available: http://www.deeplearningbook.org
5. A.G. Ivakhnenko, V.G. Lapa, *Cybernetic forecasting device*s (in Ukrainian). K.: Naukova Dumka, 1965, 216 p.
6. A.G. Ivakhnenko, G.A. Ivakhnenko, and J.A. Mueller, “Self-organization of the neural networks with active neurons,” *Pattern Recognition and Image Analysis*, vol. 4, no. 2, pp. 177–188, 1994.
7. Jürgen Schmidhuber, “Deep learning in neural networks: An overview,” *Neural Networks*, pp. 85–117, 2015. doi: 10.1016/j.neunet.2014.09.003
8. Purwono Purwono et al., “Understanding of Convolutional Neural Network (CNN): A Review,” *International Journal of Robotics and Control Systems*, vol. 2, no. 4, pp. 739–748, 2023. doi: 10.31763/ijrcs.v2i4.888
9. B. Hammer, “On the approximation capability of recurrent neural networks,” *Neurocomputing*, vol. 31, pp. 107–123, 1998. doi: 10.1016/S0925-2312(99)00174-5
10. S. Hochreiter, J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, pp. 1735–1780, 1997. doi: 10.1162/neco.1997.9.8.1735

11. C. Olah, *Understanding LSTM networks*. 2020. Available: https://colah.github.io/posts/2015-08- Understanding-LSTMs
12. Yu. Zaychenko, Galib Hamidov, "The Hybrid Deep Learning GMDH-neo-fuzzy Neural Network and Its Applications," *Proceedings of 13-th IEEE International Conference Application of Information and Communication Technologies-AICT2019. 23–25 October 2019, Baku*, pp. 72–77. doi: 10.1109/AICT47866.2019.8981725
13. Evgeniy Bodyanskiy, Yuriy Zaychenko, Olena Boiko, Galib Hamidov, and Anna Zelikman, "Structure Optimization and Investigations of Hybrid GMDH-Neo-fuzzy Neural Networks in Forecasting Problems," *System Analysis & Intelligent Computing*; Eds. Michael Zgurovsky, Natalia Pankratova. Book Studies in Computational Intelligence, SCI, vol. 1022. Springer, 2022, pp. 209–228.
14. T. Yamakawa, E. Uchino, T. Miki, and H. Kusanagi, "A neo-fuzzy neuron and its applications to system identification and prediction of the system behavior," *Proc. 2nd Intern. Conf. Fuzzy Logic and Neural Networks "LIZUKA-92", Lizuka, 1992*, pp. 477–483.
15. Ye. Bodyanskiy, O. Kuzmenko, He. Zaichenko, and Yu. Zaychenko, "Hybrid System of Computational Intelligence based on Bagging and Group Method of Data Handling," *System Research and Information Technologies*, no. 1, pp. 75–85, 2024. doi: 10.20535/SRIT.2308-8893.2024.1.06
16. "DJIA - Dow Jones Industrial Average Historical Prices," *WSJ*. Accessed on: August 1, 2024. [Online]. Available: https://www.wsj.com/market-data/quotes/index/DJIA/historical-prices

Received 04.09.2024

INFORMATION ON THE ARTICLE

Yevgeniy V. Bodyanskiy, ORCID: 0000-0001-5418-2143, Kharkiv National University of Radio Electronics, Ukraine, e-mail: yevgeniy.bodyanskiy@nure.ua

Yuriy P. Zaychenko, ORCID: 0000-0001-9662-3269, Educational and Research Institute for Applied System Analysis of the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Ukraine, e-mail: zaychenkoyuri@ukr.net

Helen Yu. Zaichenko, ORCID: 0000-0002-4630-5155, Educational and Research Institute for Applied System Analysis of the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Ukraine, e-mail: zaichenko.helen@lll.kpi.ua

Oleksii V. Kuzmenko, ORCID: 0000-0003-1581-6224, Educational and Research Institute for Applied System Analysis of the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Ukraine, e-mail: oleksii.kuzmenko@ukr.net

ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ (ШНМ) РІЗНИХ ПОКОЛІНЬ У ЗАДАЧІ ПРОГНОЗУВАННЯ У ФІНАНСОВІЙ СФЕРІ / Є.В. Бодянський, Ю.П. Зайченко, О.Ю. Зайченко, О.В. Кузьменко

Анотація. Розглянуто ШНМ різних поколінь. Досліджено ефективність використання обчислювального інтелекту в задачах коротко- та середньострокового прогнозування у фінансовій сфері. Для дослідження обрано повнозв'язну мережу прямого поширення (Back Propagation), рекурентну мережу (LSTM), гібридну мережу глибокого навчання на основі самоорганізації (GMDH-neo-fuzzy) та гібридну систему обчислювального інтелекту на основі беггінгу та методу групового урахування аргументів (HSCI-bagging). Як експериментальні параметри обрано інтервал прогнозування, кількість входів, відсоток валідаційних даних у навчальній вибірці та кількість фазифікаторів (для GMDH-neo-fuzzy). Проведено експерименти та порівняно найкращі результати, отримані для різних інтервалів прогнозування. Визначено оптимальні параметри мереж та доцільність їх використання в задачі прогнозування на різних інтервалах.

Ключові слова: покоління ШНМ, Back Propagation, LSTM, GMDH neo fuzzy, HSCI bagging.

UDC 004.738.5
DOI: 10.20535/SRIT.2308-8893.2025.1.10

ASSESSING THE IMPACT OF AI-GENERATED PRODUCT NAMES ON E-COMMERCE PERFORMANCE

O. BRATUS

Abstract. This paper studies the impact of Large Language Model (LLM) technology on the e-commerce industry. This work conducts a detailed review of the current implementation level of LLM technologies in the e-commerce industry. Next, it analyzes the approaches to detecting AI-generated text and determines the limitations of their application. The proposed methodology defines the impact of LLM models on the e-commerce industry based on a comparative analysis between indicators of machine-generated texts and e-commerce product metrics. Applying this methodology to real data, one of the most relevant data collected after the release of ChatGPT, the results of statistical analyses show a positive correlation between the studied indicators. It is proved that this dependence is dynamic and changes over time. The obtained implicit indicators measure the influence of LLM technologies on the e-commerce domain. This influence is expected to grow, requiring further research.

Keywords: large language models, AI-detection, e-commerce, product performance.

INTRODUCTION

Since the release of the first version of ChatGPT on November 30, 2022, LLMs have become integral across numerous aspects of human activity. The capabilities of these models to search for information, serve as assistants, and analyze data have made them widely applicable in various sectors, including business and industry [1]. Particularly in e-commerce — a field where Natural Language Processing (NLP) techniques were already well-integrated before the advent of LLMs — these models have found applications at every stage of interaction among customers, sellers, and products. The introduction of LLMs has inevitably transformed e-commerce practices, significantly changing the industry. Given that the presence of LLMs in a business isn't always immediately apparent, the challenge of assessing their impact on e-commerce closely ties into the ability to discern whether textual data was generated by an LLM or not.

Perplexity per token is a key metric for assessing the predictive power of language models, including prominent transformer models like BERT and GPT-4, among other LLMs. This metric has been crucial for comparing different language models on the same dataset and fine-tuning hyperparameters, though it is sensitive to linguistic characteristics and sentence length [2]. Despite its central role in developing language models, perplexity has limitations. Notably, it does not reliably characterize speech recognition performance and may not effectively indicate overfitting and generalization capabilities [3; 4]. This has led to questioning the merit of solely focusing on perplexity optimization.

Furthermore, while perplexity is a common baseline for differentiating between machine-generated and human-generated text, it is often inadequate when

© *Publisher IASA at the Igor Sikorsky Kyiv Polytechnic Institute*, 2025

used alone, leading to a shift away from methods solely reliant on statistical signatures. Instead of relying solely on raw perplexity scores, a more nuanced approach involves comparing the perplexity measurement with cross-perplexity [5]. This method assesses how unexpected one model's next token predictions are to another, providing a more distinct separation between machine and human text than perplexity alone.

Thus, to investigate the impact of LLM technology on e-commerce, the following research questions are formulated:

RQ1: Do text perplexity-based statistical indicators and e-commerce product metrics correlate?

RQ2: Does the relationship between text perplexity-based statistical indicators and e-commerce product metrics evolves over time?

This research contributes to the understanding of LLMs' influence on e-commerce. The key contributions are as follows:

1. To the best of the author's knowledge, this study is among the first to assess the impact of LLM models on e-commerce, with the introduction of a unique approach and then using it on real-world data.

2. This paper explored the relationship between text perplexity-based statistical indicators and product metrics and found a positive correlation that, as verified by statistical techniques, appears to change over time.

The structure of this paper is organized as follows: Section 2 reviews related work, Section 3 describes the methodology, Section 4 details the experiments and results, and Section 5 concludes the paper and proposes directions for future research.

RELATED WORK

LLM in NLP. Recent advancements in NLP have been significantly shaped by (LLMs like GPT-2, GPT-3, and BERT, which have established new benchmarks in various NLP tasks due to their ability to produce coherent and human-like text [6; 7]. These models have demonstrated their effectiveness beyond benchmarks and have been successfully utilized in real-world applications such as automated customer support, conversational systems, and text summarization [8; 9].

More recently, advanced LLMs, including GPT-4 [10], Gemini [11], and Llama 2 [12], have shown remarkable proficiency in natural language processing tasks [1], information retrieval [13], and various other domains [14; 15].

NLP in e-commerce. NLP techniques have been extensively utilized in e-commerce for various tasks, including sentiment analysis, recommendation systems, and search engine optimization [16; 17]. Previous research has investigated using NLP to extract product attributes, create stylistic variations of product descriptions, and generate multilingual descriptions [18; 19]. Although these methods show promise, they have yet to achieve the scalability needed to produce high-quality, human-like results. While NLP applications in business settings are not a novel concept, there has been limited exploration into their tangible effects on revenue and customer engagement.

LLM in e-commerce. The integration of LLM technology into e-commerce has not only surpassed existing NLP solutions but has also been instrumental in addressing a broader range of challenges. Key applications of LLMs in this do-

main (Fig. 1) include advanced customer support, content generation (such as product descriptions, blog posts, comments, and reviews), content evaluation (including ratings and sensitivity analysis of user feedback), recommendation systems, and search engines [20].

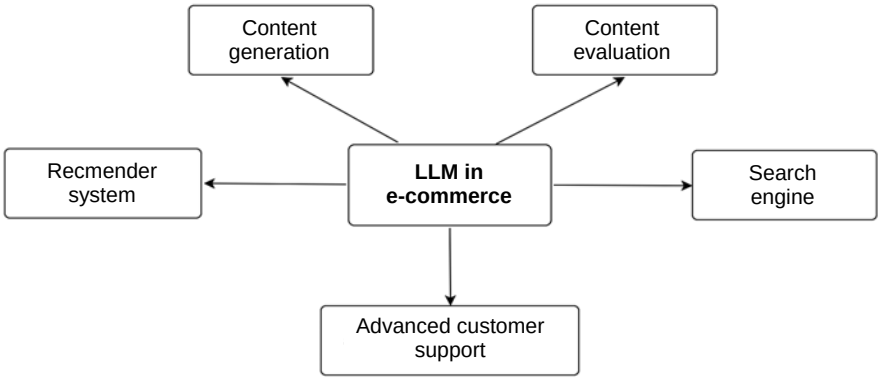


Fig. 1. Applications of LLMs in e-commerce

One notable trend is the fine-tuning of state-of-the-art LLMs for specific e-commerce tasks. For instance, LLMs created for automating product description generation enhance click-through rates and significantly reduce the manual effort required in content creation [21]. Similarly, employing LLMs for analyzing product reviews offers substantial benefits to e-commerce stakeholders — such as owners, managers, marketers, and data analysts — by providing quicker responses to customer feedback, thereby improving the overall effectiveness of e-commerce strategies [22]. In search engine optimization, LLMs are utilized for keyword selection and content enhancement [23].

Additionally, there is a growing trend towards developing families of LLM models tailored specifically for e-commerce applications. These models are not designed to be generalists across multiple domains but are specialized and optimized for e-commerce tasks, training exclusively on relevant data and targeting e-commerce metrics [24; 25]. Given the widespread adoption of LLMs in the e-commerce sector, exploring how this technology impacts the industry is crucial.

AI-generated text detection. Early efforts to detect machine-generated text have shown potential, particularly with models whose outputs are not convincingly human-like. However, the advent of transformer models for language generation [6; 7; 12; 26] has rendered many of these basic detection mechanisms ineffective. One strategy is to record [27] or watermark all generated text [28], but such preemptive measures require complete control over the generative models.

In response to the growing prevalence of machine-generated text, primarily through platforms like ChatGPT, a wave of research has focused on post-hoc detection methods. These approaches do not rely on cooperation from model developers. Detection methods can be broadly categorized into two types. The first involves training detection models, where a pre-trained language model is fine-tuned for the binary classification task of detecting machine-generated text [29–31]. Techniques such as adversarial training [32] and abstention [33] are also employed. Alternatively, instead of fine-tuning the entire model, a linear classifier can be applied to fixed learned features, allowing for the integration of commercial API outputs [34].

The second category includes methods based on statistical signatures characteristic of machine-generated text. These approaches typically require little or no training data and can be easily adapted to new model families [35]. Examples include

detectors based on perplexity [33; 36; 37], perplexity curvature [38], log-rank [39], intrinsic dimensionality of generated text [40], and n-gram analysis [41]. While this overview is not exhaustive, recent surveys can reveal further details [42–45].

From a theoretical standpoint, the main limitation of detection is that fully general-purpose language models, by definition, would be impossible to detect [46–48]. However, even models approaching this ideal may still be detectable with a sufficient number of samples [49]. In practice, the relative success of detection methods, including those proposed and analyzed in this work, provides evidence that current language models are still imperfect representations of human writing and, thus, detectable.

RESEARCH METHODOLOGY

The proposed methodology employs a specialized approach that examines the statistical properties of texts, particularly those that indicate the extent to which a text is machine-generated, and compares these with product metrics. The goal is to identify potential relationships between the two characteristics. This methodology is structured into three distinct stages (Fig. 2): 1) calculating the machine-generated characteristics of text features; 2) assessing the e-commerce product metrics; 3) conducting a statistical analysis to determine any significant correlations.

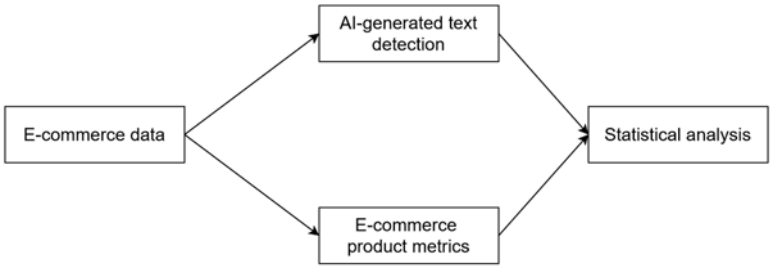


Fig. 2. Proposed methodology

AI-generated text detection. As described in related works, one of the approaches to detecting machine-generated text involves calculating specific statistical indicators of the texts and comparing them to predefined threshold values. This paper follows two critical conditions to choose a model for detecting machine-generated text. First, there is the absence of a training dataset to fine-tune classifiers for machine-generated text recognition. Second, there is no information on whether LLM models were used in generating the texts and, if so, which specific models. Therefore, a detection model that does not rely on training (zero-shot model) and is agnostic to any LLM model is required. The method, called Binoculars, meets these criteria and utilizes the binoculars score, which calculates the ratio of perplexity to cross-perplexity [5]:

$$B_{M_1,M_2}(s) = \frac{\log PPL_{M_1}(s)}{\log X - PPL_{M_1,M_2}(s)},$$

where perplexity, $\log PPL_{M_1}(s)$ is defined as the average negative log-likelihood of all tokens in the given sequence s cross-perplexity, $\log X - PPL_{M_1,M_2}(s)$, is defined as the average per-token cross-entropy between the outputs of two models, M_1 and M_2 when operating on the tokenization of the sequence s.

In other words, the numerator in this method is the perplexity, which quantifies how unexpected a string is to model M_1. Conversely, the denominator assesses how unexpected the token predictions of model M_2 are when evaluated by M_1. Intuitively, this means that a human is expected to diverge from M_1 more than M_2 could, assuming that the LLMs M_1 and M_2 re more similar to each other than they are to a human. This approach achieves state-of-the-art accuracy without requiring any training data. It can detect machine-generated text from various modern LLMs without needing model-specific adjustments. Therefore, this work utilizes the Binoculars score as a statistical signature for identifying machine-generated text.

E-commerce product metrics. Using an e-commerce dataset that captures interactions between customers and products, various metrics can be calculated to provide valuable insights into product performance and customer behaviour. Metrics related to sales and revenue include sales volume, revenue, conversion rate, and profit margin. Another category of metrics focuses on user experience, encompassing indicators such as product return rate, customer reviews, and ratings. The scope of product metrics is not confined to these examples; it is instead determined by the availability of specific features in the dataset that enable the calculation of particular metrics.

Statistical analysis. The third and final stage of the proposed methodology is a statistical comparison of machine-generated text characteristics and product metrics. Spearman's rank correlation coefficient is used to determine any relationship. It is a nonparametric measure of rank correlation that assesses how well the relationship between two features can be described using a monotonic function.

A bootstrap method is used to answer this research's second question and determine whether the relationship between the studied metrics has changed. It estimates the confidence intervals and significance of the difference between two Spearman coefficients. Bootstrapping can provide a flexible and robust way to handle non-parametric statistics without relying on normality assumptions. Bootstrap involves repeatedly resampling the data with replacement. For each bootstrap sample, the Spearman correlations for each of the two datasets are calculated, and the difference between them is computed. Then, the differences from all bootstrap samples (1000 samples in this work) are collected to form a distribution of differences and determine the confidence interval. A 95% confidence interval is used, which means the 2.5th percentile and the 97.5th percentile of the bootstrap differences are found. Suppose the 95% confidence interval does not include zero. In that case, it indicates a statistically significant difference between the two correlation coefficients and, therefore, a statistically significant change in the relationship between machine-generated text characteristics and product metrics. Otherwise, if the interval includes zero, no significant difference exists between the correlations at the chosen confidence level.

EXPERIMENTS

Dataset and Preprocessing. One of the challenges in researching the effects of LLM technology on e-commerce is the scarcity of accessible, complete, and up-to-date datasets. Given that ChatGPT was only released in November 2022, and considering the gradual integration of LLMs within the e-commerce sector, it will take some time to accumulate and publish comprehensive datasets.

The MerRec [50], introduced in March 2024, is one of the first datasets that meets these requirements. It encapsulates detailed records of user interactions on the Mercari e-commerce platform, tracking millions of users and products over six months, from May to October 2023. MerRec not only captures basic features such as user attributes (user_id, sequence_id, session_id) and product attributes (item_id, product_id) but also includes specialized data like timestamped action types, product taxonomy, and textual product descriptions, making it a rich resource for analysis.

This analysis focused on products listed during the dataset's initial (May) and final (October) months. Given limited computing resources and to minimize data skew from outliers or abnormal product behaviour, the data is preprocessed with specific criteria: only those products are selected whose names contain at least five words and are purchased at least once.

Generalized word shift graphs were utilized to enhance the clarity of product names analyzed in this research. Such visualizations provide a meaningful and interpretable summary of how individual words contribute to variations observed between two distinct text corpora [51].

For instance, the product names in the Women category for October 2023 were analyzed, featuring low ("AI-generated") and high ("human-generated") binoculars scores. Examples of top 20 product names with the lowest and the highest binoculars scores are presented in Table. Names scoring low on the binoculars scale exhibited higher standardization, including consistent word order, capitalization of each word, and numerical size descriptors. In contrast, names with high binoculars scores (likely human-generated) displayed a less structured word order, lacked punctuation and used words to describe sizes (e.g., small).

Top 20 product names in the Women category for October 2023 with the lowest and the highest binoculars scores

Top 20 product names with the lowest binoculars score	**Top 20 product names with the highest binoculars score**
Converse size 7.5 women's shoes womens ugg boots size 9	Sebek Zigvolt Acrylic Stud Earrings
Keychain Wallet, Wristlet, Bangle, Bracelet, ID Card Holder, Purse, Key Chain, G	FIGS rose joggers size Small Petite
Christian Louboutin Women Black Heels Shoes Size 8.5 (38.5)	J crew midi floral sun Dress
Vtg Sterling Silver 925 Hinged Bangle Bracelet	Motel Olivia faux leather biker jacket white
Polo Ralph Lauren Women's V-Neck T-Shirt - Size Medium - Navy Blue	She Darc sweatshirt! Size small
UGG Brookfield Brown Sheepskin Leather Boots Size 8.5	Grae Cove linen short sleeve waist tie pockets mini dress blush women's XL
Avatar: The Last Airbender Aang & Katara Mini Backpack	Hot topic rob zombie hoodie XS
Womens Old Navy Fleece Jacket Size Small	Famous magic land couples OS leggings
Nike air max 270 women size 7	August Silk womens colorful funky patterned Shorts
Old Navy Active Fleece Jacket	Kate spade Pitch Purrfect Piano Cross-body KC729 NWT
Lululemon Long Sleeve - Size 10	Beautiful Disaster Tribe Jacket Size L
Purple Hooded Long Sleeve Sweater	Express Low rise columnist pants
Tory Burch Black Leather Boots Size 10.5	New sweatshirt hoodie Jeffrey Star
Victoria's Secret PINK Bling Leggings	Hades Disneyland Spirit Jersey Small
The Nightmare Before Christmas Jack Skellington	Save for Rosemary Special love lot
Nike Air Max 2X (Women)	Hot Topic Mushroom Collar dress
Super cute and comfy pajamas	Coach Wyn Logo Plaque Small Wallet
Tommy Hilfiger Women's Medium Red and White Striped Dress	NEW bundle Victoria Secret underwear
Costume Jewelry Lot - 25 pieces - Necklaces, Bracelets, Earrings	Cat In a pumpkin earrings
	Waffle Debut Retro Sneaker leopard

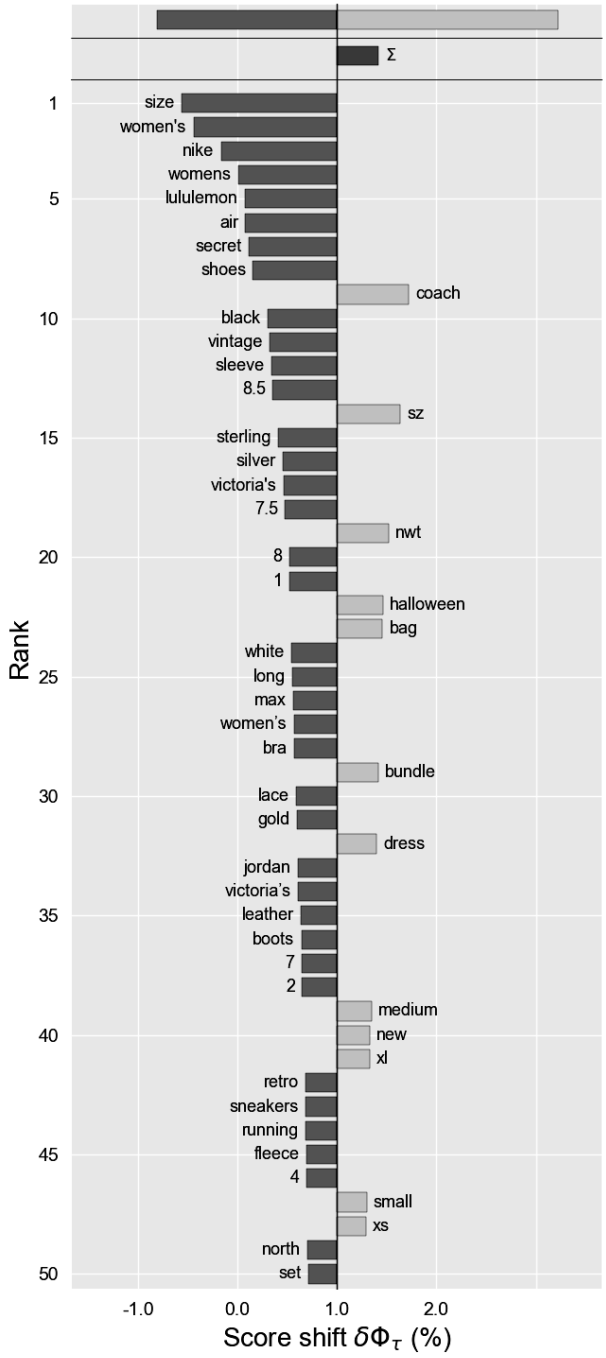


Fig. 3. Word shift graph of the product names with the lowest and highest binoculars scores

The examination using word shift graph (Fig. 3) revealed minimal significant differences in word usage between the two groups (the first group contains product names with the binoculars scores from first quartile (Q1) and the second group contains product names with the binoculars scores from fourth quartile (Q4). However, several subtle distinctions were noted. Primarily, descriptive words for sizes (e.g., small, medium, xs, xl) were used in "human-generated" names. In contrast, numerical representations (e.g., 7.5, 8, 8.5) were employed in "AI-generated" names, enhancing the accuracy and standardization of size descriptions. Additionally, abbreviations (e.g., sz, nwt) were often included in "human-generated" names. Thus, the example of the Women's product category demonstrates how product names with different binocular scores differ from each other.

LLM models. As described in section 3, the binoculars method is used as an AI-generated text detector, which requires 2 LLM models. Moreover, these models should provide access to the raw logits of all tokens in the given sequence to calculate the binoculars score. Unfortunately, most state-of-the-art LLM models (GPT-4, Claude-3, etc.) do not provide access to such logits. Therefore, the open-source LLM models are considered, and the Falcon-7b model and the Falcon-7b-instruct model are chosen, which are pretrained generative text models with 7 billion parameters and demonstrate high performance.

It was carried out on the remote resources of Google Colab and consumed approximately 10 hours of A100 GPU to generate the scores for nearly 300000 unique product names.

Evaluation metric. To investigate the impact of LLM on e-commerce (namely, on product names), and based on the features of the selected MerRec dataset (unique user actions), the conversation rate is used as one of the central business metrics in e-commerce that indicates product performance. It is defined as the ratio of the total number of customers who purchased the product compared to the total number of customers who interacted with it.

Results. The proposed methodology's performance is evaluated on the real-data MerRec dataset. Overall, a positive correlation between binoculars score and conversation rate is found, which differs depending on the product category. These results are inspected in more detail in the following.

RQ1 Do text perplexity-based statistical indicators and e-commerce product metrics correlate?

First, the conversation rate scores are calculated for all products sold in May 2023; then, for the same products, the binoculars scores of their names are calculated. After that, the Spearman correlation coefficient between these indicators is calculated, and it found that it differs significantly for products of different categories (Fig. 4). For example, for the Men and Kids categories, the correlation is the highest at 0.54 and 0.53, respectively, which indicates a moderate correlation degree. The correlation is somewhat lower, but also significant, for products in the Women category (0.28). There is a group of categories for which the correlation is positive but very weak (Sports & outdoors, Pet Supplies, Toys & Collectibles, Vintage & collectibles). There are also categories for which the correlation is practically absent, but what is important to note is that there are no products for which the Spearman correlation is negative (except Garden & outdoor).

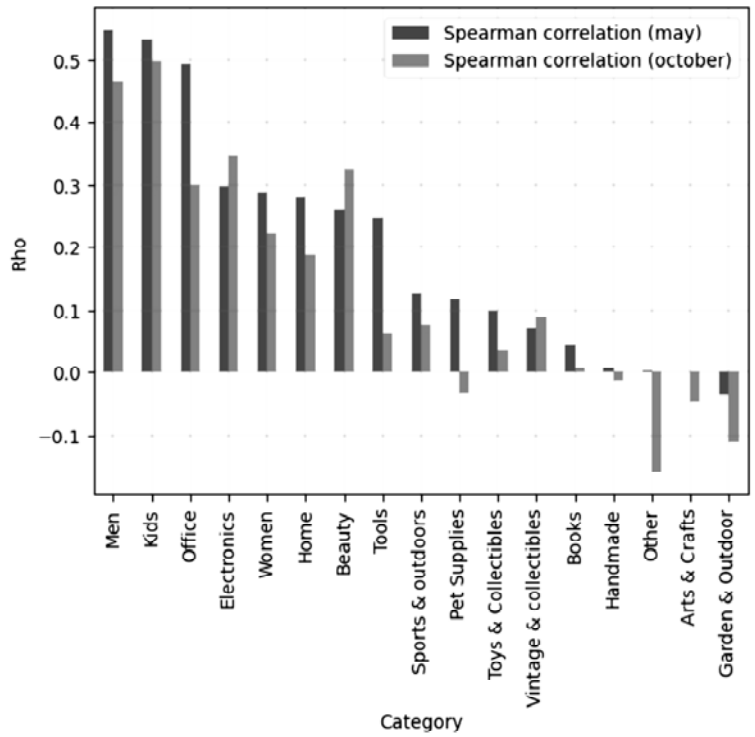


Fig. 4. Spearman correlation coefficients between binoculars score and conversation rate of products from the MerRec dataset

It is to noticed that the correlation is the largest for the most general groups (Men, Women, Kids, Office, Electronics), which are characterized by a wide range of products and their diversity, while products that can be attributed to a specific field of activity (Sports & outdoors, Pet Supplies, Vintage & collectibles, Handmade, etc.) have very weak or zero correlation. It can be assumed that for general categories, it is not easy to come up with an original product name that will distinguish it from others and interest customers; at the same time, for specific domain categories, the names of products may contain certain specifications, which will interpret them as original, which, however, is typical for them, and in no way distinguishes them from other products.

A similar analysis was conducted for products sold in October 2023. Similarly, a positive correlation between binoculars score and conversation rate is observed. However, for most categories, the correlation decreased; for a few, such as Other and Garden & outdoor, the correlation became negative, albeit very weak.

Thus, a moderate positive correlation between binoculars score (a text perplexity-based statistical indicator) and conversation rate (an e-commerce product metric) is seen. It can be interpreted that a higher probability of the product name being generated by a human (higher binoculars score) correlates with better product performance.

RQ2 Does the relationship between text perplexity-based statistical indicators and e-commerce product metrics evolves over time?

A statistical comparison of the correlation coefficients of the data for May and October is performed. It is found that for most categories, there is a statistically significant change in correlation (Fig. 5). Thus, finding a boxplot with

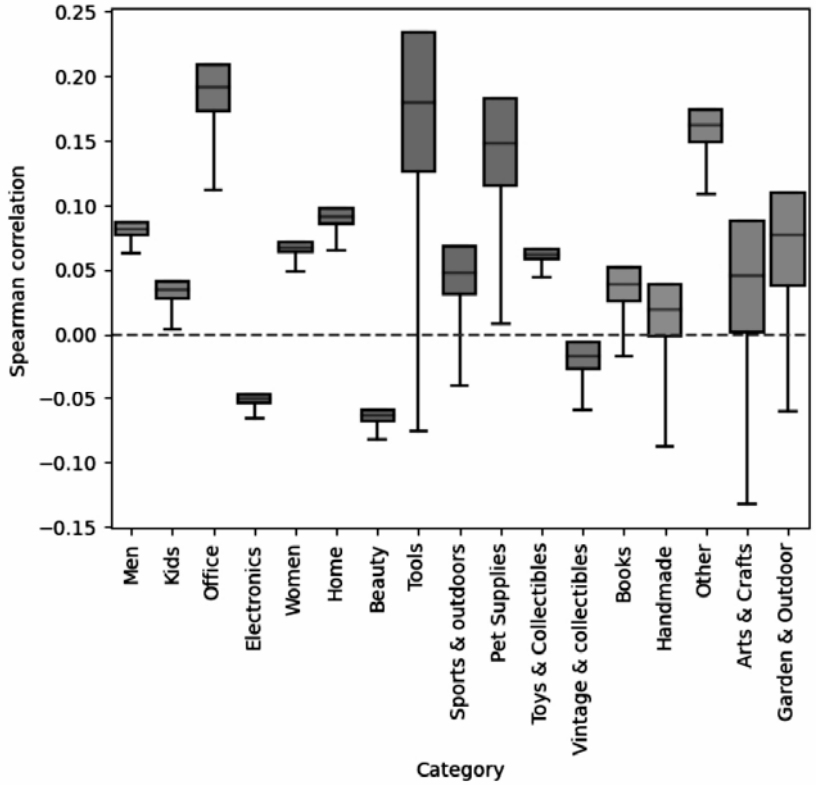


Fig. 5. Distribution of Spearman correlation coefficients across different product categories. Boxplots show median (red line) and 25- and 75-percentiles with whiskers ranging from 2.5- to 97.5-percentile

whiskers entirely above zero indicates a significant decrease in the correlation between binoculars score and conversation rate and placing it below zero, on the contrary, indicates an increase in correlation. It can be concluded that out of all 15 categories, for seven categories (including those with the highest correlation coefficients in May), the correlations decreased statistically; only for three categories increased, and for the rest of the categories, they remained unchanged (or their change is statistically insignificant).

So, for six months, from May to October, for most products, there is a trend to decrease the correlation coefficient between binoculars score (text perplexity-based statistical indicator) and conversation rate (e-commerce product metric). This may be due to the increased use of LLM technology to generate product names, but it is still small.

CONCLUSIONS

In this work, the methodology to determine the impact of AI-generated product names on e-commerce performance is proposed; namely, the relationship between the binoculars score of product names and the conversation rate of products is investigated. It examines in detail the current level of implementation of LLM technology in the field of e-commerce, considering a wide range of problems solved by language models. In addition, the existing state-of-the-art detection methods of machine-generated texts are described, and one of those methods that performs zero-shot and model-agnostic detection is used. Proposed approach is applied to real data for 2023 and a positive correlation between binoculars score (text perplexity-based statistical indicator) and conversation rate (e-commerce product metric) is found. This positive correlation tends to decrease, which is verified statistically. Thus, the impact of LLM technology on e-commerce is observed, and only an increase in this impact is expected in the future.

For future work, a semantic analysis of the comparison of product names over time on changing typical words in the product names triggered by the activity of LLM models can be conducted, which may be fascinating, but this is a question for further research.

REFERENCES

1. W.X. Zhao et al., “A survey of large language models,” *ArXiv*, 2023. doi: https://doi.org/10.48550/arXiv.2303.18223
2. A. Miaschi, D. Brunato, F. Dell’Orletta, and G. Venturi, “What makes my model perplexed? A linguistic investigation on neural language models perplexity,” in *Proceedings of Deep Learning Inside Out (DeeLIO): The 2nd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pp. 40–47, 2021. doi: 10.18653/v1/2021.deelio-1.5
3. D. Klakow, J. Peters, “Testing the correlation of word error rate and perplexity,” *Speech Communication*, vol. 38, no. 1–2, pp. 19–28, 2002. doi: 10.1016/S0167-6393(01)00041-3
4. S.F. Chen, D. Beeferman, and R. Rosenfeld, *Evaluation metrics for language models*. 1998. Available: https://kilthub.cmu.edu/articles/Evaluation_Metrics_For_Language_Models/6605324/files/12095765.pdf
5. A. Hans et al., “Spotting LLMs with binoculars: Zero-shot detection of machine-generated text,” *ArXiv*, 2024. doi: https://doi.org/10.48550/arXiv.2401.12070

6. A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019. Available: https://insightcivic.s3.us-east-1.amazonaws.com/language-models.pdf
7. T.B. Brown, "Language models are few-shot learners," *ArXiv*, 2020. doi: https://doi.org/10.48550/arXiv.2005.14165
8. D. Adiwardana et al., "Towards a human-like open-domain chatbot," *ArXiv*, 2020. doi: https://doi.org/10.48550/arXiv.2001.09977
9. P. Lewis et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9459–9474, 2020. Available: https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html
10. B. Miranda, A. Lee, S. Sundar, A. Casasola, and S. Koyejo, "Beyond Scale: The Diversity Coefficient as a Data Quality Metric for Variability in Natural Language Data," *ArXiv*, 2024. doi: https://doi.org/10.48550/arXiv.2306.13840
11. Gemini Team Google, "Gemini: a family of highly capable multimodal models," *ArXiv*, 2023. doi: https://doi.org/10.48550/arXiv.2312.11805
12. H. Touvron et al., "Llama 2: Open foundation and fine-tuned chat models," *ArXiv*, 2023. doi: https://doi.org/10.48550/arXiv.2307.09288
13. S.E. Spatharioti, D.M. Rothschild, D.G. Goldstein, and J.M. Hofman, "Comparing traditional and LLM-based search for consumer choice: A randomized experiment," *ArXiv*, 2023. doi: https://doi.org/10.48550/arXiv.2307.03744
14. S. Frieder, J. Berner, P. Petersen, and T. Lukasiewicz, "Large language models for mathematicians," *ArXiv*, 2023. doi: https://doi.org/10.48550/arXiv.2312.04556
15. F. Zeng, W. Gan, Y. Wang, N. Liu, and P.S. Yu, "Large language models for robotics: A survey," *ArXiv*, 2023. doi: https://doi.org/10.48550/arXiv.2311.07226
16. G. Sousa, "Natural Language Processing and its applications in e-business," *Cadernos de Investigação do Mestrado em Negócio Eletrónico*, vol. 2, 2022. doi: https://doi.org/10.56002/ceos.0070_cimne_1_2
17. Y. Huang, "Research on the Application of Natural Language Processing Technology in E-commerce," in *ISCTT 2021; 6th International Conference on Information Science, Computer Technology and Transportation, 2021*, pp. 1–5. Available: https://ieeexplore.ieee.org/abstract/document/9738909
18. M. Chen, Q. Tang, S. Wiseman, and K. Gimpel, "Controllable paraphrase generation with a syntactic exemplar," *ArXiv*, 2019. doi: https://doi.org/10.48550/arXiv.1906.00565
19. Q. Chen, J. Lin, Y. Zhang, H. Yang, J. Zhou, and J. Tang, "Towards knowledge-based personalized product description generation in e-commerce," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019*, pp. 3040–3050. doi: 10.1145/3292500.333072
20. Q. Ren et al., "A survey on fairness of large language models in e-commerce: progress, application, and challenge," *ArXiv*, 2024. doi: https://doi.org/10.48550/arXiv.2405.13025
21. J. Zhou, B. Liu, J.N.A.Y. Hong, K.-C. Lee, and M. Wen, "Leveraging Large Language Models for Enhanced Product Descriptions in eCommerce," *ArXiv*, 2023. doi: https://doi.org/10.48550/arXiv.2310.18357
22. K.I. Roumeliotis, N.D. Tselikas, and D.K. Nasiopoulos, "LLMs in e-commerce: a comparative analysis of GPT and LLaMA models in product review evaluation," *Natural Language Processing Journal*, vol. 6, p. 100056, 2024. doi: 10.1016/j.nlp.2024.100056
23. G. Chodak, K. Błażyczek, "Large Language Models for Search Engine Optimization in E-commerce," in *International Advanced Computing Conference*, pp. 333–344, 2023. doi: 10.1007/978-3-031-56700-1_27
24. B. Peng, X. Ling, Z. Chen, H. Sun, and X. Ning, "eCeLLM: Generalizing Large Language Models for E-commerce from Large-scale, High-quality Instruction Data," *ArXiv*, 2024. doi: https://doi.org/10.48550/arXiv.2402.08831

25. C. Herold, M. Kozielski, L. Ekimov, P. Petrushkov, P.-Y. Vandenbussche, and S. Khadivi, "LiLiuM: eBay's Large Language Models for e-commerce," *ArXiv*, 2024. doi: https://doi.org/10.48550/arXiv.2406.12023
26. A. Chowdhery et al., "Palm: Scaling language modeling with pathways," *Journal of Machine Learning Research*, vol. 24, no. 240, pp. 1–113, 2023. Available: http://jmlr.org/papers/v24/22-1144.html
27. K. Krishna, Y. Song, M. Karpinska, J. Wieting, and M. Iyyer, "Paraphrasing evades detectors of AI-generated text, but retrieval is an effective defense," *Advances in Neural Information Processing Systems*, vol. 36, 2024. Available: https://proceedings.neurips.cc/paper_files/paper/2023/hash/575c450013d0e99e4b0ecf82bd1afaa4- Abstract-Conference.html
28. J. Kirchenbauer, J. Geiping, Y. Wen, J. Katz, I. Miers, and T. Goldstein, "A watermark for large language models," in *International Conference on Machine Learning*, 2023, pp. 17061–17084. Available: https://proceedings.mlr.press/v202/ kirchenbauer23a.html
29. I. Solaiman et al., "Release strategies and the social impacts of language models," *ArXiv*, 2019. doi: https://doi.org/10.48550/arXiv.1908.09203
30. R. Zellers et al., "Defending against neural fake news," *Advances in Neural Information Processing Systems*, vol. 32, 2019. Available: https://proceedings.neurips.cc/paper/2019/hash/3e9f0fc9b2f89e043bc6233994dfcf76-Abstract.html
31. X. Yu et al., "GPT paternity test: GPT generated text detection with GPT genetic inheritance," *CoRR*, 2023. Available: https://arxiv.org/pdf/2305.12519v2
32. X. Hu, P.-Y. Chen, and T.-Y. Ho, "Radar: Robust AI-text detection via adversarial learning," *Advances in Neural Information Processing Systems*, vol. 36, pp. 15077–15095, 2023. Available: https://proceedings.neurips.cc/paper_files/paper/2023/hash/30e15e5941 ae0cdab7ef58cc8d59a4ca-Abstract-Conference.html
33. Y. Tian et al., "Multiscale positive-unlabeled detection of AI-generated texts," *ArXiv*, 2023. Available: https://arxiv.org/pdf/2305.18149
34. V. Verma, E. Fleisig, N. Tomlin, and D. Klein, "Ghostbuster: Detecting text ghostwritten by large language models," *ArXiv*, 2023. doi: https://doi.org/10.48550/arXiv.2305.15047
35. J. Pu, Z. Huang, Y. Xi, G. Chen, W. Chen, and R. Zhang, "Unraveling the mystery of artifacts in machine generated text," in *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pp. 6889–6898, 2022. Available: https://aclanthology.org/2022.lrec-1.744
36. C. Vasilatos, M. Alam, T. Rahwan, Y. Zaki, and M. Maniatakos, "HowkGPT: Investigating the detection of ChatGPT-generated university student homework through context-aware perplexity analysis," *ArXiv*, 2023. doi: https://doi.org/10.48550/arXiv.2305.18226
37. Y. Wang et al., "M4: Multi-generator, multi-domain, and multi-lingual black-box machine-generated text detection," *ArXiv*, 2023. doi: https://doi.org/10.48550/arXiv.2305.14902
38. E. Mitchell, Y. Lee, A. Khazatsky, C.D. Manning, and C. Finn, "DetectGPT: Zero-shot machine-generated text detection using probability curvature," in *International Conference on Machine Learning*, pp. 24950–24962, 2023. Available: https://proceedings.mlr.press/v202/mitchell23a.html
39. J. Su, T.Y. Zhuo, D. Wang, and P. Nakov, "DetectLLM: Leveraging log rank information for zero-shot detection of machine-generated text," *ArXiv*, 2023. doi: https://doi.org/10.48550/arXiv.2306.05540
40. E. Tulchinskii et al., "Intrinsic dimension estimation for robust detection of AI-generated texts," *Advances in Neural Information Processing Systems*, vol. 36, 2024. Available: https://proceedings.neurips.cc/paper_files/paper/2023/hash/7baa48bc166aa2013d78cbdc15010530-Abstract-Conference.html
41. X. Yang, W. Cheng, Y. Wu, L. Petzold, W.Y. Wang, and H. Chen, "DNA-GPT: Divergent N-Gram Analysis for Training-Free Detection of GPT-Generated Text," *ArXiv*, 2023. doi: https://doi.org/10.48550/arXiv.2305.17359

42. S.S. Ghosal, S. Chakraborty, J. Geiping, F. Huang, D. Manocha, and A.S. Bedi, "Towards possibilities & impossibilities of AI-generated text detection: a survey," *ArXiv*, 2023. doi: https://doi.org/10.48550/arXiv.2310.15264
43. R. Tang, Y.-N. Chuang, and X. Hu, "The science of detecting LLM-generated text," *Communications of the ACM*, vol. 67, issue 4, pp. 50–59, 2024. doi: 10.1145/3624725
44. M. Dhaini, W. Poelman, and E. Erdogan, "Detecting ChatGPT: A survey of the state of detecting ChatGPT-generated text," *ArXiv*, 2023. doi: https://doi.org/10.48550/arXiv.2309.07689
45. B. Guo et al., "How close is ChatGPT to human experts? Comparison corpus, evaluation, and detection," *ArXiv*, 2023. doi: https://doi.org/10.48550/arXiv.2301.07597
46. R. Varshney, N.S. Keskar, and R. Socher, "Limits of detecting text generated by large-scale language models," in *2020 Information Theory and Applications Workshop (ITA), 2020*, pp. 1–5. doi: 10.1109/ITA50056.2020.9245012
47. H. Helm, C.E. Priebe, and W. Yang, "A Statistical Turing Test for Generative Models," *ArXiv*, 2023. doi: https://doi.org/10.48550/arXiv.2309.08913
48. V.S. Sadasivan, A. Kumar, S. Balasubramanian, W. Wang, and S. Feizi, "Can AI-generated text be reliably detected?," *ArXiv*, 2023. doi: https://doi.org/10.48550/arXiv.2303.11156
49. S. Chakraborty, A.S. Bedi, S. Zhu, B. An, D. Manocha, and F. Huang, "On the possibilities of AI-generated text detection," *ArXiv*, 2023. doi: https://doi.org/10.48550/arXiv.2304.04736
50. L. Li, Z. A. Din, Z. Tan, S. London, T. Chen, and A. Daptardar, "MerRec: A Large-scale Multipurpose Mercari Dataset for Consumer-to-Consumer Recommendation Systems," *ArXiv*, 2024. doi: https://doi.org/10.48550/arXiv.2402.14230
51. R. J. Gallagher et al., "Generalized word shift graphs: a method for visualizing and explaining pairwise comparisons between texts," *EPJ Data Science*, vol. 10, no. 1, Jan. 2021. doi: 10.1140/epjds/s13688-021-00260-3

Received 02.09.2024

INFORMATION ON THE ARTICLE

Oleksandr S. Bratus, ORCID: 0009-0003-5004-1652, Educational and Research Institute for Applied System Analysis of the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Ukraine, e-mail: olexandr.bratus@gmail.com

ОЦІНЮВАННЯ ВПЛИВУ НАЗВ ПРОДУКТІВ, СТВОРЕНИХ ШТУЧНИМ ІНТЕЛЕКТОМ, НА ЕФЕКТИВНІСТЬ ЕЛЕКТРОННОЇ КОМЕРЦІЇ / О.С. Братусь

Анотація. Досліджено вплив великих мовних моделей (LLM) на електронну комерцію. Здійснено детальний огляд поточного рівня впровадження LLM у електронній комерції. Проаналізовано існуючі підходи до детекції текстів, згенерованих штучним інтелектом (ШІ), та визначено обмеження їх застосування. Запропоновано методологію визначення впливу LLM на електронну комерцію на основі порівняння індикаторів ШІ-згенерованих текстів та продуктових метрик. Продемонстровано застосування методології на реальних даних, що зібрані після релізу ChatGPT, і отримано результати статистичного аналізу, які показують додатну кореляцію між досліджуваними показниками. Підтверджено наявність динаміки цієї залежності та її зміни з часом. Отримані неявні індикатори вимірюють вплив LLM технології на сферу електронної комерції. Очікуємо, що вплив зростатиме, потребуючи подальших досліджень.

Ключові слова: великі мовні моделі, ШІ-детекція, електронна комерція, ефективність продукту.

ВІДОМОСТІ ПРО АВТОРІВ

Андросов Дмитро Васильович,
аспірант кафедри штучного інтелекту ННІПСА КПІ ім. Ігоря Сікорського, Україна, Київ

Бодянський Євгеній Володимирович,
професор, доктор технічних наук, професор кафедри штучного інтелекту Харківського національного університету радіоелектроніки, Україна, Харків

Братусь Олександр Сергійович,
аспірант кафедри математичних методів системного аналізу ННІПСА КПІ ім. Ігоря Сікорського, Україна, Київ

Виклюк Ярослав Ігорович,
професор, доктор технічних наук, професор кафедри штучного інтелекту Національного університету «Львівська політехніка», Україна, Львів

Городецький Віктор Георгійович,
доцент, кандидат фізико-математичних наук, доцент кафедри автоматизації електротехнічних та мехатронних комплексів Навчально-наукового інституту енергозбереження та енергоменеджменту КПІ ім. Ігоря Сікорського, Україна, Київ

Зайченко Олена Юріївна,
професор, доктор технічних наук, професор кафедри математичних методів системного аналізу ННІПСА КПІ ім. Ігоря Сікорського, Україна, Київ

Зайченко Юрій Петрович,
професор, доктор технічних наук, професор кафедри математичних методів системного аналізу ННІПСА КПІ ім. Ігоря Сікорського, Україна, Київ

Корбан Дмитро Вікторович,
доцент, кандидат технічних наук, доцент кафедри управління судном Національного університету «Одеська морська академія», Україна, Одеса

Котенко Олег Васильович,
старший викладач кафедри безпеки життєдіяльності, екології та хімії Одеського національного морського університету, Україна, Одеса

Кузьменко Олексій Віталійович,
аспірант кафедри системного проектування ННІПСА КПІ ім. Ігоря Сікорського, Україна, Київ

Курдюк Сергій Вікторович,
доктор філософії, старший науковий співробітник Інституту Військово-Морських Сил Національного університету «Одеська морська академія», Україна, Одеса

Мартьянов Дмитро Ігорович,
асистент кафедри штучного інтелекту Національного університету «Львівська політехніка», Україна, Львів

Мельник Ігор Віталійович,
професор, доктор технічних наук, професор кафедри електронних пристроїв та систем факультету електроніки КПІ ім. Ігоря Сікорського, Україна, Київ

Мельник Олексій Миколайович,
професор, доктор технічних наук, професор кафедри судноводіння і морської безпеки Одеського національного морського університету, Україна, Одеса

Невінський Денис Володимирович,
кандидат технічних наук, доцент кафедри електронних засобів інформаційно-комп'ютерних технологій Інституту телекомунікацій, радіоелектроніки та електронної техніки Національного університету «Львівська політехніка», Україна, Львів

Недашківська Надія Іванівна,
професор, доктор технічних наук, доцент кафедри кафедри математичних методів системного аналізу ННІПСА КПІ ім. Ігоря Сікорського, Україна, Київ

Онищенко Олег Анатолійович,
професор, доктор технічних наук, професор кафедри управління судном Національного університету «Одеська морська академія», Україна, Одеса

Очеретна Валентина Валеріївна,
доцент, кандидат технічних наук, доцент кафедри судноводіння і морської безпеки Одеського національного морського університету, Україна, Одеса

Петренко Анатолій Іванович,
професор, доктор технічних наук, професор кафедри системного проектування ННІПСА КПІ ім. Ігоря Сікорського, Україна, Київ

Попов Андрій Юрійович,
аспірант кафедри математичних методів системного аналізу ННІПСА КПІ ім. Ігоря Сікорського, Україна, Київ

Починок Аліна Володимирівна,
кандидат технічних наук, старший науковий співробітник Науково-дослідного інституту електроніки та мікросистемної техніки КПІ ім. Ігоря Сікорського, Україна, Київ

Романюк Вадим Васильович,
професор, доктор технічних наук, професор кафедри економічної кібернетики та інформаційних систем Вінницького торговельно-економічного інституту Державного торговельно-економічного університету, Україна, Вінниця

Сем'янів Ігор Олександрович,
доцент, кандидат медичних наук, доцент кафедри фтизіатрії та пульмонології Буковинського державного медичного університету, Україна, Чернівці

Скрипка Михайло Юрійович,
аспірант кафедри електронних пристроїв та систем факультету електроніки КПІ ім. Ігоря Сікорського, Україна, Київ

Ткачук Андрій Васильович,
старший викладач кафедри фізики та електротехніки Хмельницького національного університету, Україна, Хмельницький

Ткачук Ганна Сергіївна,
доцент, кандидат технічних наук, доцент кафедри хімії та хімічної інженерії Хмельницького національного університету, Україна, Хмельницький

Щербина Ольга Василівна,
доцент, кандидат технічних наук, доцент кафедри експлуатації флоту і технології морських перевезень Одеського національного морського університету, Україна, Одеса