UDC 004.942(045)

BUILDING A SCALABLE DATASET FOR FRIDAY SERMONS OF AUDIO AND TEXT (SAT)

Samah A. A. – Postgraduate student of Department of Information Systems, Faculty of Computing and Information Technology, and Lecturer of Department of Management Information Systems, Faculty of Economics and Administration, King Abdul Aziz University, Jeddah, Mecca, Saudi Arabia.

Dimah H. A. – PhD, Associate Professor, Associate Professor of Department of Information Systems, Faculty of Computing and Information Technology, King Abdul Aziz University, Jeddah, Mecca, Saudi Arabia.

Hassanin M. A. – Dr. Sc., Professor, Professor of Department of Information Technology, Faculty of Computing and Information Technology, King Abdul Aziz University, Jeddah, Mecca, Saudi Arabia.

ABSTRACT

Context. Today, collecting and creating datasets in various sectors has become increasingly prevalent. Despite this widespread data production, a gap still exists in specialized domains, particularly in the Islamic Friday Sermons (IFS) domain. It is rich with theological, cultural, and linguistic studies that are relevant to Arab and Muslim countries, not just religious discourses.

Objective. The goal of this research is to bridge this lack by introducing a comprehensive Sermon Audio and Text (SAT) dataset with its metadata. It seeks to provide an extensive resource for religion, linguistics, and sociology studies. Moreover, it aims to support advancements in Artificial Intelligence (AI), such as Natural Language Processing and Speech Recognition technologies.

Method. The development of the SAT dataset was conducted through four distinct phases: planning, creation and processing, measurement, and deployment. The SAT dataset contains a collection of 21,253 audio and corresponding transcript files that were successfully created. Advanced audio processing techniques were used to enhance speech recognition and provide a dataset that is suitable for wide-range use.

Results. The fine-tuned SAT dataset achieved a 5.13% Word Error Rate (WER), indicating a significant improvement in accuracy compared to the baseline model of Microsoft Azure Speech. This achievement indicates the dataset's quality and the employed processing techniques' effectiveness. In light of this, a novel Closest Matching Phrase (CMP) algorithm was developed to enhance the high confidence of equivalent speech-to-text by adjusting lower ratio phrases.

Conclusions. This research contributes significant impact and insight into different studies, such as religion, linguistics, and sociology, providing invaluable insights and resources. In addition, it is demonstrating its potential in Artificial Intelligence (AI) and supporting its applications. In future research, we will focus on enriching this dataset expansion by adding a sign language video corpus, using advanced alignment techniques. It will support ongoing Machine Translation (MT) developments for a broader understanding of Islamic Friday Sermons across different linguistics and cultures.

KEYWORDS: Friday Sermons, Khutbah, Arabic speech recognition, Audio and text dataset, Machine translation.

ABBREVIATIONS

AI is an Artificial Intelligent;
ArSL is Arabic Sign Language;
ASR is Automatic Speech Recognition;
CMP is a Closest Matching Phrase;
P1 is a Name of Preacher;
P2 is an Age of Preacher;
P3 is an Original Country of Preacher;
P4 is an Academic Qualification of Preacher;
P5 is a Years of Experience of Preacher;
DL is a Deep Learning;
IFS is an Islamic Friday Sermons;
ML is a Machine Learning;
MT is a Machine Translation;
NLP is a Natural Language Processing;
PCM is a Pulse Code Modulation;
SAT is a Friday Sermon Audio and Text
S6 is a Title of Sermon;
S7 is a Type of Sermon (topic);
S8 is a Duration of Sermon;
S9 is a Date of Sermon;
S10 is a Place of Sermon;
S11: Language of Sermon,
S12 is other languages of Sermon translated into;
S13 is other sign languages used for translating Sermon;
S14 is a Language complexity of Sermon;
S15 is a Reliability of Manarat Al-Haramain Website;
S16 is a Reliability of AL-Khutaba Forum Website.
SL is a Sign Language;
WER is Word Error Rate.

NOMENCLATURE

A_i is an Audio recording to the *i*th item;
FT is a full text of one Friday Sermon;
k is representing the number of raters or judges;
N is total number of pairs in the dataset;
n is a number of observation (items) or cases being assessed;
R is a similarity between transcript and current phrase using Sequence Matcher (ratio);
r is a Pearson correlation coefficient;
SS is a sum of squares for total ranks;
T_i is a corresponding transcript to the *i*th item;
W is Kendall's Coefficient of Concordance;
X and *Y* are indicating the variables;
$\bar{x}$ and $\bar{y}$ are indicating the means of the two variables;

DOI 10.15588/1607-3274-2024-2-10

x_i is a rank or score given to the *i*th item by raters;

$\bar{x}$ is a mean (average) rank of all items assessed.

INTRODUCTION

In the past few years, large-scale datasets have become an essential step in applying artificial intelligence (AI) technologies, such as machine learning (ML) and deep learning (DL), in various sectors. These datasets can be created or collected from different types of data, which could be text, audio, video, or pictures. Each of these types of datasets has different ways of annotating, processing, and analyzing it, in order to develop or enhance the system. Overall, it supports decision-making in a specific sector. Thus, the task of collecting and creating a dataset, usually, requires a huge extensive effort from researchers in order to reach the expanded dataset in a certain domain [1,2].

In Arab countries, many researchers conducted their efforts to create and collect a huge Arabic dataset that serves many fields, such as education and healthcare, where they used AI technologies [3–5]. However, some fields like religion did not receive more attention from researchers, especially, in creating and collecting a dataset of sign language (SL) which is considered as a main unified communication language used by deaf communities [6].

Generally, in the religious domain, Islamic Friday Sermons (IFS), which are a key aspect of religious practice delivered during congregational prayers on Fridays, remain understudied. A few researches have been introduced in analyzing and understanding religious texts that are related to Sermons. Their focus was on the linguistic perspective (rhetorical structure of the Sermon), specifically, from pragmatics and discourse analysis aspects [7–9]. This ISF is a rich source of theological, cultural, and linguistic knowledge. Due to that, the aim of this research is to create a comprehensive Sermon dataset, a beneficial resource for researchers working with Islamic Sermons.

Despite increasing interest in this type of scientific research, leading to the development of various Natural Language Processing (NLP) and ML applications [10], these works still have limitations in scope and are not suited for large-scale computational analysis. The reason behind that is a lack of a Sermon dataset that has volume, value, variety, and metadata availability, a gap that this study aims to address.

The Object of Study is the process of collection, creation, and analysis of a comprehensive large-scale Friday Sermon dataset including audio, and text. This process includes creating a dataset and an algorithm implemented to enhance the recognition.

The subject of study is a methodology for creating a dataset of Friday sermons and identifying the type of dataset (audio, text, Sign Language (SL) videos). Another subject is identifying significant parameters that need to be considered from the Friday sermons presenter (Preacher) and the Friday sermons content in the collected dataset. In addition, the way of evaluating this created Friday sermons dataset.

The purpose of the work is to create, collect, and evaluate that aimed at enhancing language processing and recognition technologies. Moreover, this work was conducted to fill the existing gap in large-scale computational analysis of Friday Sermons by providing a rich dataset that has volume, value, variety, and accessible metadata. In addition, it supports the fields of Islamic Studies, Social Sciences, Linguistics, and AI with a useful resource.

The Islamic Friday Sermon (IFS) which is called in Arabic (Khutba AL-Jummah or Friday Khutbah) is a formal religious speech introduced on each Friday of the week. In Islam, Friday is considered the greatest day for Muslims to prepare themselves by praying in the mosque and listening to the Sermon [11].

Some researchers indicate that the IFS have a significant influence on humans' beliefs, attitudes, and behaviors. Also, it can influence the religious and cultural identity of Muslim communities. It can solve some issues in communities, like social and political issues such as inequality, injustice, and discrimination. Moreover, the Friday Sermon may play an important role in shaping national identity. From this standpoint, we can consider the Friday sermon data as valuable data that deserves study to under-stand the nature of its impact on different societies. In addition to the possibility of benefiting from the impact of Friday sermons on strengthening national identity, consolidating beliefs, controlling the behavior of community members, and directing them in the right direction [12].

These Islamic Friday Sermons will be composed and re-viewed based on the selected topic by the Preacher (Presenter of Friday Sermon), which is a person who delivers a Sermon to the congregation. The topic that was selected can be related to religion, community issues, morality, con-temporary challenges...etc. [13]. One of the researchers mentioned, generally, religious sermons are divided into four main types. First is religious education for the public. Second is proving faith in the souls. Third is correction of faults and prohibition of evils. Fourth is invitation to Islam or its defending [14].

In general, the speech of the Sermon on this greatest day should be introduced by the Preacher in a clear and interesting manner using understandable vocabulary. Mainly, the IFS duration without the Azan and prayer is around 30 to 40 minutes and consists of two Sermons where there is a short silence around 1 to 3 minutes between them. Usually, the first Sermon is longer than the second. There is Azan before the first Sermon and at the end of the second Sermon, there is prayer. These two parts are called the beginning and closing parts of the sermon (Sermon Prayer) included regularly in the structure of a Sermon. Whereas, the two Sermons that are in the middle are the body of the Sermon [9, 15], as shown in Fig. 1.

DOI 10.15588/1607-3274-2024-2-10

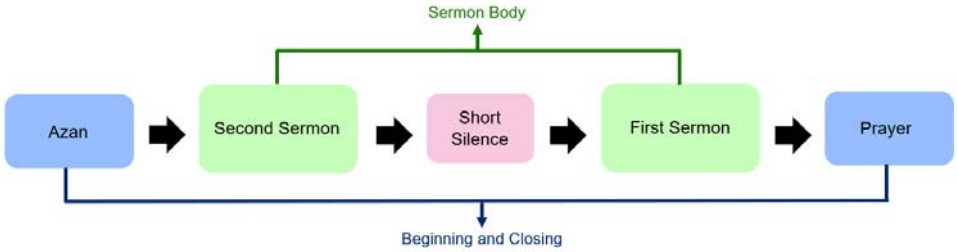


Figure 1 – Block diagram of Islamic Friday Sermon structure

Typically, the first Sermon contains the following: 1) Hamd Allah or Praise (thanks Allah). 2) Salawat, and Salam for the prophet Mohammed (Peace Be Upon Him). 3) Discussion of the main content or theme (topic) and reading some Ayat from AL-Qur'an (Recitation of Quranic Verses). 4) Advice and reminders with supplication which is called in Arabic (Dua'a). It is calling upon Allah (God) with respect and faithful for different reasons such as forgiveness, helping, and safeguarding the country and its leaders. However, the second Sermon started once again by Hamd Allah or Praising and Salawat, and Salam for the prophet Mohammed (Peace Be Upon Him). Then, the Preacher will continue the same topic that began in the first Sermon with more highlighting of more important points in the topic. Moreover, concentrates on reminding the congregation of certain Islamic obligations, virtues, or main issues related to the community. Finally, it will be ending also by supplication to all Muslim communities around the world and reminding them by doing what Allah said in order to achieve good deeds [16].

1 PROBLEM STATEMENT

The problem at hand is the lack of a scalable dataset for Friday sermons in both audio and text formats, which hinders the development and evaluation of automated systems for analyzing and understanding sermon content. The absence of such a dataset limits the advancements in NLP and speech recognition research specifically tailored for sermon analysis and related applications.

The current state of available datasets for Friday sermons is either limited in size, restricted to specific languages or regions, or lacks the necessary annotations for comprehensive analysis. This scarcity prevents researchers and developers from effectively training and evaluating machine learning models and algorithms for tasks such as sentiment analysis, topic extraction, speaker identification, or content summarization within the context of sermon texts and audio recordings.

Furthermore, the complexity of these Friday Sermons, besides the intense need for accurate dataset representation in digital form (audio, text), requires a dataset that is not only extensive in volume and variety but also rich in metadata to support computational analyses. So, we can find that were created dataset through this study includes a scalable dataset for Friday sermons in both audio and text formats, encompassing diverse languages, regions, and religious denominations. This dataset has been annotated with relevant metadata such as speaker information, sermon topic, date, and location, enabling researchers and developers to explore various aspects of sermon content using both NLP and speech recognition techniques.

Thus, the created dataset presented as:

$$\text{SAT} = \{(A_i, T_i)\}_{i=1}^{N},$$

where the SAT represents our created Sermon Audio and Text dataset, A_i indicates audio recording, T_i indicates the corresponding transcript, and N represents the total number of pairs in the dataset.

Moreover, for enhancing the accuracy of recognition by utilizing our SAT dataset, a similarity matching algorithm for finding the Closest Matching Phrase (CMP) between the transcript T and the full-text FT was used. Therefore, we can say that using our created SAT dataset which includes (Audio and Text) in any customized speech recognition application (tunned) will enhance the accuracy by reducing the WER of ASR as output for any Speech recognition system, as shown in the formula:

$$\text{SAT}=\{(A_i,T_i)\}_{i=1}^{N} \rightarrow \text{ASR}(\text{WER}_{after\_tuning} < \text{WER}_{before\_tuning}).$$

To identify the Closest Matching Phrase CMP within FT that most closely matches T we defined a function using similarity measure R, which is employed as:

$$f(T, FT) \rightarrow \text{CMP},$$

where the similarity measure R is obtained from the Sequence Matcher algorithm which is the ratio of similarity between T and the current phrase being evaluated within FT. Also, we defined the minimum ratio minR which means if the value R is less than the minR that means can not be accepted to be similar.

Therefore, the minR ≤ R ≤ maxR, where minR=0.50 and maxR is the maximum achievable ratio, ensures R falls within this range to be considered a valid match.

Hence, the main challenge of this work seeks to address a large-scale dataset and comprehensive metadata for the Friday Sermon Audio and Text (SAT) dataset. In addition, a novelty algorithm is implemented to enhance the recognition. Overall, this work tackles the existing gap in the analysis and processing of Islamic Friday Sermons.

DOI 10.15588/1607-3274-2024-2-10

2 REVIEW OF THE LITERATURE

In terms of the linguistic studies field, some study efforts have focused on different aspects of the IFS. One of these studies used 65 texts of the Yemeni-Arab Sermon to study the usage of deixis analysis. This deixis analysis helps people understand the meaning behind certain sentences based on their context. In general, deixis is divided into five types: they are person deixis, place deixis, time deixis, social deixis, and discourse deixis. Through this study, the researchers focused on studying deixis analysis from pragmatic and discourse perspectives. They had a limitation in using a small dataset of Sermons that needed to be translated into English for conducting their experiments [17]. Similarly, a study [9] used deixis in the English Islamic Friday Sermon using 70 texts from the English Friday Sermon dataset from multiple online sources. This study ended by acknowledging the small size of the dataset as a limitation. Another study was conducted based on the interpersonal model of metadiscourse for analyzing 30 text and speech English Friday Sermon datasets that were collected from various online sources. Also, they highlighted the limitations of the Sermon dataset [15], [18]. A study [19] focused on directive speech acts performed in the Sermon using the 56 Sermon dataset from the Islamic Religious Council of Singapore. They found that Friday Sermons use different strategies of directive speech acts.

In addition, one of the studies focused on the phoneme distribution in Malay Friday Sermon derived from 52 speech transcripts that are available on a government website. They reached the same limitation of having a small number of words collected and analyzed [7].

In terms of sociolinguistics and discourse analysis, two of the studies focused on the Sermon's duration. The first study of [11] conducted an analysis of Friday Sermon duration. They found that a shorter Sermon may be indicative of the Preacher's expertise in religious affairs. The second study used a descriptive method (questionnaire) in order to assess the congregation's understanding of the Friday Sermon discourse. The result of their study was that most congregations preferred Sermons with a duration of 15–20 minutes [20]. In studying the content and thematic analysis, [21] carried out a content analysis of Friday Sermons by the Turkish-Islamic Union for Religious Affairs in Germany, integrating sociolinguistics and discourse analysis. However, this study was limited to local text Friday Sermons that may not have received more attention from all Muslims around the world. Their dataset was 481 that were obtained from 2011 to 2019 on the DİTİB website. Another study conducted a thematic analysis of the Friday Sermon in Negeri Sembilan. They highlighted the importance of selecting topics that engage the congregations while considering their cultural background and educational level. However, their limitation was that the study was confined to Sermons from one region [22].

Other studies have employed a multidisciplinary approach to scrutinize the Sermon. One of the studies used ML techniques to evaluate the impact of Turkey's Friday Sermons on Twitter users. However, this study was focused on examining only one Sermon feature, which is the topics that are handled in Sermons [10].

Based on the illustrated previous studies, we can conclude that there are a few researches that have been introduced to analyzing and understanding religious texts that are related to Sermons. Their focus was on the rhetorical structure of the Sermon, specifically pragmatics and discourse analysis aspects, by utilizing a limited speech and text Sermon dataset. These types of scientific research have gained significant attention among researchers and opened avenues for the development of various NLP and ML applications for studying more parameters of the Sermon dataset. For example, themes (topic or domain), title, duration, date of the Sermon, location (place of the Sermon), and language of the Sermon … etc. Also, from some studies, we found that we need to be aware of the Preacher's parameters, whereas a study [7] emphasized the importance of the Preacher's expertise in religious affairs in conveying the concept of the Sermon to the congregations in a short duration. Thus, we can highlight some of Preacher's parameters, such as their years of experience, their original country, and so on.

Still, these works have limitations and are not appropriate for large-scale computational analysis. The reason behind that, from our perspective, is a lack of the Sermon dataset and its metadata availability, which is a gap that this study aims to address. In our study, we are going to create a dataset of Sermons that contain Arabic speech and text.

3 MATERIALS AND METHODS

The collection and creation of our dataset followed a structured, four main phase approach. It is designed to ensure the dataset's integrity, relevance, and utility. Each phase contained specific stages (steps) that should be successfully finished to move on to the next step in the next phase.

The nine stages are illustrated in Fig. 2. Each phase and its stages will be explained in more detail.

4.1 Planning Phase: this phase includes three main stages, which are: A) Design and implement a questionnaire. B) Analyze the questioner. C) Identify the parameters of data collection. The explanation of these stages is as follows:

A) Designing and Implement Questionnaire: we used a questionnaire in order to ask the specialists in data science about the important parameters that should be included in our data and metadata. It was designed in three main parts in accordance with the axes of the questionnaire:

1) Personal Data. 2) Data for Preacher (Presenter of Friday sermon). 3) Data for Friday sermon. Each part was written and designed to collect specific data related to this study's objectives (see in the appendix Fig. A1, Fig. A2, Fig. A3, and Fig. A4).

It was distributed electronically using a Google Form. We used expert ratings for multiple parameters of Preacher and Sermon.

DOI 10.15588/1607-3274-2024-2-10

The response of (n = 50) was obtained by 28 males and 22 females. The parameters encompassed characteristics of Preacher and Sermon, with five related to Preacher and eleven related to Sermon. Each expert has rated each parameter's im-portance based on (large, medium, and little).

B) Analyze Questionnaire: our analysis of the questionnaire was conducted based on the following: (1) Describing the collected data from expert evaluations for each parameter. (2) Finding correlations between experts' evaluations of each parameter. (3) Measuring the agreement between experts' evaluations for each parameter. These steps will support decision-making about the important parameters that should be considered in creating Fraidy sermon data and its metadata.

(1) Description Analysis of Parameters (Based on Expert Evaluation): in order to analyze the expert evaluation data for identifying which parameters are important and needed to be included in our created dataset, we convert them on a scale from 1 (little importance) to 3 (large importance). Also, we add notation for each parameter. The descriptive statistics for each parameter (Preacher and Sermon), including the mean, median, and standard deviation, are shown in Table 1.

These statistics provide insights into the perceived importance of each parameter.

In Preacher parameters, the Academic Qualification P4, "Years of Experience" P5, and "Name of Preacher" P1 had a slightly high mean rating of 2.54, 2.42, and 2.34 respectively, which means these three parameters are significantly important. Conversely, the "Age of Preacher" P2 had a lower mean rating of (x =1.46), implying less perceived importance.

In Sermon parameters, the "Title of Sermon" S6 obtained (x= 2.86) mean rating, which indicates it is significantly important. Also, it received the highest median rating of 3.0, which means this parameter is important from most experts' perspectives. By looking at the standard deviation for these ratings, we can see the level of consensus or disagreement among the experts.

The parameters with a lower standard deviation indicate a greater consensus among experts regarding their importance. For example, "Title of Sermon" S6 showed the least standard deviation, which refers to a strong agreement among experts on its significance.

Moreover, the "Reliability of Manarat Al-Haramain website" S15 and the "Other sign languages used for translating Sermon" S13 received significant importance, with mean ratings of 2.74 and 2.62, respectively.

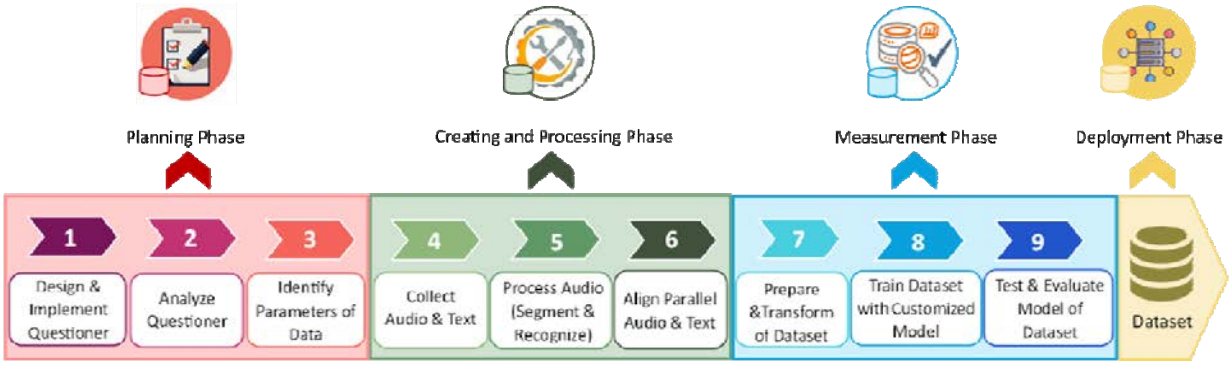


Figure 1 – Phases and stages of Sermon dataset collection

Table 1 – Descriptive statistics for the Preacher and Sermon parameters

	Notation	Parameter Name	Mean	Median	StdDev
Preacher Parameters	P1	Name of Preacher	2.34	3.0	0.772222
	P2	Age of Preacher	1.46	1.0	0.645550
	P3	Original Country of Preacher	1.84	2.0	0.817163
	P4	Academic Qualification of Preacher	2.54	3.0	0.734291
	P5	Years of Experience of Preacher	2.42	3.0	0.784805
Sermon Parameters	S6	Title of Sermon	**2.86**	**3.0**	**0.452205**
	S7	Type of Sermon (topic)	2.52	3.0	0.646498
	S8	Duration of Sermon	2.42	3.0	0.702474
	S9	Date of Sermon	2.48	3.0	0.706818
	S10	Place of Sermon	2.58	3.0	0.609114
	S11	Language of Sermon	2.48	3.0	0.706818
	S12	Other languages of Sermon translated into	2.56	3.0	0.674915
	S13	Other sign languages used for translating Sermon	**2.62**	**3.0**	**0.567486**
	S14	Language complexity of Sermon	2.26	2.0	0.694292
	S15	Reliability of Manarat Al-Haramain Website	**2.74**	**3.0**	**0.486973**
	S16	Reliability of AL-Khutaba Forum Website	2.40	3.0	0.699854

DOI 10.15588/1607-3274-2024-2-10

Also, S15 and S13 the same as “Title of Sermon” S6 received a 3.0 median rating while the standard deviation of both parameters was low, which means there is a strong agreement among experts on its significance.

(2) Correlation Analysis of Parameters (Based on Expert Evaluation): We calculated the Pearson correlation coefficient (*r*) between each of the two parameters as expressed in the equation (1).

$$r = \frac{\sum\left(X - \bar{x}\right)\left(Y - \bar{y}\right)}{\sqrt{\sum\left(X - \bar{x}\right)^2 \sum\left(Y - \bar{y}\right)^2}}, \quad (1)$$

where *X* and *Y* indicate the variables, $\bar{x}$ and, $\bar{y}$ indicate the means of the two variables [23].

We used the heatmap visualization using the Seaborn library in Python. Fig. 3 shows a valuable insight into the relationship between different parameters associated with the Preacher and the Sermon itself. The X-axis presents Sermons’ parameters whereas the Y-axis presents Preachers’ parameters Mainly, the positive correlation between Preacher and Sermon parameters suggests that the experts perceive the increasing importance of Preacher parameters the same as increasing Sermon parameters.
Interestingly, the “original country of Preacher” P3 has a strong positive correlation with the “language of the Sermon” S11, indicating that the language used in the Sermon is highly important and that its importance will increase if “Preacher’s original country” is increasing.

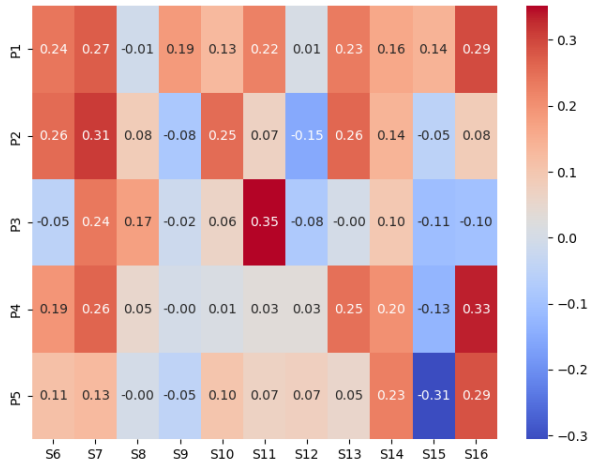


Figure 3 – Heat map correlation between Preacher and Sermon parameters. (P1) Name of Preacher; (P2) Age of Preacher; (P3) Original Country of Preacher; (P4) Academic Qualification of Preacher; (P5) Years of Experience of Preacher; (S6) Title of Sermon; (S7) Type of Sermon(topic); (S8) Duration of Sermon; (S9) Date of Sermon; (S10) Place of Sermon; (S11) Language of Sermon; (S12) Other languages of Sermon translated into; (S13) Other sign languages used for translating Sermon; (S14) Language complexity of Sermon; (S15) Reliability of Manarat Al-Haramain Website; (S16) Reliability of AL-Khutaba Forum Website

Thus, we can find that the language used in the Sermon is highly influenced by the Preacher’s original country. Therefore, these two parameters should be considered in IFS metadata.

Also, we can see that “Preacher’s name” P1, “academic qualifications of Preacher” P4, and “years of experience” P5 show a positive correlation with the “reliability of AL-Khutaba Forum website” S16 a website that provides a source for written texts of Friday sermon for various Sermon places. This suggests that a specific Preacher’s name with higher qualifications and more experience tend to be associated with more reliable content on the AL-Khutaba Forum website.

On the contrary, the “reliability of the Manarat Al-Haramain website” S15 was not affected positively by all Preacher parameters because this website was provided by the government as a source for visual videos of Sermons. Thus, it certainly achieved high important ratings from experts without looking at other parameters’ impact. Overall, this could mean that the expert’s rating sees a connection between these parameters and believes they both contribute to the effectiveness or impact of IFS metadata. However, it’s crucial to note that this is only an indication of how the parameters are related in terms of their perceived importance. It does not necessarily mean that they influence each other in a casual way. For studying the effectiveness and causes, a more in-depth analysis would be necessary with IFS metadata.

(3) Inter-annotator Agreement of Parameters (Based on Expert Evaluation): This study used Kendall’s Coefficient of Concordance (Kendall’s *W*), which is a measurement tool of a non-parametric test for rank correlations and for inter-reliability where its agreement is from 0 (no agreement) to 1 (complete agreement) [24]. The categories degree scale of Kendall’s *W* is illustrated in Table 2.

Table 2 – Categories of Kendall’s W interpretation

W	**Interpretation**
0	No agreement
0.10	Weak agreement
0.30	Moderate agreement
0.60	Strong agreement
1	Perfect agreement

To use Kendall’s *W*, the rate for each item should be rearranged so that it is given by each rater as a rank starting from 1, 2, 3 … etc. If there is more than one item that has the same rate, such as item_1 =2, item_2 =2, each of the two items will have a different rank. Then, the summation of their ranks will be divided by the total number of items that are given the same rate (1 + 2 / 2). Thus, the result of 1.5 will be given to item_1 and item_2 as rank. After that, we calculate the Kendall’s coefficient (*W*) using the following equation:

$$W = \frac{12 \;\; SS_{Total} \;\; Ranks}{k^2\left(n^3 - n\right)}, \quad (2)$$

where *SS* calculated by the formula:

DOI 10.15588/1607-3274-2024-2-10

$$SS = \sum_{i=1} \left(x_i - \bar{x}\right)^2, \quad (3)$$

where x represents the total ranks for each i item that given by raters, $\bar{x}$ is the mean rank of x_i, and n is the number of items or cases being assessed, k represents the number of raters or judges [25].

In our study, we used SPSS to calculate the value of Kendall's $W = 0.210$, which indicates that there is a slight level of agreement among the raters for the ordinal or ranked data. Table 3 represents the mean ranks using Kendall's W, where this rank shows us, which parameters were rated most favorably.

Table 3 – Ranks using Kendall's W test

	Mean Rank
P1	8.15
P2	3.45
P3	5.59
P4	9.38
P5	8.75
S6	11.14
S7	9.15
S8	8.51
S9	8.75
S10	9.38
S11	9.00
S12	9.22
S13	9.63

P1: Name of Preacher, P2: Age of Preacher, P3: Original Country of Preacher, P4: Academic Qualification of Preacher, P5: Years of Experience of Preacher, S6: Title of Sermon, S7: Type of Sermon (topic), S8: Duration of Sermon, S9: Date of Sermon, S10: Place of Sermon, S11: Language of Sermon, S12: Other languages of Sermon translated into, S13: Other sign languages used for translating Sermon, S14: Language complexity of Sermon, S15: Reliability of Manarat Al-Haramain Website, S16: Reliability of AL-Khutaba Forum Website.

As we can see in Table 3, the lower parameter in rank is "Age of AL Preacher" P2 = 3.45, which indicates that it does not have a high chance of being selected as a parameter for the Sermon dataset. At the same time, in the parameter of "Original Country of Preacher" P3 has a rank of 5.59, which is a low rank and does not rate it as the most important parameter. In contrast, "Title of Sermon" S6 and "Reliability of Manarat Al-Haramain Website" S15 stand out with higher rankings of 11.14 and 10.24, respectively. Moreover, S13, P4, S10, S12, S7, and S11 have ranks around 9, which is a high level of agreement between the expert's raters regarding the importance or evaluation of these parameters (rated most favorably).

C) Identify Parameters of Data Collection: the domain of our dataset that will be created is the Islamic Friday Sermons as we can spotlight the importance of the Islamic Friday based on the expert's perspective on the evaluation questionnaire. We illustrated in the previous studies that researchers focused their studies on the collection of either text or speech Sermon data, not both [9, 21]. However, in our study, the aim is to collect a dataset with a size of 100, including 50 audio and 50 corresponding texts, with a total of 50 Islamic Friday sermons from the Grand Mosque (Masjid al-Haram) in Makkah, located in Saudi Arabia. Where experts also highlight the significance of the Grand Mosque. In addition to that, the experts' spotlight on the data type's importance in having Fraidy Sermon text and audio. Through this re-search, we will focus on collecting and creating audio and text. The intention is to utilize this comprehensive dataset in some applications of ML and DL techniques.

In addition to identifying the size and type of dataset, we identify the parameters for both Preacher and Sermon that should be collected in order to create Friday Sermons metadata. We identify the Preacher's parameters and the Sermon's parameters based on the results (Key Findings) of the evaluation questionnaire.

– Key findings in evaluating preacher and sermon parameters:

– In preacher parameters: The experts' evaluation of the importance of Preacher parameters revealed that the "Age of Preacher" has less significance, suggesting it may not contribute significantly to creating our dataset. On the other hand, the following parameters of "Name of Preacher", "Academic Qualification of Preacher", and "Years of Experience of Preacher" had more significant importance, warranting their inclusion in our dataset. Although the "Original Country of Preacher" had less importance, it has a strong correlation with the "language of the Sermon", making it relevant to the "Place of Sermon" and the "Language" used. Based on these strong correlations between the "Original Country of Preacher", "language of the Sermon", and "Place of Sermon" it is reasonable to eliminate the "Original Country of Preacher" which can be inferred from the place and language used for Sermon. Since the place and language used for the Sermon can already provide insights into the cultural and linguistic context, retaining the "Original Country" parameter may not contribute significantly to identifying the more important features for dataset creation. By eliminating this parameter, you can focus on gathering and incorporating the more essential features that have a direct impact on the Sermon.

– In sermon parameters: Several key findings appeared from the evaluation of the im-portance of Sermon parameters. The "Title of Sermon" was observed to hold significant importance, serving as a concise representation of the main theme or topic. Similarly, the "Topic of Sermon" was identified as another crucial parameter, reflecting the subject and content of the Sermon and it figures relevance to the audience's engagement. Also, the "Duration of Sermon" played a significant role in its importance, as we infer from one of the previous studies. It proved that the "Duration of Sermon" influences the audience's attention [20]. Additionally, the "Language of Sermon" was found to be significant, affecting audience accessibility and understanding. In short, both the "Date" and "Place" of the Sermon were deemed significant, as they contributed to the overall impact and resonance of the audience. The parameter of "Other languages Sermon

DOI 10.15588/1607-3274-2024-2-10

is translated into" showed slight importance. However, we believe that this parameter may gain significant importance in future work, especially when researchers who specialize in language translation focus their interest on the different languages into which Sermon is translated. Another parameter that proved to be highly important in our dataset is "Other sign languages used for translating Sermon". Including SL videos and text of Sermon in our data is crucial for making Sermon accessible to the deaf community. Moreover, for blind people, the audio of Sermons is also referred to by some experts as having significant importance to be included in our dataset.

In terms of sourcing Sermon videos and texts, we relied on two websites, Manarat Al-Haramain and AL-Khutaba Forum. Manarat Al-Haramain proved to be a more reliable source as it is backed by the government, whereas AL-Khutaba Forum, though still valuable, was considered less reliable for our dataset. In general, we will consider these two websites as significant sources for obtaining video and text for creating our dataset. On the other hand, we observed that the parameter "Complexity of Sermon" is less important compared to the other parameters. Therefore, we decided to eliminate this parameter from our dataset.

Based on Fig. 2 of the stages of 3.2 Creating and Processing Phase: in this phase, we start to create and process our Sermon dataset, considering the important parameters. Then, processing the audio (segment and analyze) was implemented. Finally, the alignment of the parallel (audio and text) was done in the last step in the creating and processing phase.

In order to go through these three stages (steps) of creating and processing phase, the two approaches (high and low level) were used; see Fig. 4 and Fig. 5.

In the high-level approach, we divide the process of creating our Friday Sermons audio and text dataset (SAT) into multiple modules. The first module is data preprocessing, which is part of the audio and text dataset collection stage. Then, the data seg-mentation and recognition, data annotation, and metadata creation modules were considered as part of the processing data (segment and recognize) stage. While the data verification, and data correction and unification modules are part of aligning parallel audio and text.

The three stages and their relevant modules in the high-level approach for collecting and creating SAT presents in Fig. 4. However, the low-level approach includes the subprocesses (steps) of each of these modules shown in Fig. 5. The deep explanation of each module shows as following:

1) Data preprocessing module: It includes the following subprocess:

– Obtaining the text and video Sermons: we collected videos of Sermons from the website of Manarat Al-Hramain [26], which is released by the Saudi Arabian government, and also had a high level of agreement between the experts' evaluations regarding the importance parameter (rated most favorably) (see Table 2).

From the Manarat Al-Haramain website, we collect around 50 mp4 videos of Fraidy Sermon with a normal resolution of 480p with 30fps that are held in the Holy Mosque of Makkah. On the other hand, we collect 50 texts of Sermons corresponding to each collected Sermon video from the website of the AL-Khutaba Forum, which encompasses Sermons delivered in multiple places in Saudi Arabian mosques, such as Riyad and Jeddah. Also, it has Sermons in different countries, such as al-Aqsa mosque, Egypt mosques, and so on [27]. Based on experts' evaluation regarding the importance parameter (rated most favorably), the AL-Khutaba Forum website recorded a slightly high mean rank (see Table 2).

Therefore, we considered this website a reliable source for gathering relevant text, where each text appeared on the website in docx format.

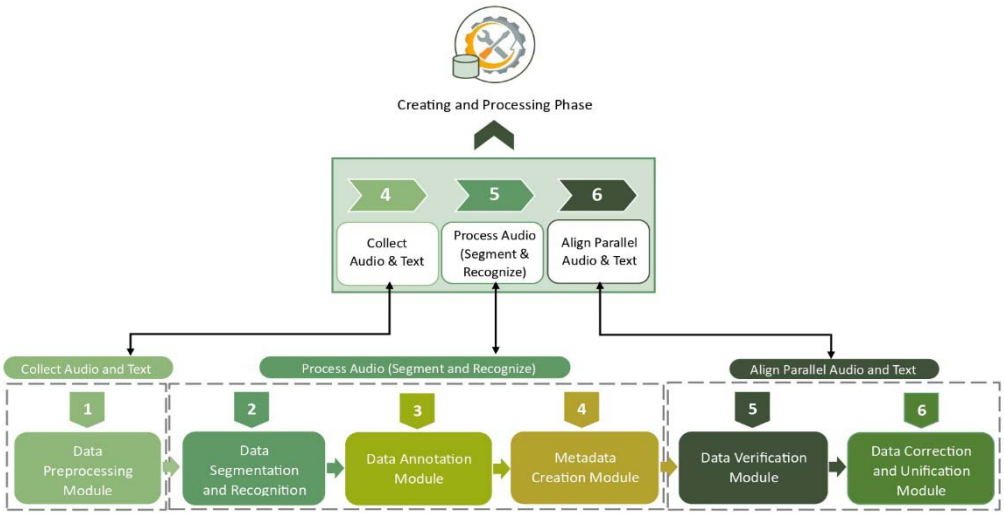


Figure 4 – High-Level approach of collecting and creating SAT

DOI 10.15588/1607-3274-2024-2-10

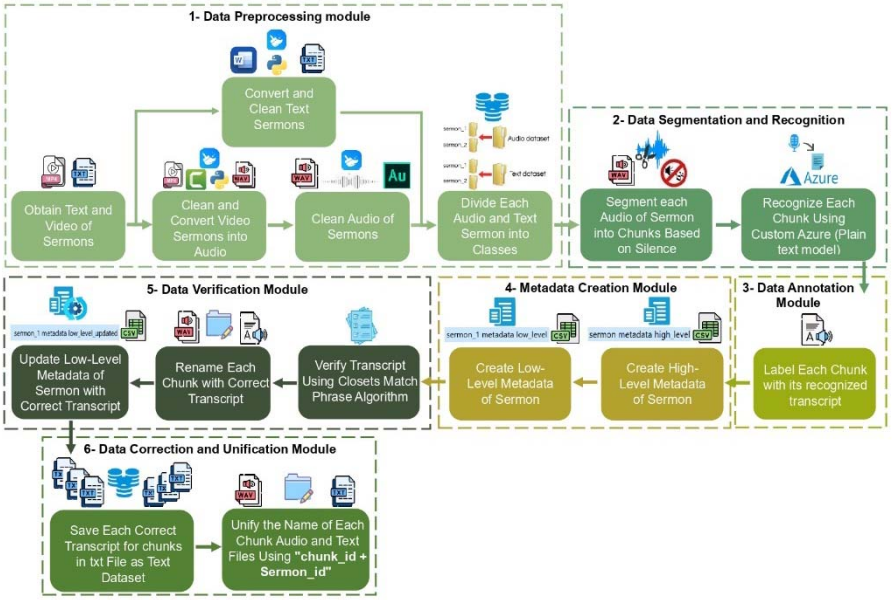


Figure 5 – Low-Level approach of collecting and creating SAT

– Cleaning and converting video into audio: as a next step, we start to clean each video using “Camtasia” (video editing software) by removing the Azan from the beginning and Prayer from the end of the Sermon and just saving the Sermon body (see the block diagram for the structure of Sermon in Fig. 1). Then, we convert the cleaned video into audio (wav format) using the Python Programming Language with the “moviepy” library. After that, we cleaned each audio from noise using “Adobe Audition” (Audio editing software). Also, we removed any stuttering, crying, and coughing…etc., that may be contained in the wav audio. As a result, the cleaned wav audio for each Sermon recorded has an average duration of 17 seconds and 29 minutes at 16 Hz sampling rate, 16-bit PCM (Pulse Code Modulation), and one number of channels.

On the other hand, the collected text Sermon converted from docx into txt format to be suitable for the next stages of annotation and validation. We normalize texts by removing diacritics from each txt file.

– Dividing each audio and text into classes: as a final step in the data preprocessing module, we divide each cleaned wav and txt file of Sermon into two mains separated folders for our dataset, where each folder obtains multiple classes of Sermons, as shown in Fig. 6.

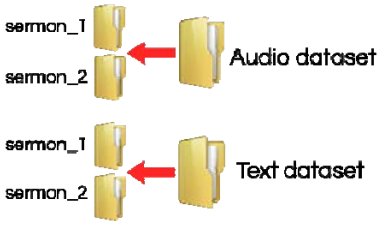


Figure 6 – SAT dataset folders

As represented in Table 4 and 5, each wav file of Fraidy Sermon was named as follows: Sermon id, Sermon date using Islamic calendar, and Preacher’s name where the Preacher’s name includes first, middle, and last name. For example: Sermon_23 (23-12-1443) Sheikh Usama Khayyat.

The txt file of Sermon was named as follows: Sermon title + _without_diacritics. For example: “Winning the bliss comes by following the straight path_without_diacritics”.

Table 4 – An example of audio file name for one Sermon

Audio Dataset	
Folder Name	**File Name**
Sermon_23	Sermon_23 (23-12-1443) الشيخ أسامه خياط
	Sermon_23 (23-12-1443) Sheikh Usama Khayyat

Table 5 – An example of text file name for one Sermon

Text Dataset	
Folder Name	**File Name**
Sermon_23	الفوز بالنعيم باتباع الصراط المستقيم_without_diacritics
	Winning the bliss comes by following the straight path_without_diacritics”

2) Data segmentation and recognition module**:** this module includes two subprocesses as following:

– Segmentation based on silence: we utilized the “split_on_silence” function from Python in order to segment the audio of each Sermon into smaller chunks based on silence. By adjusting some parameters, like (silence threshold) from 40 to 50 and (minimum silence length) at least 200ms, we ensured that segments were properly identified and the segmentation was done well. In addition to that, (keep silence) 200ms of silence were added at the beginning and end of each segment to maintain the completeness of each chunk. The number of chunks for each audio Sermon is varying depending on the speech patterns and silence duration of each Preacher.

DOI 10.15588/1607-3274-2024-2-10

The generated chunks from one Sermon were named sequentially as chunk0, chunk1, chunk2, and so on.

– Recognition using Microsoft Azure: we utilized Microsoft Azure Customization in order to recognize each audio chunk of the Sermon. The plain text model, specifically the Speech Studio Custom Model, was employed because, in our case, the Sermon speech contains difficult words and is considered a special domain. Also, we imported the Speech SDK package from Azure Cognitive Services in Python to configure and build the custom model using our Azure portal's subscription key and endpoint. Training and customizing the model for Sermons' domain were used in the Speech Studio. This enabled us to obtain speech recognition for each chunk, which will be utilized in the next annotation module [28].

3) Data annotation module: the recognized speech in each audio chunk will be used for labeling each chunk with its relevant content that is recognized by Microsoft Speech Azure. All of these rec-ognized chunks of one Sermon are saved in one folder named "Sermon_id_recognizedChunks", where the Sermon_id could be Sermon_1, Sermon_2, and so on.

4) Metadata creation module: this module plays a crucial role in analyzing and organizing the Sermon dataset by creating two (CSV) files for both high-level and low-level metadata. This metadata provides descriptive information about Preacher and Sermon for better searchability, organization, classification, contextual understanding, and analysis of the dataset. In general, this module contains two subprocesses: creating high-level metadata and creating low-level metadata. This created Sermon metadata facilitates efficient management and utilization of our Sermon dataset.

The high-level metadata obtains general information about each Sermon and the person who introduced it (Preacher). Where this information based on 12 features as illustrates in following (Table 6).

In the low-level metadata of SAT, we present deeper information about each chunk's audio files of each Sermon where it captures more detailed information. This can include technical details, such as the following: (Sermon ID, Chunk ID, duration of each chunk (start and end time for each chunk), transcript of each chunk, total number of chunks, silence threshold, minimum silence length, and keep silence).

5) Data verification module**:** this module consists of three subprocess which are the following:

– Verifying transcript using Closets Matching Phrase algorithm: we create a similarity matching algorithm for finding the Closest Matching Phrase (CMP) between (Tran-script) and the (full text). Where the transcript is generated using Azure speech and saved on low-level metadata (CSV file). The way that the CPM algorithm is used to verify transcripts from any speech recognition engine is shown in Fig. 7.

Table 6 – SAT dataset's of high-level metadata features description

Features	**Variables Description**
Sermon ID	Each sermon has a unique ID, for example, sermon_1, sermon_2, sermon_3 ….
Name	Name of preacher who deliver the sermon
Academic Qualification	Educational background of preacher, such as أصول الفقه "The Principles of Jurisprudence" and so on
Experience	Number of years that being a preacher in introducing Fraidy Sermon
Title	Title of Sermon
Domain	Topic or theme that Sermon belongs
Date	Date of the Sermon that was introduced in (Arabic calendar -Hijri)
Language	Language of Sermon, it could be Arabic, English, and other language
Duration	Length of each Sermon starts from the beginning preacher's speech until the end, formatted as (hour: minutes: second)
Location	Place where the Sermon is delivered, for example, Holly Mosque in Mecca or any other mosque or religious center
Other language	Whether the Sermon is translated into other languages, formatted as (No = 0, Yes =1)
Arabic Sign Language	Whether the Sermon is interpreted or translated into Arabic Sign Language, formatted as (No = 0, Yes =1)

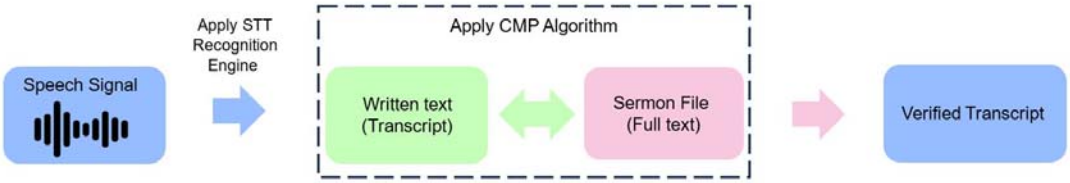


Figure 7 – Block diagram of transcript verification using CMP Algorithm

DOI 10.15588/1607-3274-2024-2-10

The algorithm allows customization through the optional parameters (window size) and (min ratio). Where window size determines the size of the sliding window used for comparison, and min ratio sets the minimum required matching ratio for a phrase to be considered a close match. After running the algorithm, the Closest Matching Phrases with their maximum ratio will be printed. This printed ratio helps us to identify phrases with low ratios that require correction.

Algorithm 1: Find Closest Matching Phrase (CMP) in Text

Require: T (string): The transcript to match
FT (string): The full text to search within
WIND (integer, optional, default=5): Number of words around the transcript to consider during matching
minR (float, optional, default=0.70): Minimum similarity ratio to consider a match
Ensure: CMP (string): Closest matching phrase within the full text
maxR (float): Similarity ratio of the closest matching phrase

```
Procedure Find_CMP (T, FT, W=5, minR=0.70)
    TW ⟵ split T into words
    FTW ⟵ split FT into words
    Initialize tracking variables
    maxR ⟵ 0
    CMP ⟵ NULL
     Iterate through words in the FT
    for I = 0 to (length of FTW – length of TW + 1) do
      Expand the window around the current position
      for j = –WIND to WIND do
        Extract words from the current window
        end_index ⟵ i + length of TW + j – 1
        current_phrase_words ⟵ subarray of FTW from index i to end_index
        current_phrase ⟵ join current_phrase_words into a string
        Calculate similarity ratio
        R ⟵ calculate similarity between T and current_phrase using SequenceMatcher (ratio)
          Update tracking variables if R is greater than maxR
          if R > maxR then
            maxR ⟵ R
            if maxR >= minR then
              CMP ⟵ current_phrase
          end if
        end if
    end for
    end for
  return the CMP and its similarity R
  return CMP, maxR
end procedure
```

– Renaming each chunk with the correct transcript: based on using the developed algorithm, the updated transcript will be automatically based on the calculated similarity between T (transcript) and current_phrase using SequenceMatcher (ratio).

If the updated max ratio is more than the minimum ratio = 0.70, a similar transcript from the full-text file will be printed as a verified transcript otherwise the current transcript will be printed where it needs to be checked manually.

If the updated max ratio is less than the minimum ratio = 0.70, we take a look at the audio chunk associated with each low-ratio phrase listened to, and a comparison is made with the transcript in the CSV file and the full text in the text file. Once the correct transcript is determined, we start to replace the transcript that needs to be updated manually based on manual records for a transcript that has a low ratio.

In some cases of the lower ratio, if the recognized transcript cannot be found in the full-text file, it means that the Preacher used a new phrase that has not been written in the full text.

– Updating low-level metadata with correct transcript: we save the result of the transcript correction based on using similarity matching in the updated CSV file for low-level metadata.

6) Data correction and unification module: in this final module, we process the following:

– Saving the correct transcript in a txt file: from the corrected CSV file that contains the transcript of each chunk we save these transcripts as separate files (txt format) for each chunk.

– Unifying the files' names in two datasets (text and audio): in order to unify the file names in the two dataset types (text and audio) to easily follow the audio file with a corresponding text file, we named each generated text file as following structure (chunk_id + Sermon_id.txt). On the other hand, we named each audio file as (as fol-

DOI 10.15588/1607-3274-2024-2-10

lowing structure (chunk_id + Sermon_id.wav). After that, each text file will be saved in the text dataset while each audio file will be saved in the audio dataset, as shown in the organized files of the SAT dataset in Fig. 8.

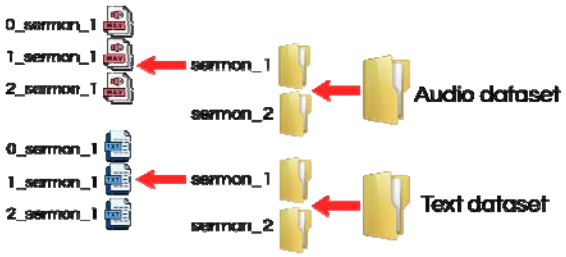


Figure 8 – SAT dataset with equivalent files

4 EXPERIMENTS

In this section, the outcomes of the Measurement and Deployment phases will be presented for the SAT dataset. The objectives of these last two phases were to evaluate the accuracy of our dataset, followed by deploying it strategically in order to ensure its accessibility and applicability in real-world scenarios.

3.3 Measurement Phase: to measure the accuracy of the created dataset SAT we implement three main stages, A) Prepare and transform the dataset B) Train with a customized model. C) Test and evaluate a model of the dataset.

In the preparing and transforming, in order to apply the model, we prepared SAT by adding all txt files (transcripts) in one txt file where each line contains one single audio information (audio file name with extension (.wav) followed by space then transcript). More importantly, the text should be normalized (does not include punctuations or diacritics). Also, the text encoding must be UTF-8 BOM (txt format) While the audio should be in WAV format with a sample rate of 8,000 or 16,000 Hz. The maximum length for each audio file is 2 hours for testing and 60 seconds for training. The audio files and the transcript file should be grouped in a zip file with a maximum size of 2 GB.

In the training, we used a custom model speech recognition Speech to Text (STT) provided by Microsoft Azure for training in order to improve recognition accuracy. In our case, we used 10136 audio files with a sample rate of 16,000 Hz and around 14 hours for 50 sermons.

We trained using a custom speech recognition model with 30 percent of the weight and during training; the labeled text was normalized to increase the readability.

In the testing and evaluation, we tested 8526 audio files within around 5 hours and a half. Overall, after applying our new algorithm CMP to enhance speech recognition to achieve high confidence in equivalent speech-to-text. Our customized model (fine-tuned with the SAT dataset) achieved a 5.13% Word Error Rate (WER), which indicates that our created dataset SAT with speech recognition model performed better than the base model. Our custom model performance was compared to the base model performance, the result is presented in Table 7.

Table 7 – Comparison of WER scores for Azure Custom Model (fine-tuned with our SAT dataset) and Microsoft Azure Speech Model

Model	Dev	Test	WER	Insertion	Substitution	Deletion
Customized Speech (Our SAT dataset)	**54.29**	**45.71**	**5.13**	**0.39**	**3.49**	**1.24**
Microsoft Azure Speech	54.29	45.71	15.61	2.20	11.80	1.60

3.4 Deployment Phase: This phase is considered the last phase in the dataset collection and creation. It starts by publishing our SAT through one of the popular platforms, such as GitHub, Mendeley Data …etc, which helps other researchers to refine and deploy any other method and model depending on the needs. We will publish the documentation about SAT metadata to support users and engage them in future research for more under-standing of this dataset. As a result of this engagement, the collected and created SAT will be maintained and updated.

5 RESULTS

By following the four phases (Fig. 2) with its stages to collect and create our dataset SAT, we successfully created a dataset that included 21,253 WAV audio files and corresponding 21,253 TXT transcript files of 50 Friday sermons.

This SAT dataset can be used for the exploration and analysis of Sermon content, delivery, and various linguistic aspects by specialists and other researchers. Table 8 summarizes the information about SAT.

Table 8 – Summarizing of SAT dataset Information

	Audio	Text
Number of Sermons	50	50
Number of Files	21253	21253
Format	WAV	TXT
Total Size	1.56 GB	1.24 MB
Total Duration of All Sermons	14h 34m 31s	
Average Duration for Each Sermon	17m 29s	
Total Words	83141	
Distinct Words	25226	
Total Number of Preachers	9	

For more details about the various durations for all50 Friday sermons of Masjid al-Haram illustrates in Fig. 9.

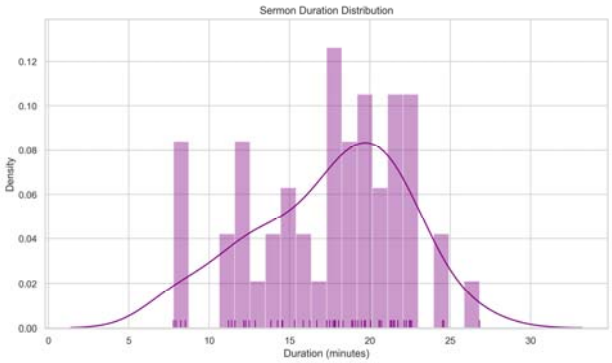


Figure 9 – Duration distribution of 50 Friday Sermons

DOI 10.15588/1607-3274-2024-2-10

Moreover, Fig.10 presents the duration of Sermon by domain where we have 4 main domains as the previous researchers specify [14]. The Invitation to Islam or Its Defending has a high duration of around 27 minutes whereas the other 3 domains of Religious Education for the Public, Proving Faith in the Souls, and Correction of Faults and Prohibition of Evils obtain around 18 minutes.

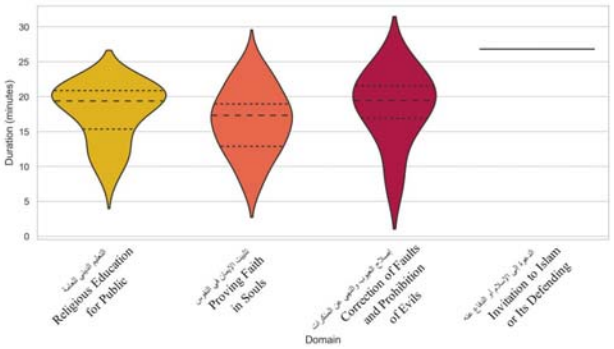


Figure 10 – Duration distribution of 50 Friday Sermons by domain

In terms of Automatic Speech Recognition (ASR), our SAT dataset can be utilized for speech recognition systems because of the potential variation in terminology, speaking styles, and SAT dataset) will be suitable for this task.

6 DISCUSSION

The successful creation of 21,253 WAV audio files and the equivalent number of TXT (transcripts) for the 50 Friday Sermons dataset marks a significant advancement in the religious discourse domain. This SAT dataset not only facilitates a comprehensive exploration of sermon content but also can be used as a valuable resource. Moreover, the diversity of our dataset and its metadata availability has the potential to support nuanced research on language use in religious settings. It can be used in discourse analysis and MT, for instance, from Arabic speech into another language, or into other Sign Languages. Despite its strengths, the dataset's scope which is limited to sermons from the Grand Mosque (Masjid al-Haram) in Makkah, suggests a direction-expanding dataset in the future direction using our methods to include a broader of sources, which seeks to be AI-driven in religious contexts.

CONCLUSIONS

In this research, we have completed four main phases, including planning, creating and processing, measuring, and deploying phases. We have curated and prepared a part of the Sermon audio and text dataset (SAT), providing valuable resources for future research and implementation in various sectors. Moreover, our created dataset SAT achieved less WER in fine-tuned using the Azure custom model compared with the Azur baseline model by 10.48 %. **Moreover, this research developed a CMP algorithm** for enhancing the customized Azure speech recognition to verify our SAT by correcting phrases that have a lower ratio, which leads to reducing the WER.

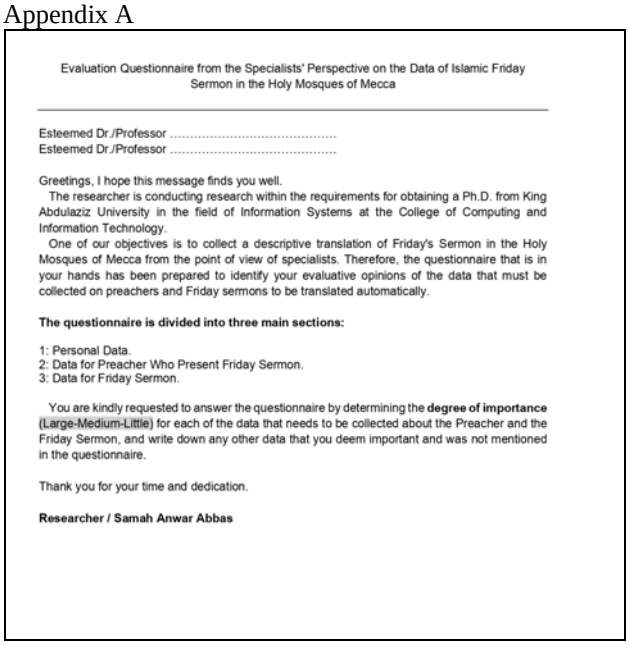
In future work, we will expand our SAT with ArSL to achieve a variety of datasets. Also, we are going to explore advanced alignment techniques and algorithms to improve the accuracy and efficiency of the ArSL video of the Sermon to support advanced machine translation techniques.

ACKNOWLEDGEMENTS

The authors extend gratitude to the Presidency of Religious Affairs at the Grand Mosque and the Prophet's Mosque for facilitating access to essential data sources, which significantly enriched this study.

APPENDICES

Appendix A


Evaluation Questionnaire from the Specialists' Perspective on the Data of Islamic Friday Sermon in the Holy Mosques of Mecca

Esteemed Dr./Professor .....
Esteemed Dr./Professor .....

Greetings, I hope this message finds you well.
The researcher is conducting research within the requirements for obtaining a Ph.D. from King Abdulaziz University in the field of Information Systems at the College of Computing and Information Technology.
One of our objectives is to collect a descriptive translation of Friday's Sermon in the Holy Mosques of Mecca from the point of view of specialists. Therefore, the questionnaire that is in your hands has been prepared to identify your evaluative opinions of the data that must be collected on preachers and Friday sermons to be translated automatically.

The questionnaire is divided into three main sections:

1: Personal Data.
2: Data for Preacher Who Present Friday Sermon.
3: Data for Friday Sermon.

You are kindly requested to answer the questionnaire by determining the **degree of importance** (Large-Medium-Little) for each of the data that needs to be collected about the Preacher and the Friday Sermon, and write down any other data that you deem important and was not mentioned in the questionnaire.

Thank you for your time and dedication.

Researcher / Samah Anwar Abbas

Figure A1: Evaluation Questionnaire from the Specialists' Perspective on the Data of Islamic Friday Sermon in the Holy Mosques of Mecca: (the purpose of investigation)

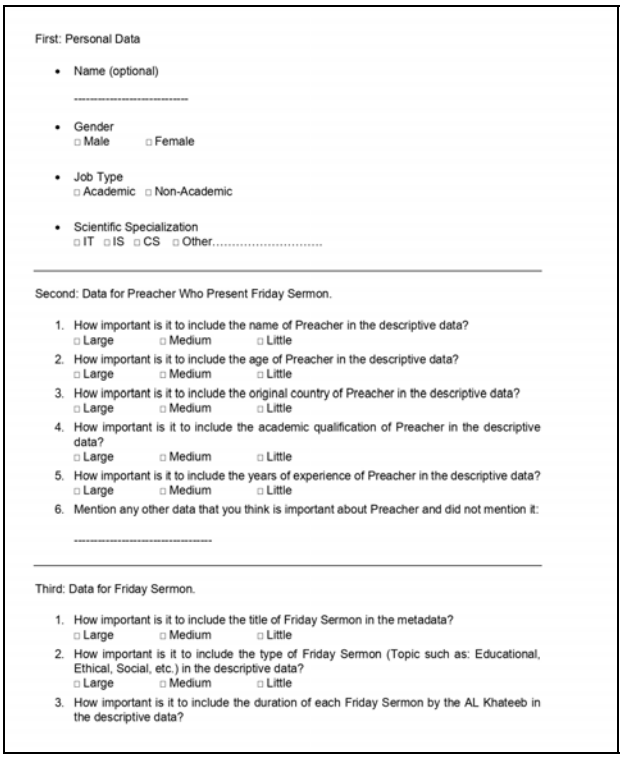

Consent for Participation in this Research:

I, the undersigned participant of this study, have been made aware of the research's objectives, as well as its potential advantages and risks. I comprehend that my involvement in this research is voluntary and I have the right to withdraw at any moment without having to justify my decision. I hereby agree to partake in this study.

□ I agree to participate in this study.
□ I do not agree to participate in this study.

Figure A2: Evaluation Questionnaire from the Specialists' Perspective on the Data of Islamic Friday Sermon in the Holy Mosques of Mecca: (the consent for participation)

DOI 10.15588/1607-3274-2024-2-10



First: Personal Data

- Name (optional)

- Gender
  □ Male □ Female
- Job Type
  □ Academic □ Non-Academic
- Scientific Specialization
  □ IT □ IS □ CS □ Other.....

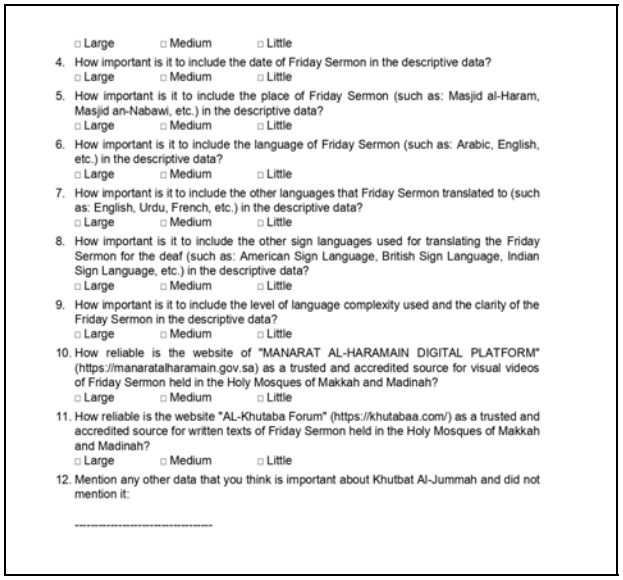
Second: Data for Preacher Who Present Friday Sermon.

1. How important is it to include the name of Preacher in the descriptive data?
   □ Large □ Medium □ Little
2. How important is it to include the age of Preacher in the descriptive data?
   □ Large □ Medium □ Little
3. How important is it to include the original country of Preacher in the descriptive data?
   □ Large □ Medium □ Little
4. How important is it to include the academic qualification of Preacher in the descriptive data?
   □ Large □ Medium □ Little
5. How important is it to include the years of experience of Preacher in the descriptive data?
   □ Large □ Medium □ Little
6. Mention any other data that you think is important about Preacher and did not mention it:

Third: Data for Friday Sermon.

1. How important is it to include the title of Friday Sermon in the metadata?
   □ Large □ Medium □ Little
2. How important is it to include the type of Friday Sermon (Topic such as: Educational, Ethical, Social, etc.) in the descriptive data?
   □ Large □ Medium □ Little
3. How important is it to include the duration of each Friday Sermon by the AL Khateeb in the descriptive data?

Figure A3: Evaluation Questionnaire from the Specialists' Perspective on the Data of Islamic Friday Sermon in the Holy Mosques of Mecca: (presenting the Three axes including questions about the level of importance of Preacher and Friday Sermon parameters)



□ Large □ Medium □ Little

4. How important is it to include the date of Friday Sermon in the descriptive data?
   □ Large □ Medium □ Little
5. How important is it to include the place of Friday Sermon (such as: Masjid al-Haram, Masjid an-Nabawi, etc.) in the descriptive data?
   □ Large □ Medium □ Little
6. How important is it to include the language of Friday Sermon (such as: Arabic, English, etc.) in the descriptive data?
   □ Large □ Medium □ Little
7. How important is it to include the other languages that Friday Sermon translated to (such as: English, Urdu, French, etc.) in the descriptive data?
   □ Large □ Medium □ Little
8. How important is it to include the other sign languages used for translating the Friday Sermon for the deaf (such as: American Sign Language, British Sign Language, Indian Sign Language, etc.) in the descriptive data?
   □ Large □ Medium □ Little
9. How important is it to include the level of language complexity used and the clarity of the Friday Sermon in the descriptive data?
   □ Large □ Medium □ Little
10. How reliable is the website of "MANARAT AL-HARAMAIN DIGITAL PLATFORM" (https://manaratalharamain.gov.sa) as a trusted and accredited source for visual videos of Friday Sermon held in the Holy Mosques of Makkah and Madinah?
    □ Large □ Medium □ Little
11. How reliable is the website "AL-Khutaba Forum" (https://khutabaa.com/) as a trusted and accredited source for written texts of Friday Sermon held in the Holy Mosques of Makkah and Madinah?
    □ Large □ Medium □ Little
12. Mention any other data that you think is important about Khutbat Al-Jummah and did not mention it:

Figure A4: Evaluation Questionnaire from the Specialists' Perspective on the Data of Islamic Friday Sermon in the Holy Mosques of Mecca: (presenting the Three axes including questions about the level of importance of Preacher and Friday Sermon parameters) cont

REFERENCES

1. Chen H., Xie W., Vedaldi A., Zisserman A. VGGSound: A Large-Scale Audio-Visual Dataset, 2020.
2. Cohn N., Cardoso B., Klomberg B., Hacımusaoğlu I. The Visual Language Research Corpus (VLRC), *An Annotated Corpus of Comics from Asia, Europe, and the United States.* Lang Resources & Evaluation, 2023, DOI:10.1007/s10579-023-09673-0.
3. Mounsef J., Hasib M., Raza A. Building an Arabic Dialectal Diagnostic Dataset for Healthcare, *IJACSA*, 2022, No.13, DOI:10.14569/IJACSA.2022.01307100.
4. Alfraidi T., Abdeen M. A. R., Yatimi A., Alluhaibi R., Al-Thubaity A. The Saudi Novel Corpus, *Design and Compilation. Applied Sciences*, 2022, No. 12, P. 6648, DOI:10.3390/app12136648.
5. Abdelhay M., Mohammed A., Hefny H.A. Deep Learning for Arabic Healthcare, *MedicalBot. Soc. Netw. Anal. Min.* 2023, No. 13, P. 71. DOI:10.1007/s13278-023-01077-w.
6. Abbas S., Al-Barhamtoshy H., Alotaibi F. Towards an Arabic Sign Language (ArSL) Corpus for Deaf Drivers. *PeerJ Comput. Sci.*, 2021, No. 7, e741, DOI:10.7717/peerj-cs.741.
7. Asyafie M. A., Harun M., Shapiai M. I., Khalid P. I. Identification of Phoneme and Its Distribution of Malay Language Derived from Friday Sermon Transcripts. In Proceedings of the 2014 IEEE Student Conference on Research and Development, 2014, December, pp. 1–6.
8. Saddhono K., Rakhmawati A. Sociolinguistic Studies of Friday Sermon Using Javanese as an Effort to Preserves Indigenous Language in Java Island, *In Proceedings of the 2nd International Conference on Sociology Education*, SCITEPRESS – Science and Technology Publications. Bandung, Indonesia, 2017, pp. 829–833.
9. Alkhawaldeh A. A. Deixis in English Islamic Friday Sermons, *A Pragma-Discourse Analysis. Studies in English Language and Education*, 2022, No. 9, pp. 418–437, DOI:10.24815/siele.v9i1.21415.
10. Aksoy O. Preaching to Social Media: Turkey's Friday Khutbas and Their Effects on Twitter, SocArXiv, 2021, May, No. 12.
11. Usman A. H., Iskandar A. Analysis of Friday Sermon Duration, *Intellectual Reflection of Classical and Contemporary Islamic Scholars. Journal of Religious & Theological Information*, 2022, No. 21, pp. 68–81, doi:10.1080/10477845.2021.1928349.
12. Gürlesin Ö. F. Understanding the Political and Religious Implications of Turkish Civil Religion in The Netherlands: A Critical Discourse Analysis of ISN Friday Sermons. *Religions*, 2023, No. 14, P. 990, DOI:10.3390/rel14080990.
13. Nor M.R.M. Multicultural Discourse from the Minbar: A Study on Khutbah Texts Prepared by Jakim Malaysia. In; Fukami N., Sato S., Eds.; *JSPS Asia and Africa Science Platform Program.* Organization of Islamic Area Studies, Waseda University. Tokyo, Japan, 2012, pp. 55–62 ISBN 978-4-904039-52-6.
14. Ismail Ali Mohammed Art of Oratory and Skills of Orator. Researches in the preparation of preacher preacher. Fifth edition, Dar Alkalema, Cairo-Egypt, 2016.
15. Mahmood I., Kasim Z. Metadiscourse Resources across Themes of Islamic Friday Sermon, 2021, No. 21, pp. 45–61, DOI: 10.17576/gema-2021-2101-01-03.
16. Sukarno S., Salikin H. The The Generic Structure Potential of Friday Sermons in Jember, *International Journal of Linguistics and Translation Studies.* Indonesia, 2022, No. 3, pp. 56–73. DOI:10.36892/ijlts.v3i1.207.
17. Mohammed Saleh Al-Hamzi A., Sumarlam, Santosa R., Jamal M. A Pragmatic and Discourse Study of Common Deixis Used by Yemeni-Arab Preachers in Friday Islamic Sermons at Yemeni Mosques, *Cogent Arts & Humanities*

DOI 10.15588/1607-3274-2024-2-10

2023, No. 10, P. 2177241, doi:10.1080/23311983.2023.2177241.
18. Mahmood I., Kasim Z. Interpersonal Metadiscursive Features in Contemporary Islamic Friday Sermon, *3L: Language, Linguistics, Literature*, 2019, No. 25, pp. 85–99. DOI:10.17576/3L-2019-2501-06.
19. Wardoyo C. Directive Speech Acts Performed in Khutbah (Islamic Friday Sermon), 2017.
20. Fahruroji F., Rakhmat M., Shodiq M. The Understanding of Friday Prayer Attendees (Mustamik) Towards Friday Sermon Discourse, 2017, P. 779.
21. Carol S., Hofheinz L. A Content Analysis of the Friday Sermons of the Turkish-Islamic Union for Religious Affairs in Germany (DİTİB), *Politics and Religion*, 2022, DOI:10.1017/S1755048321000353.
22. Jafilus M., Asha'ari M. F., Rasit R. Thematic Analysis of the Content of the Friday Sermon in Negeri Sembilan, IJARBSS, 2021, No. 11, pp. 84–98, DOI:10.6007/IJARBSS/v11-i6/10087.
23. Numeracy, Maths and Statistics – Academic Skills Kit Available online: https://www.ncl.ac.uk/webtemplate/ask-assets/external/maths-resources/statistics/regression-and-correlation/strength-of-correlation.html (accessed on 19 February 2024).
24. Moslem S., Ghorbanzadeh O., Blaschke T., Duleba S. Analysing Stakeholder Consensus for a Sustainable Transport Development Decision by the Fuzzy AHP and Interval AHP, *Sustainability*, 2019, No. 11, P. 3271, DOI:10.3390/su11123271.
25. Encyclopedia of Statistics in Behavioral Science; Everitt B., Howell D. C., Eds. John Wiley & Sons. Hoboken, N. J, 2005, ISBN 978-0-470-86080-9.
26. MNARAT AL-HARAMAIN Available online: https://manaratalharamain.gov.sa/home (accessed on 25 June 2023).
27. Khutaba Forum Available online: https://khutabaa.com/en (accessed on 25 June 2023).
28. Beatman A. Improve Speech-to-Text Accuracy with Azure Custom Speech | Azure Blog | Microsoft Azure Available online: https://azure.microsoft.com/en-us/blog/improve-speechtotext-accuracy-with-azure-custom-speech/ (accessed on 23 September 2023).

Received 04.03.2024.
Accepted 26.04.2024.

УДК 004.942(045)

СТВОРЕННЯ МАСШТАБОВАНОГО НАБОРУ ДАНИХ ДЛЯ П'ЯТНИЧНИХ ПРОПОВІДЕЙ З АУДІО ТА ТЕКСТУ (ПАТ)

Самах А. А. – д-р філософії кафедри інформаційних систем факультету обчислювальної техніки та інформаційних технологій та викладач кафедри інформаційних систем управління факультету економіки та адміністрації Університету короля Абдула Азіза, Джидда, Мекка, Саудівська Аравія.

Дімах Х. А. – д-р філософії, доцент, доцент кафедри інформаційних систем, факультет обчислювальної техніки та інформаційних технологій, Університет короля Абдула Азіза, Джидда, Мекка, Саудівська Аравія.

Хасанін М. А. – д-р техн. наук, професор, професор кафедри інформаційних технологій, факультет обчислювальної техніки та інформаційних технологій, Університет короля Абдула Азіза, Джидда, Мекка, Саудівська Аравія.

АНОТАЦІЯ

Актуальність. Сьогодні збір і створення наборів даних у різних секторах стає все більш поширеним. Незважаючи на таке поширене створення даних, досі існує прогалина в спеціалізованих областях, зокрема в області Ісламських п'ятничних проповідей. Вона багата на теологічні, культурні та лінгвістичні дослідження, які стосуються арабських і мусульманських країн, а не лише релігійних дискурсів.

Мета. Мета цього дослідження полягає в тому, щоб усунути цю нестачу, створивши повний набір даних аудіо та тексту проповідей із його метаданими. Це спрямоване надати великий ресурс для вивчення релігії, лінгвістики та соціології. Крім того, це дозволить підтримати досягнення у сфері штучного інтелекту, таких як технології обробки природної мови та розпізнавання мовлення.

Метод. Розробка набору даних проходила у чотири окремі етапи: планування, створення та обробка, вимірювання та розгортання. Набір даних містить колекцію з 21 253 аудіо та відповідних файлів розшифровки, які були успішно створені. Удосконалені методи обробки звуку були використані для покращення розпізнавання мовлення та надання набору даних, який підходить для широкого використання.

Результати. Тонко налаштований набір даних досяг 5,13% частоти помилок у словах (Word Error Rate – WER), що вказує на значне покращення точності, порівняно з базовою моделлю Microsoft Azure Speech. Це досягнення вказує на якість набору даних і ефективність використовуваних методів обробки. У світлі цього було розроблено новий алгоритм фрази з найбільшою відповідністю, щоб підвищити високу надійність еквівалентного мовлення до тексту шляхом коригування фраз із меншим співвідношенням.

Висновки. Це дослідження створює ресурс для поєднання різних досліджень, таких як релігієзнавство, лінгвістика та соціологія. Крім того, воно демонструє потенціал у сфері штучного інтелекту і підтримує його програми. У майбутніх дослідженнях ми зосередимося на збагаченні цього розширення набору даних шляхом додавання відеокорпусу мовою жестів, використовуючи вдосконалені методи вирівнювання. Він підтримуватиме поточні розробки машинного перекладу для ширшого розуміння ісламських п'ятничних проповідей у різних мовах і культурах.

КЛЮЧОВІ СЛОВА: п'ятничні проповіді, хутба, розпізнавання арабської мови, набір звукових і текстових даних, машинний переклад.

DOI 10.15588/1607-3274-2024-2-10

ЛІТЕРАТУРА

1. VGGSound / [H. Chen, W. Xie, A. Vedaldi, A. Zisserman]. –A Large-Scale Audio-Visual Dataset, 2020.
2. The Visual Language Research Corpus (VLRC) / [N. Cohn, B. Cardoso, B. Klomberg, I. Hacımusaoğlu] // An Annotated Corpus of Comics from Asia, Europe, and the United States. – Lang Resources & Evaluation. – 2023, DOI: 10.1007/s10579-023-09673-0.
3. Mounsef J. Building an Arabic Dialectal Diagnostic Dataset for Healthcare / J. Mounsef, M. Hasib, A. Raza // IJACSA. – 2022. – No. 13. DOI:10.14569/IJACSA.2022.01307100.
4. The Saudi Novel Corpus: Design and Compilation / [T. Alfraidi, M.A.R.Abdeen, A. Yatimi et al.] // Applied Sciences. – 2022. – No. 12. – P. 6648. DOI: 10.3390/app12136648.
5. Abdelhay M. Deep Learning for Arabic Healthcare / M. Abdelhay, A. Mohammed, H. A. Hefny // MedicalBot. Soc. Netw. Anal. Min. – 2023. – No. 13. – P. 71. DOI: 10.1007/s13278-023-01077-w.
6. Abbas S. Towards an Arabic Sign Language (ArSL) Corpus for Deaf Drivers / S. Abbas, H. Al-Barhamtoshy, F. Alotaibi // PeerJ Comput. Sci. – 2021. – No. 7. – e741, DOI:10.7717/peerj-cs.741.
7. Identification of Phoneme and Its Distribution of Malay Language Derived from Friday Sermon Transcripts / [M. A. Asyafie, M. Harun, M. I. Shapiai, P. I. Khalid] // In Proceedings of the 2014 IEEE Student Conference on Research and Development. –2014. – December. – P. 1–6.
8. Saddhono K. Sociolinguistic Studies of Friday Sermon Using Javanese as an Effort to Preserves Indigenous Language in Java Island / K. Saddhono, A. Rakhmawati // In Proceedings of the 2nd International Conference on Sociology Education. – SCITEPRESS – Science and Technology Publications : Bandung, Indonesia, 2017. – P. 829–833.
9. Alkhawaldeh, A.A. Deixis in English Islamic Friday Sermons: A Pragma-Discourse Analysis / A. A. Alkhawaldeh // Studies in English Language and Education. – 2022. – No. 9. – P. 418–437. DOI:10.24815/siele.v9i1.21415.
10. Aksoy O. Preaching to Social Media: Turkey's Friday Khutbas and Their Effects on Twitter / O. Aksoy // SocArXiv. – 2021. – May. – No. 12.
11. Usman, A.H.; Iskandar, A. Analysis of Friday Sermon Duration: Intellectual Reflection of Classical and Contemporary Islamic Scholars / A. H. Usman, A. Iskandar // Journal of Religious & Theological Information. – 2022. – No. 21. – P. 68–81. DOI:10.1080/10477845.2021.1928349.
12. Gürlesin Ö. F. Understanding the Political and Religious Implications of Turkish Civil Religion in The Netherlands: A Critical Discourse Analysis of ISN Friday Sermons / Ö. F. Gürlesin // Religions. – 2023. – No. 14. – P. 990, DOI:10.3390/rel14080990.
13. Nor, M.R.M. Multicultural Discourse from the Minbar: A Study on Khutbah Texts Prepared by Jakim Malaysia / M.R.M. Nor, In N. Fukami, S. Sato, Eds. // JSPS Asia and Africa Science Platform Program. – Organization of Islamic Area Studies, Waseda University : Tokyo. – Japan, 2012. – P. 55–62. ISBN 978-4-904039-52-6.
14. Ismail Ali Mohammed Art of Oratory and Skills of Orator: Researches in the preparation of preacher preacher / Ismail Ali Mohammed. – Fifth edition, Dar Alkalema : Cairo-Egypt, 2016.
15. Mahmood I. Metadiscourse Resources across Themes of Islamic Friday Sermon / I. Mahmood, Z. Kasim. – 2021. – No. 21. – P. 45–61. DOI:10.17576/gema-2021-2101-01-03.
16. Sukarno S. The The Generic Structure Potential of Friday Sermons in Jember / S. Sukarno, H. Salikin // International Journal of Linguistics and Translation Studies. – Indonesia. – 2022. – No. 3. – P. 56–73. DOI:10.36892/ijlts.v3i1.207.
17. A Pragmatic and Discourse Study of Common Deixis Used by Yemeni-Arab Preachers in Friday Islamic Sermons at Yemeni Mosques / [A. Mohammed Saleh Al-Hamzi, Sumarlam, R. Santosa, M. Jamal] // Cogent Arts & Humanities. – 2023. – No. 10. – P. 2177241, DOI:10.1080/23311983.2023.2177241.
18. Mahmood I. Interpersonal Metadiscursive Features in Contemporary Islamic Friday Sermon / I. Mahmood, Z. Kasim // 3L: Language, Linguistics, Literature. – 2019. – No. 25. – P. 85–99. DOI:10.17576/3L-2019-2501-06.
19. Wardoyo C. Directive Speech Acts Performed in Khutbah (Islamic Friday Sermon) / C. Wardoyo. – 2017.
20. Fahruroji F. The Understanding of Friday Prayer Attendees (Mustamik) Towards Friday Sermon Discourse / F. Fahruroji, M. Rakhmat, M. Shodiq, 2017. – P. 779.
21. Carol S. A Content Analysis of the Friday Sermons of the Turkish-Islamic Union for Religious Affairs in Germany (DİTİB) / S. Carol, L. Hofheinz // Politics and Religion. – 2022. DOI:10.1017/S1755048321000353.
22. Jafilus M. Thematic Analysis of the Content of the Friday Sermon in Negeri Sembilan / M. Jafilus, M. F. Asha'ari, R. Rasit // IJARBSS. – 2021. – No. 11. – P. 84–98. DOI:10.6007/IJARBSS/v11-i6/10087.
23. Numeracy, Maths and Statistics – Academic Skills Kit Available online: https://www.ncl.ac.uk/webtemplate/ask-assets/external/maths-resources/statistics/regression-and-correlation/strength-of-correlation.html (accessed on 19 February 2024).
24. Stakeholder Consensus for a Sustainable Transport Development Decision by the Fuzzy AHP and Interval AHP / [S. Moslem, O. Ghorbanzadeh, T. Blaschke, S. Duleba] // Sustainability. – 2019. – No. 11. – P. 3271. DOI:10.3390/su11123271.
25. Encyclopedia of Statistics in Behavioral Science / B. Everitt, D. C. Howell, Eds. – John Wiley & Sons : Hoboken, N.J, 2005. ISBN 978-0-470-86080-9.
26. MNARAT AL-HARAMAIN Available online: https://manaratalharamain.gov.sa/home (accessed on 25 June 2023).
27. Khutaba Forum Available online: https://khutabaa.com/en (accessed on 25 June 2023).
28. BeatmanA. Improve Speech-to-Text Accuracy with Azure Custom Speech | Azure Blog | Microsoft Azure Available online: https://azure.microsoft.com/en-us/blog/improve-speechtotext-accuracy-with-azure-custom-speech/ (accessed on 23 September 2023).

DOI 10.15588/1607-3274-2024-2-10

УДК 004.827:519.816

РОЗРОБКА МЕТОДИКИ ОЦІНЮВАННЯ ЗНАЧЕНЬ ФУНКЦІЇ НАЛЕЖНОСТІ НА ОСНОВІ ГРУПОВОЇ ЕКСПЕРТИЗИ У МЕТОДІ НЕЧІТКОГО ДЕРЕВА РІШЕНЬ

Швед А. В. – д-р техн. наук, професор, професор кафедри інженерії програмного забезпечення Чорноморського національного університету імені Петра Могили, Миколаїв, Україна.

АНОТАЦІЯ

Актуальність. Останнім часом нечіткі дерева рішень набули широкого застосування при вирішенні задач класифікації та регресії. У випадку відсутності об'єктивної інформації для побудови функції належності елементів вузлам дерева, єдиним шляхом отримання інформації є залучення експертів. У випадку залучення групи фахівців виникає задача агрегування думок експертів з метою синтезу групового рішення. Об'єктом дослідження є групові експертні оцінки ступеню належності елемента заданому класу, атрибуту, що потребують структуризації та агрегування при побудові та аналізі нечіткого дерева рішень.

Метою роботи є розробка методики визначення значень функції належності елемента заданому класу (атрибуту) за результатами опитування групи експертів при побудові та аналізі нечітких дерев рішень.

Метод. Методика дослідження ґрунтується на комплексному застосуванні математичного апарату теорії правдоподібних та парадоксальних міркувань та методів нечіткої логіки для вирішення задачі агрегування нечітких експертних оцінок значень атрибутів (ознак) класифікації при побудові та аналізі нечіткого дерева рішень. Запропонований підхід використовує механізм комбінування експертних свідоцтв, сформованих в рамках гібридної моделі Дезера-Смарандаке, на основі правила перерозподілу конфліктів PCR5 для побудови групового рішення.

Результати. Розглянуті питання структуризації нечіткої експертної інформації та запропонована методика синтезу групової експертної оцінки відносно значень атрибутів (ознак) класифікації при побудові та аналізі нечіткого дерева рішень.

Висновки. Дістали подальшого розвитку моделі та методи структуризації та синтезу групових рішень в умовах нечіткої експертної інформації. На відміну від існуючих експертних методів визначення значень функції належності в умовах групового вибору, запропонований підхід дозволяє синтезувати групове рішення з різним значенням конфліктної маси при комбінуванні вихідних експертних свідоцтв. Такий підхід дозволяє коректно агрегувати як узгоджені, так і суперечливі (конфліктні) експертні свідоцтва.

КЛЮЧОВІ СЛОВА: теорія правдоподібних та парадоксальних міркувань, нечіткі дерева рішень, правило перерозподілу конфліктів.

АБРЕВІАТУРИ

ДР – дерева рішень;

ЕП – експертні переваги;

ЕС – експертні свідоцтва;

ТДС – теорія Дезера-Смарандаке, теорія правдоподібних та парадоксальних міркувань;

ТДШ – теорія Демпстера-Шейфера, теорія свідоцтв;

ШІ – штучний інтелект;

НОМЕНКЛАТУРА

A – множина атрибутів (ознак) класифікації, $a_l \in A$;

B_i – компонента профілю ЕП, кожен елемент якої відображає вибір експерта E_i відносно значень термів лінгвістичної змінної β для заданої сукупності об'єктів O;

P^{gr} – груповий профіль ЕП;

D^{Ω} – множина всіх можливих підмножин, які можуть бути сформовані на множині Ω, на основі операцій об'єднання та перетину, включаючи порожню множину;

E – група експертів;

Gr – синтаксичне правило, що породжує нові терми;

H_i – компонента профілю ЕП, кожен елемент якої відображає оцінку упевненості експерта E_i в тому, що для заданої сукупності об'єктів O лінгвістична змінна β набуває визначені в B_i значення;

M_i – компонента профілю ЕП, кожен елемент якої містить числові значення основної маси ймовірності (функції належності) для визначених в B_i термів лінгвістичної змінної β;

Mp – множина належностей;

Mr – семантичне правило генерації нечітких множин для кожного терму лінгвістичної змінної;

O – множина аналізованих об'єктів класифікації;

P – множина індивідуальних профілів ЕП;

$T(\beta)$ – терм-множина лінгвістичної змінної β;

V_{a_l} – множина значень атрибуту a_l;

U_k – область визначення змінної k (універсум);

$\tilde{X}$ – нечітка множина;

α – найменування нечіткої змінної;

β – найменування лінгвістичної змінної;

$\mu_{\tilde{X}}(u)$ – характеристична функція належності елемента u нечіткій множині $\tilde{X}$;

Ω – основа задачі;

d_c – атрибут-рішення;

d_{metric} – метрика відстані між групами ЕС;

$m_k(Y_k)$ – основна маса ймовірності, призначена множині Y_k;

DOI 10.15588/1607-3274-2024-2-11

$m_{d+1}(\Omega)$ – основна маса ймовірності, призначена множині Ω;

w_i – коефіцієнт компетентності експерта E_i;

$\oplus$ – правило комбінування ЕС.

ВСТУП

Сучасні тенденції розвитку інформаційних технологій сприяють активній інтеграції методів ШІ у всі сфери людської діяльності. Використання технологій ШІ дозволяє швидше та ефективніше створювати більш точні моделі процесів, що протікають у складних соціальних, економічних, технічних, організаційних та інших системах під впливом сукупності різного роду факторів зовнішнього та внутрішнього оточення. Ключовою особливістю технологій ШІ є здатність навчатися, накопичувати знання та застосовувати їх. Саме машинне навчання, як один із ключових напрямів розвитку ШІ, дозволяє створювати на основі отриманих знань, досвіду та нових даних моделі здатні до навчання та адаптації для вирішення поставленої задачі. Однією з основних задач машинного навчання є задача класифікації, суть якої полягає у формуванні розбиття вихідної сукупності об'єктів на визначені класи відповідно до заданого набору ознак. Даний тип завдань у машинному навчанні відноситься до розділу контрольованого машинного навчання (навчання з учителем), і передбачає наявність навчальної вибірки для виявлення закономірностей у вихідному наборі даних та побудови вирішальних правил класифікації на їх основі.

Серед методів класифікації досить широкого поширення набув метод ДР [1–3]. Дерева рішень є графічним методом побудови класифікаційної моделі, що легко інтерпретується та дозволяє обробляти як категоріальні, так і інтервальні дані. Даний метод не вимагає спеціальної підготовки вихідних даних і дозволяє оперувати великими обсягами даних (*Big data*), їх можна ефективно застосовувати до даних з пропущеними значеннями атрибутів. У той же час дерева чутливі до зміни в навчальній вибірці, що може призвести до коригування побудованої моделі (класифікаційних правил). У процесі побудови та навчання дерева необхідно дотримуватись балансу між точністю та складністю одержуваної ієрархічної структури (можуть утворюватись занадто складні конструкції, які недостатньо повно відображають наявні дані, внаслідок чого може виникати проблема перенавчання дерева) [4].

Метод ДР набув широкого застосування в галузі машинного навчання, прогнозного моделювання та прийняття рішень у різних галузях: медичні дослідження [5–6], фінансовий сектор, маркетинг [7], діагностика несправностей у технічних системах [8], та ін.

Останнім часом активного поширення набув метод побудови ДР на основі нечіткого підходу [9–10].

Об'єктом дослідження є групові експертні оцінки ступеню належності елемента заданому класу, атрибуту, що потребують структуризації та агрегування при побудові та аналізі нечіткого ДР.

Предметом дослідження є моделі та методи структуризації групових експертних оцінок на основі математичного апарату теорії правдоподібних та парадоксальних міркувань та нечіткої логіки.

Метою роботи є розробка методики визначення значень функції належності елемента заданому класу (атрибуту) за результатами опитування групи експертів при побудові та аналізі нечітких ДР.

1 ПОСТАНОВКА ЗАДАЧІ

Для побудови ДР вихідна сукупність даних може бути сформована на основі різних джерел, і являти собою статистичну, аналітичну, експериментальну / емпіричну інформацію, що отримана на основі методів спостереження (реєстрації, моніторингу), вимірювань (експериментів, тестів), тощо. У випадку повної відсутності об'єктивної вихідної інформації єдиним шляхом отримання даних є залучення сторонніх осіб (інтерв'ю, опитування, фокус-групи, методи експертних оцінок, тощо). Експертні знання можуть бути отримані у формі індивідуальних та групових експертних оцінок. Зазвичай для аналізу проблемної ситуації залучають групу експертів (фахівців певної предметної області), оскільки групова експертиза дає можливість отримати більш об'єктивну оцінку на основі аналізу певної сукупності індивідуальних думок експертів.

При аналізі та побудові ДР (при вирішенні задач класифікації та регресії) для графічного подання вихідної сукупності даних, відображення їх семантики використовується реляційна система – таблиця, рядки якої відповідають аналізованим об'єктам, а стовпчики – ознакам (критеріям, атрибутам) цих елементів. В комірці на перетині j-го рядка та l-го стовпчика відображається значення l-ї ознаки для j-го елементу, таким чином кожен рядок таблиці відображає один об'єкт (приклад) і відповідні йому значення атрибутів. Останній рядок зазвичай являє собою атрибут-рішення.

Таким чином, сукупність вихідних даних, яку необхідно класифікувати на основі визначеної множини критеріїв (атрибутів), можна подати у формі $\mathrm{DT} = (O, A, V, d_c, C, f)$, де $O = \{o_j \mid j = \overline{1,z}\}$ – не порожня скінчена множина аналізованих об'єктів; $A = \{a_l \mid l = \overline{1,m}\}$ – не порожня скінчена множина примітивних атрибутів (ознак); $V = \bigcup_{a_l \in A} V_{a_l}$, V_{a_l} – множина значень атрибуту a_l (область атрибуту a_l); d_c ($|d_c| = 1$) являє собою атрибут-рішення, який характеризує можливі класи, до яких може бути віднесено об'єкт $o_j \in O$; C – множина значень атрибуту d_c; f – інформаційна функція, така, що $\forall a_l \in A$, $o \in O$, $f(o, a_l) \in V_{a_l}$.

DOI 10.15588/1607-3274-2024-2-11

Припустимо, що $\exists a_t \in A$, для яких значення множини V_{a_t} формується на основі групової експертної оцінки. Задача полягає в агрегування відповідних значень релевантних атрибутів $a_t^i(o_j)$, що формуються на основі оцінок експертів E_i, $i=\overline{1,n}$, і синтезу групової оцінки: $\forall t=\overline{1,k}: \underset{i}{agr}(a_t^i(o_j)) \to a_t^{gr}(o_j)$, $o_j \in O$, $1 \le k \le m$. Агрегування групових експертних оцінок здійснюється для кожного атрибуту окремо.

2 ОГЛЯД ЛІТЕРАТУРИ

В даний час запропоновано значну кількість алгоритмів побудови дерев рішень [1–3, 11–14], серед яких найбільшого поширення набули методи ID3 [11, 12], C4.5 [13] та CART [14]. Більшість відомих алгоритмів, при пошуку оптимальної структури, використовують стратегію жадібного пошуку. У [8] для синтезу дерева запропоновано використовувати стохастичні, евристичні методи багатовимірної безградієнтної оптимізації, що дозволяє отримувати структуру дерева з кращими апроксимаційними властивостями.

Класичний метод побудови ДР передбачає, що кожен аналізований об'єкт належить конкретному заздалегідь визначеному класу (вузлу). Однак у реальній практиці можуть виникати ситуації, при яких не завжди вдається досягти однозначної належності аналізованого об'єкта до певної категорії (класу). У цьому випадку виникає невизначеність при спробі віднести об'єкт до одного із заданих класів, або встановити чітке та однозначне значення класифікаційного атрибуту (ознаки). Математичний апарат нечіткої логіки дозволяє ефективно моделювати ситуації, за яких об'єкт може належати як одному, так і декільком класам одночасно, але з різним ступенем, який можна охарактеризувати значенням функції належності об'єкту до заданого класу [15–16]. Зазвичай нечіткість проявляється у ситуаціях, коли на основі якісних оцінок оцінюються значення кількісних параметрів досліджуваного об'єкту, процесу або явища.

Застосування нечітких ДР (*fuzzy decision trees*) дозволяє для кожного класифікаційного атрибуту (ознаки) виділити декілька лінгвістичних значень і встановити відповідні ступені належності об'єктів (прикладів) до них [9–10]. Таким чином, кожний об'єкт може мати властивості більше одного лінгвістичного значення, що характеризує певний заданий атрибут.

3 МАТЕРІАЛИ І МЕТОДИ

Дерева рішень складаються із двох основних компонент: процедури побудови символічного дерева та процедури виведення для прийняття рішення. У загальному випадку процес побудови ДР складається з наступних послідовних етапів: вибір атрибуту розбиття; вибір критерію зупинення навчання; вибір методу відсікання гілок; оцінка точності побудованої моделі.

У нечіткому ДР кожний атрибут розглядається як лінгвістична змінна, яка може приймати деяке заздалегідь зумовлене вербальне значення, одне чи декілька. Кожному значенню лінгвістичної змінної відповідає певна нечітка множина зі своєю функцією належності.

Введемо деякі позначення. Припустимо задана деяка універсальна множина U, тоді нечітка множина $\tilde{X} \subseteq U$ є множиною всіх впорядкованих пар виду [15–16]:

$$\tilde{X} = \{(U, \mu_{\tilde{X}}(u))\}, \quad u \in U, \ \mu_{\tilde{X}}(u) \in Mp. \qquad (1)$$

Нечіткі та лінгвістичні змінні використовуються для опису нечітких множин на основі слів та / або словосполучень природньої людської мови, що є більш природнім для людини, ніж оперувати кількісними значеннями.

Нечітка змінна задається трійкою [15–16]:

$$<\alpha, U_\alpha, \tilde{X}>. \qquad (2)$$

В якості значень лінгвістичної змінної використовуються словесний (вербальний) опис різних об'єктів, процесів та явищ.

В теорії нечітких множин лінгвістична змінна визначається наступним кортежем [15–16]:

$$<\beta, T(\beta), U_\beta, Gr, Mr>. \qquad (3)$$

Функція належності будується, зазвичай, на основі статистичних даних (частотний метод), або на основі оцінок експертів (одного чи групи). У разі відсутності достатньої кількості статистичної інформації єдиним можливим способом побудови функції належності є проведення експертного опитування.

Умовно методи побудови функцій належності можна поділити на дві групи: прямі та непрямі [17–18]. У прямих методах ступінь належності задається безпосередньо експертами (метод відносних частот, та ін.). У непрямих методах значення функції належності визначаються виходячи із заздалегідь сформульованих умов відповідно до заданого алгоритму (метод попарних порівнянь, та ін.).

Ступінь суб'єктивності одержаних ЕП можна зменшити в умовах групової експертизи. У цьому випадку виникають дві основні задачі: оцінка узгодженості ЕП і побудова узагальненої оцінки.

Розглянемо основні положення підходу, при якому значення лінгвістичної змінної, що ставиться у відповідність до деякого атрибуту ДР, визначається на основі групової експертизи.

Припустимо, група експертів $E = \{E_i \mid i = \overline{1,n}\}$ оцінюючи значення лінгвістичної змінної β для заданої сукупності об'єктів $O = \{o_j \mid j = \overline{1,z}\}$ сформувала множину профілів ЕП виду P = <B, L>. За умови, що

DOI 10.15588/1607-3274-2024-2-11

сукупність значень лінгвістичної змінної β утворює терм-множину $T(\beta)=\{t_l \mid l=\overline{1,k}\}$, перша компонента кортежу являє собою сукупність $\mathrm{B}=\{B_i \mid i=\overline{1,n}\}$, кожен елемент якої $B_i=\{b_j^i \mid j=\overline{1,z}\}$ може приймати значення відповідного терму або декількох термів, що набуває лінгвістична змінна β для об'єкту o_j.

Друга компонента L містить оцінки експертів, на основі яких можуть бути обраховані числові значення функцій належності для визначених в b_j^i термів лінгвістичної змінної β відносно об'єкта o_j, що встановлені експертом E_i.

Задача полягає у синтезі групового профілю $\mathrm{P}^{gr}=<\mathrm{B}^{gr},\mathrm{L}^{gr}>$. Кожен елемент $b_j^{gr}\in \mathrm{B}^{gr}$, $j=\overline{1,z}$ відображає групове рішення відносно значення що набуває лінгвістична змінна β для об'єкту o_j, і формується на основі агрегування оцінок $B_i=\{b_j^i \mid j=\overline{1,z}\}$, $\forall i=\overline{1,n}$. Елементи $l_j^{gr}\in \mathrm{L}^{gr}$, $j=\overline{1,z}$ містять значення функції належності визначених в b_j^{gr} термів лінгвістичної змінної β, отримані за результатами агрегування індивідуальних ЕП. Якщо задано декілька лінгвістичних змінних $\{\beta_1,...,\beta_q\}$, значення яких встановлюються шляхом експертного опитування, то буде сформовано відповідна кількість групових профілів $\left\{\mathrm{P}_1^{gr},...,\mathrm{P}_q^{gr}\right\}$.

Для синтезу групової оцінки в роботі використано математичний апарат ТДС в рамках якого множина $T(\beta)$ розглядається як основа задачі Ω [19]. На відміну від моделі Шейфера (ТДШ), основа задачі Ω розглядається виключно як множина вичерпних елементів. Такі елементи можуть формально описувати неточні, нечіткі (розмиті) поняття та знання про оточуючий світ, наприклад, відтінки кольору, градації віку, температури і т.п. Внаслідок чого деякі елементи можуть перекривати один одного, і відповідно неможливо досягти їх взаємної виключності.

В рамках нотації ТДС на множині вихідних даних $\Omega = T(\beta)$ може бути сформовано $|D^{\Omega}|$ підмножин на основі операцій ∪ та ∩, включаючи порожню множину ∅ [19].

Таким чином експертом можуть бути виділені підмножини $X_i\subseteq D^{\Omega}$, $i=\overline{1,|D^{\Omega}|}$, що задовольняють умовам [19]:

$$
\begin{aligned}
&1.\ X_i=\{\varnothing\};\\
&2.\ X_i=\{t_l\}\ -\text{експертом обраний (оцінений)}\\
&\text{один елемент } t_l\in\Omega.\\
&3.\ X_i=\{t_l \mid l=\overline{1,p}\},\ p<k\ -\text{експертом виділено } p \text{ елементів } t_l\in\Omega.\\
&4.\ X_i=\Omega=\{t_l \mid l=\overline{1,k}\};\\
&5.\ \text{якщо } X_i, X_j\subset D^{\Omega}, \text{ тоді } X_i\cap X_j\in D^{\Omega} \text{ та}\\
&X_i\cup X_j\in D^{\Omega}.
\end{aligned}
\qquad (4)
$$

ТДС оперує двома видами моделей: вільна та гібридна модель [19]. Вільна модель містить всі можливі підмножини основи задачі. Гібридна модель визначається із вільної моделі шляхом введення обмежень на деякі підмножини елементів X_i із множини D^{Ω}, за умови, що $X_i\neq\varnothing$. Це пояснюється тим, що в реальних задачах немає необхідності визначати основні маси ймовірності всім можливим підмножинам D^{Ω}, тому, що завжди можливо існування елементів, які є взаємовиключними.

Будемо вважати, що в процесі експертного опитування на основі множини $T(\beta)$ експертом E_i можуть бути сформовані підмножини, що відображають його судження, або одноелементні, $|b_j^i|=1$, або використовуючи операцію ∪. Таким чином кількість можливих виділених підмножин становить 2^{Ω}, враховуючи порожній вибір ($b_j^i=\{\varnothing\}$).

Обмеження, що накладаються, і умови проведення процедури експертного опитування можуть привести до наступних ситуацій.

Випадок 1. Експерт може встановити відповідність об'єкта до декількох термів лінгвістичної змінної, але з різним ступенем належності (безпосередньо задати значення функцій належності).

За результатами експертного опитування формуються профілі ЕП виду P = <B, M>. Перша компонента кортежу являє собою сукупність $\mathrm{B}=\{B_i \mid i=\overline{1,n}\}$, кожен елемент якої $B_i=\{b_j^i \mid j=\overline{1,z}\}$ відображає вибір експерта E_i відносно значень, що може набувати лінгвістична змінна β для об'єкту o_j. При цьому $b_j^i=\{Y_k \mid k=\overline{1,d}\}$, $|Y_k|=1$, $d<2^{|\Omega|}$ являє собою більш ніж одне значення (декілька термів лінгвістичної змінної). Друга компонента кортежу являє собою сукупність $\mathrm{M}=\{M_i \mid i=\overline{1,n}\}$, кожен елемент якої $M_i=\{\mathrm{m}_j^i \mid j=\overline{1,z}\}$ містить основну масу ймовірності m_j^i призначену для b_j^i, що відповідає числовим значення функцій належності для визначених в b_j^i термів лінгвістичної змінної β відносно об'єкта o_j, які встановлено експертом E_i. При цьому $\mathrm{m}_j^i=\{m_k \mid k=\overline{1,d}\}$, $d<2^{|\Omega|}$, $\forall i,j:|\mathrm{m}_j^i|=|b_j^i|$, $i=\overline{1,n}$, $j=\overline{1,z}$. Таким чином, елемент $m_k=\mathrm{m}_j^i$ містить значення функції належності терму $Y_k\subseteq b_j^i$ лінгвістичної змінної β для об'єкту o_j задане експертом E_i.

DOI 10.15588/1607-3274-2024-2-11

Виходячи з нотації ТДС, кожен елемент $Y_k \subseteq b_j^i$ повинен підпорядковуватися системі правил (4); значення, що містить $m_k = \mathrm{m}_j^i$ відповідає системі умов:

$$0 \le m_k(Y_k) \le 1,\ \forall (Y_k \in b_j^i),\ m_k(\varnothing) = 0,\quad \sum_{Y_k \in b_j^i} m_k(Y_k) = 1. \tag{5}$$

Агрегування суджень експертів виконується у відповідності до запропонованої процедури:

1.1 Структуризація задачі. Для кожного $o_j \in O$ на основі сформованого набору профілів ЕП Р формується кортеж $< B_j^*, M_j^*, T_j^* >$, де $B_j^* = \{b_j^i\}$, $i = \overline{1,n}$ – сукупність тверджень групи експертів; $M_j^* = \{\mathrm{m}_j^i\}$, $i = \overline{1,n}$ – сукупність відповідних значень функції належності, сформованих на основі суджень групи експертів; множину $T_j^* = \{t_f^{*j} \mid f = \overline{1,v}\}$, $v \le |T(\beta)|$, утворюють елементи множини $T(\beta)$ на основі яких утворені елементи $b_j^i \in B_j^*$, $i = \overline{1,n}$.

1.2 Визначення порядку агрегування (комбінування) ЕС. Для комбінування обирається пара ЕС $b_j^i, b_j^h \in B_j^*$, таких, що при $i \ne h$ $\min d_{metric}(\mathrm{m}_j^i, \mathrm{m}_j^h) \in [0;1]$, де d_{metric} – деяка метрика відстані між групами ЕС [20–23].

1.3 Агрегування ЕС здійснюється шляхом комбінування отриманих основних мас ймовірності $M_j^* = \{\mathrm{m}_j^i \mid i = \overline{1,n}\}$ та $B_j^* = \{b_j^i\}$, за всіма експертами E_i, $i = \overline{1,n}$, для кожного $o_j \in O$ окремо: $\mathrm{M}_j^{comb} = \mathrm{m}_j^1 \oplus \mathrm{m}_j^2 \oplus ... \oplus \mathrm{m}_j^n$, $B_j^{comb} = b_j^1 \oplus b_j^2 \oplus ... \oplus b_j^n$, де $\mathrm{m}_j^i \in M_j^*$, $b_j^i \in B_j^*$. В якості правила комбінування запропоновано використовувати правило PCR5 [19].

Комбінована маса ймовірності $m_{PCR5}(C)$ згідно з правилом перерозподілу конфліктів PCR5 ($\forall C \subset D^\Omega \setminus \{\varnothing\}$) визначається відповідно до

$$m_{PCR5}(C) = m_{12}(C) + \sum_{\substack{Y \in D^\Omega \setminus \{X\} \\ Y \cap X = \varnothing}} \left[\frac{m_1(X)^2 \cdot m_2(Y)}{m_1(X) + m_2(Y)} + \frac{m_2(X)^2 \cdot m_1(Y)}{m_2(X) + m_1(Y)} \right]. \tag{6}$$

Правило комбінування PCR5 перерозподіляє основну масу ймовірності, віднесену до порожньої множини, на підмножини, залучені до локальних конфліктів, пропорційно основним масам ймовірності цих підмножин. Ця властивість дозволяє коректно поводитися з узагальненою масою ймовірності, що віднесена до порожніх перетинів. Правило комбінування PCR5 дозволяє обробляти ЕП у ситуації, коли конфліктна маса впевненості досягає максимально можливого значення (набуває 1), при цьому будуть розраховані комбіновані основні маси ймовірності для всіх виділених експертами підмножин, включаючи одноелементні.

Випадок 2. Експерт може встановити відповідність об'єкта до декількох термів лінгвістичної змінної із різним ступенем упевненості у своєму виборі.

За результатами експертного опитування формуються профілі ЕП виду P = <B, H>.

Перша компонента кортежу являє собою сукупність $\mathrm{B} = \{B_i \mid i = \overline{1,n}\}$, кожен елемент якої $B_i = \{b_j^i \mid j = \overline{1,z}\}$ відображає вибір експерта E_i відносно значень, що може набувати лінгвістична змінна β для об'єкту o_j. При цьому $b_j^i = \{Y_k \mid k = \overline{1,d}\}$, $|Y_k| > 1$, $d < 2^{|\Omega|}$ ($|Y_k| > 1$, то всі елемент в групі Y_k є рівнозначними, може бути заданий тільки один терм із групи, наприклад, якщо $Y_k = \{t_1, t_2\}$, то мається на увазі, що β набуде тільки одне із можливих значень, або t_1,або t_2).

Друга компонента кортежу являє собою сукупність $\mathrm{H} = \{H_i \mid i = \overline{1,n}\}$, $H_i = \{h_j^i \mid j = \overline{1,z}\}$. Кожен елемент $X_k \in h_j^i$, $k = \overline{1,d}$ відображає оцінку упевненості експерта E_i в тому, що для об'єкта o_j лінгвістична змінна β набуває значення терму $Y_k \in b_j^i$. Виходячи з нотації ТДС, кожен елемент $Y_k \subseteq b_j^i$ повинен підпорядковуватися системі правил (4); кожен елемент $X_k \in h_j^i$ може бути виражений в рамках певної заданої шкали, використовуючи діапазон чисел від 0 до деякого заданого N ($N > 0$).

Агрегування суджень експертів виконується у відповідності до запропонованої процедури:

2.1 Структуризація задачі. Для кожного $o_j \in O$ на основі сформованого набору профілів ЕП Р формується кортеж $< B_j^*, H_j^*, T_j^* >$, де $B_j^* = \{b_j^i\}$, $i = \overline{1,n}$ – сукупність тверджень групи експертів; $H_j^* = \{h_j^i\}$, $i = \overline{1,n}$ – сукупність відповідних оцінок впевненості експертів в своєму виборі; множину $T_j^* = \{t_f^{*j} \mid f = \overline{1,v}\}$, $v \le |T(\beta)|$, утворюють елементи множини $T(\beta)$ на основі яких сформовані $b_j^i \in B_j^*$, $i = \overline{1,n}$.

2.2 Визначення основних мас ймовірності, що відповідають виділеним підмножинам $Y_k \subseteq b_j^i$, $\forall b_j^i \in B_j^*$.

DOI 10.15588/1607-3274-2024-2-11

Для кожної сформованої системи підмножин $b_j^i = \{Y_k \mid k = \overline{1,d}\}$ буде отримано вектор $\mathrm{m}_j^i = \{m_k \mid k = \overline{1,d+1}\}$, елементи якого відповідають умові (3) і визначаються відповідно до виразу [24]:

$$m_k(Y_k) = \frac{R_1}{R_2 + \sqrt{d}}, \quad m_{d+1}(\Omega) = \frac{\sqrt{d}}{R_2 + \sqrt{d}}, \qquad (7)$$

де R_1 відповідає значенню $X_k \in h_j^i$; R_2 відповідає значенню $\sum_{k=1}^{d} X_k$; d – загальна кількість сформованих E_i підмножин $Y_k \in b_j^i$.

Значення $m_{d+1}(\Omega)$ є мірою невизначеності вибору експерта E_i відносно значень, що може набувати лінгвістична змінна β для об'єкту o_j..

Якщо задано вектор коефіцієнтів компетентності експертів $W = \{w_i \mid i = \overline{1,n}\}$, то

$$R_1 = X_k \cdot w_i; \quad R_2 = \sum_{k=1}^{d} X_k \cdot w_i. \qquad (8)$$

На основі отриманих оцінок m_j^i для кожного аналізованого $o_j \in O$ формується вектор $M_j^* = \{\mathrm{m}_j^i \mid i = \overline{1,n}\}$.

2.3 Визначення порядку агрегування (комбінування) ЕП: обирається така пара $b_j^i, b_j^h \in B_j^*$, що при $i \neq h$ $\min d_{metric}(\mathrm{m}_j^i, \mathrm{m}_j^h) \in [0;1]$.

2.4 Агрегування ЕП здійснюється шляхом комбінування отриманих основних мас ймовірності $M_j^* = \{\mathrm{m}_j^i \mid i = \overline{1,n}\}$ та $B_j^* = \{b_j^i\}$, за всіма експертами E_i, $i = \overline{1,n}$, для кожного $o_j \in O$ окремо. В якості правила комбінування запропоновано використовувати правило PCR5.

Результатом комбінування є вектор $B_j^{comb} = \{Y_k^{comb} \mid k = \overline{1,v}\}$, $v \le 2^{|\Omega|}$ і вектор $\mathrm{M}_j^{comb} = \{m(Y_k^{comb}) \mid k = \overline{1,v}\}$, відповідно.

2.5 Розрахунок верхньої і нижньої межі ймовірності для кожного $t_f^{*j} \in T_j^*$ у відповідності до виразів [19]:

– функція впевненості (довіри) $Bel : D^{\Omega} \to [0;1]$:

$$Bel(\theta) = \sum_{b_j^{*i} \subseteq \theta, b_j^{*i} \in B_j^*} m(b_j^{*i}); \qquad (9)$$

– функція правдоподібності $Pl : D^{\Omega} \to [0;1]$:

$$Pl(\theta) = \sum_{b_j^{*i} \cap \theta \neq \varnothing, b_j^{*i} \in B_j^*} m(b_j^{*i}). \qquad (10)$$

Значення функції $Bel(\theta)$ відображають ступінь підтримки, що надається підмножині θ.

Значення функції $Pl(\theta)$ відображають ступінь потенційної підтримки, яка може бути надана підмножині θ.

Формування інтервалів $[Bel(\{t_f^{*j}\}), Pl(\{t_f^{*j}\})$ для кожного $t_f^{*j} \in T_j^*$, $f = \overline{1,v}$.

4 ЕКСПЕРИМЕНТИ

Продемонструємо запропоновані вище підходи на прикладі вирішення задачі синтезу групових рішень щодо значень лінгвістичної змінної β = «Якість», яка може набувати наступні значення $T(\beta)$ = {«низька», «середня», «висока», «дуже висока»}.

Приклад 1. Припустимо, група експертів $\mathrm{E} = \{E_i \mid i = \overline{1,n}\}$, $n = 3$, оцінюючи значення заданої лінгвістичної змінної β для заданої сукупності аналізованих об'єктів $O = \{o_j \mid j = \overline{1,z}\}$, $z = 3$, сформувала профілі ЕП P = <B, M>, де $\mathrm{B} = \{B_i \mid i = \overline{1,n}\}$, $B_i = \{b_j^i \mid j = \overline{1,z}\}$. Результати експертного опитування наведені у табл.1.

Таблиця 1 – Профілі ЕП (приклад 1)

P	E_1				E_2			
	t_1	t_2	t_3	t_4	t_1	t_2	t_3	t_4
o_1		0,2	0,8				0,8	0,2
o_2	0,15	0,85			0,6	0,4		
o_3			0,3	0,7		0,6	0,4	
P	E_3							
	t_1	t_2	t_3	t_4				
o_1		0,7	0,3					
o_2	0,7	0,3						
o_3			0,6	0,4				

Таблиця 1 відображає лише суб'єктивні судження 3 експертів відносно значення лінгвістичної змінної β для заданої множини аналізованих об'єктів $O = \{o_j \mid j = \overline{1,3}\}$.

Приклад 2. Припустимо, група експертів $\mathrm{E} = \{E_i \mid i = \overline{1,n}\}$, $n = 3$, оцінюючи значення заданої лінгвістичної змінної β для аналізованого об'єкту o_1, сформувала профілі експертних переваг виду P = <B, H>.

Результати експертного опитування наведені у табл. 2.

DOI 10.15588/1607-3274-2024-2-11

Таблиця 2 – Профілі ЕП (приклад 2)

P	E_1		E_2		E_3	
	$Y_k \subseteq b_j^1$	$X_k \subseteq h_j^1$	$Y_k \subseteq b_j^2$	$X_k \subseteq h_j^2$	$Y_k \subseteq b_j^3$	$X_k \subseteq h_j^3$
o_1	$\{t_2\}$	4	$\{t_3\}$	8	$\{t_2\}$	7
	$\{t_3\}$	8	$\{t_4\}$	2	$\{t_3, t_4\}$	2

Значення $X_k \in h_j^i$ визначались за 10-ти бальною шкалою (0 – відповідає найнижчому ступеню упевненості; 10 – експерт абсолютно впевнений в своєму виборі).

Приклад 3. Розглянемо приклад бінарної класифікації за методом нечіткого ДР при вирішенні наступної задачі: необхідно оцінити можливість надання кредиту потенційному позичальнику – фізичній особі. В якості атрибутів класифікації розглядаються: a_1 – кредитний рейтинг, який визначається на основі скорингової (бальної) оцінки кредитоспроможності потенційного позичальника; a_2 – платоспроможність позичальника (рівень середньомісячного доходу за останні шість місяців).

Атрибут-рішення d_c має два значення: «Видати кредит» – 1 / «Відмовити у видачі кредиту» – 0.

Припустимо, атрибут a_1 = «Кредитний рейтинг» може приймати значення t_1 = «низький», t_2 = «середній», t_3 = «високий»; атрибут a_2 = «Платоспроможність» може приймати значення t_1 = «низька», t_2 = «середня», t_3 = «висока».

За результатами обробки персональних даних п’яти потенційних позичальників було побудовано скорингову карту та отримано чисельні оцінки атрибутів a_1 та a_2. На основі проведеного експертного опитування групою із трьох експертів за отриманими чисельними оцінками класифікаційних атрибутів було визначено ступінь належності кожного об’єкту (потенційного позичальника) до відповідних значень атрибутів.

Результати експертної оцінки наведені в табл. 3.

Таблиця 3 – Профілі ЕП за атрибутами (лінгвістичними змінними) a_1 та a_2

P	Атрибут a_1									Атрибут a_2									d_c
	E_1			E_2			E_3			E_1			E_2			E_3			
	t_1	t_2	t_3	t_1	t_2	t_3	t_1	t_2	t_3	t_1	t_2	t_3	t_1	t_2	t_3	t_1	t_2	t_3	
a_1	0,2	0,8		0,45	0,55		0,6	0,4		0,6	0,4		0,4	0,6		0,7	0,3		0
a_2	0,15	0,85		0,6	0,4			0,3	0,7		0,35	0,65	0,4	0,6			0,3	0,7	1
a_3		0,3	0,7		0,4	0,6	0,2	0,8			0,2	0,8		0,6	0,4		0,7	0,3	1
a_4	0,8	0,2		0,35	0,65		0,3	0,7			0,35	0,65		0,8	0,2		0,4	0,6	1
a_5	0,3	0,7		0,55	0,45		0,65	0,35		0,3	0,7		0,55	0,45		0,65	0,35		0

5 РЕЗУЛЬТАТИ

Розглянемо реалізацію пропонованих алгоритмів для синтезу групового рішення стосовно значення лінгвістичної змінної β для об’єкту o_1.

Аналізуючи дані таблиці 1 можна бачити, що для $o_1 \in O$ на основі множини значень T = {t_1, t_2, t_3, t_4}, що може приймати лінгвістична змінна β = «Якість», групою експертів була сформована сукупність $B_1^* = \{b_1^i\}$ і сукупність оцінок $M_1^* = \{\mathrm{m}_1^i\}$, $i = \overline{1,n}$, де

$$b_1^1 = \{\{t_2\},\{t_3\}\}; \qquad \mathrm{m}_1^1 = \{0{,}2,\ 0{,}8\};$$
$$b_1^2 = \{\{t_3\},\{t_4\}\}; \qquad \mathrm{m}_1^2 = \{0{,}8,\ 0{,}2\}$$
$$b_1^3 = \{\{t_2\},\{t_3\}\}; \qquad \mathrm{m}_1^3 = \{0{,}7,\ 0{,}3\}.$$

На основі значень множини B_1^* сформуємо множину $T_1^* = \{t_2, t_3, t_4\}$.

Таблиця 4 – Ступінь перетину виділених експертами підмножин (приклад 1)

B	**Експерт** E_2		
	b_j^i	t_3	t_4
Експерт E_1	t_2	$E_1(t_2) \cap E_2(t_2) = \varnothing$	$E_1(t_2) \cap E_2(t_4) = \varnothing$
	t_3	$E_1(t_3) \cap E_2(t_3) = t_3$	$E_1(t_3) \cap E_2(t_4) = \varnothing$

Виходячи із даних табл. 4 маємо три локальні конфлікти: $E_1(t_2) \cap E_2(t_3)$, $E_1(t_2) \cap E_2(t_4)$ та $E_1(t_3) \cap E_2(t_4)$.

Розрахуємо комбіновані значення основної маси ймовірності виділених підмножин за привалом (6):

$m_{123}\{t_2\} = 0{,}17;$ $m_{123}\{t_3\} = 0{,}77;$ $m_{123}\{t_4\} = 0{,}06.$

Таким чином, маємо $\mathrm{B}^{gr} = \{t_2, t_3, t_4\}$, $\mathrm{M}^{gr} = \{0{,}17;\ 0{,}77;\ 0{,}06\}$ відповідно.

За даними таблиці 2 експертами була сформована сукупність $B_1^* = \{b_1^i\}$ і сукупність оцінок $O_1^* = \{o_1^i\}$, $i = \overline{1,n}$, де

$$b_1^1 = \{\{t_2\},\{t_3\}\}; \qquad o_1^1 = \{4;\ 8\};$$
$$b_1^2 = \{\{t_3\},\{t_4\}\}; \qquad o_1^2 = \{8;\ 2\};$$
$$b_1^3 = \{\{t_2\},\{t_3, t_4\}\}; \qquad o_1^3 = \{7;\ 2\}.$$

На основі значень множини B_1^* сформуємо множину $T_1^* = \{t_2, t_3, t_4\}$.

Розрахована основна маса ймовірності виділених фокальних елементів відповідно до виразу (7):

E_1: $m(t_2) = 0{,}3$; $m(t_3) = 0{,}6$; $m(\Omega) = 0{,}1$;
E_2: $m(t_3) = 0{,}7$; $m(t_4) = 0{,}18$; $m(\Omega) = 0{,}12$;
E_3: $m(t_2) = 0{,}67$; $m(t_3, t_4) = 0{,}19$; $m(\Omega) = 0{,}14$.

DOI 10.15588/1607-3274-2024-2-11

Комбінована основної маси ймовірності виділених підмножин визначена у відповідності до (6):

$m_{123}\{t_2\} = 0{,}316$; $m_{123}\{t_3\} = 0{,}63$;
$m_{123}\{t_4\} = 0{,}05$; $m_{123}\{t_3, t_4\} = 0{,}0023$;
$m_{123}\{\Omega\} = 0{,}0017$.

Значення функцій (9) та (10) для кожного елемента множини T_1^*:

$$t_2: \begin{cases} Bel(\{t_2\}) = 0{,}316; \\ Pl(\{t_2\}) = 0{,}3177; \end{cases} \quad t_3: \begin{cases} Bel(\{t_3\}) = 0{,}63; \\ Pl(\{t_3\}) = 0{,}634; \end{cases}$$

$$t_4: \begin{cases} Bel(\{t_4\}) = 0{,}05; \\ Pl(\{t_4\}) = 0{,}054. \end{cases}$$

Із наведених результатів видно, що найбільше значення функції довіри та правдоподібності отримав терм t_2; найменше – терм t_4.

Для переходу від інтервальних до точкових оцінок введемо коефіцієнт песимізму $\gamma \in [0,1]$:

$$\gamma \cdot Bel(\theta) + (1-\gamma) \cdot Pl(\theta), \qquad (11)$$

Отримані точкові оцінки можуть бути нормовані та приведені до одиничного інтервалу. При $k = 0{,}6$ маємо:

$\mu(t_2) = 0{,}32$; $\mu(t_3) = 0{,}63$; $\mu(t_4) = 0{,}05$.

На основі запропонованого в роботі підходу щодо визначення агрегованих експертних оцінок значень функції належності, за результатами обробки даних табл. 3 було побудовано агреговані експертні оцінки, табл. 5.

Таблиця 5 – Агреговані експертні оцінки значень класифікаційних атрибутів

p^{gr}	a_1			a_2			d_c
	t_1	t_2	t_3	t_1	t_2	t_3	
a_1	0,39	0,61		0,65	0,35		0
a_2	0,12	0,7	0,18	0,17	0,39	0,44	1
a_3	0,055	0,56	0,385		0,54	0,46	1
a_4	0,43	0,57			0,51	0,49	1
a_5	0,53	0,47		0,32	0,56	0,12	0

Побудоване за вихідними даними табл. 5 ДР на основі алгоритму [10] зображено на рис. 1.

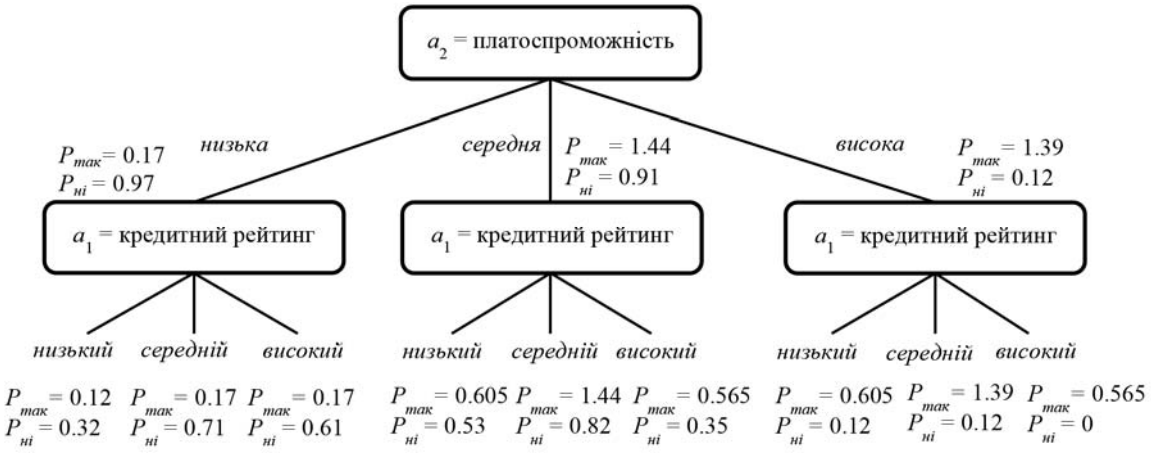


Рисунок 1 – Графічне подання побудованого нечіткого ДР

6 ОБГОВОРЕННЯ

Запропонована в роботі процедура визначення значень функції належності на основі групової експертної оцінки грунтується на математичному апараті нечіткої логіки та теорії правдоподібних та парадоксальних міркувань. Агрегування ЕП (свідоцтв) відбувається шляхом їх комбінування на основі обраного правила [19, 25–27]. Основною проблемою, що виникає в процесі комбінування ЕС, отриманих на основі двох або більше незалежних груп свідоцтв (експертів), є поводження з конфліктами. Причиною виникнення конфліктів є неузгодженість окремих груп ЕС, коли окремі вихідні ЕС (фокальні елементи) не перетинаються.

Виділяють локальний конфлікт, який виникає в результаті порожнього перетину двох вихідних фокальних елементів, і глобальний (сума всіх локальних конфліктів). Якщо вихідні фокальні елементи перетинаються частково, то комбінована маса ймовірності віддається підмножині, яка є результатом такого перетину. Таким чином, результуючий фокальний елемент безпосередньо залежить від ступеня перетину вихідних фокальних елементів – чим вищий ступінь перетину вихідних фокальних елементів, тим менше втрачається вихідної інформації (менше виникає локальних конфліктів) і тим достовірнішими будуть результати комбінування.

Для коректного поводження з конфліктними масами ймовірності у роботі запропоновано використовувати правила комбінування, що дозволяють враховувати ступінь перетину вихідних фокальних елементів і коректно перерозподіляти конфліктні маси ймовірності на підмножини, що залучені до локальних конфліктів [19]. Правила перерозподілу конфліктів PCR1-PCR5 можуть бути використані як на моделі Шейфера [25–27], так і на гібридній моделі Дезера-

DOI 10.15588/1607-3274-2024-2-11

Смарандаке [19]. Правило PCR5 є єдиним правилом, застосування якого дозволяє розділити кожну локальну конфліктну масу ймовірності на частки, які пропорційно перерозподіляються на підмножини, залучені до конфлікту, у відповідності до основних мас ймовірності вихідних фокальних елементів. В цьому випадку не відбувається втрати інформації (маси ймовірності, що відповідає порожнім перетинам фокальних елементів), також не відбувається «розмиття» комбінованих мас ймовірності на результуючих фокальних елементах (утворених шляхом об'єднання вихідних підмножин), як при використанні правил на основі диз'юнктивного консенсусу [25]. Більшість правил комбінування, що базуються на принципах кон'юнктивного консенсусу [25], також мають ряд недоліків, до яких можна віднести втрату частини інформації пов'язаної з вихідними фокальними елементами, що не перетинаються, (така маса ймовірності може відноситись до порожньої множини; до основи задачі або використовуватися для нормування отриманих результатів); так зване «стиснення» вихідних фокальних елементів (крім ситуації, коли вони ідентичні один одному) при утворенні результуючих підмножин, отриманих шляхом перетину вихідних фокальних елементів; не береться до уваги ступінь перетину вихідних фокальних елементів (за винятком правила Жанга).

Втрату вихідної інформації у процесі комбінування можна зменшити за рахунок вибору оптимального порядку комбінування ЕС з огляду на міру близькості між ними [20–23].

Ще одним способом отримання агрегованої оцінки у разі відсутності прийнятного рівня узгодженості є виявлення та виключення конфліктних ЕС, або розбиття вихідної сукупності ЕС на кілька підгруп, усередині яких свідоцтва характеризуються прийнятним рівнем узгодженості [28].

ВИСНОВКИ

У роботі розглянуто задачу визначення значень функції належності аналізованої сукупності об'єктів до визначених термів лінгвістичних змінних, що ставляться у відповідність класифікаційним ознакам у методі нечіткого ДР в умовах групового вибору. Розглянуто два випадки: задані безпосередньо числові оцінки функції належності групою експертів; отримано індивідуальні ЕП щодо ступеня належності об'єкта деякому вербальному значенню лінгвістичної змінної асоційованої з класифікаційним атрибутом. Синтез групового рішення виконано на основі математичного апарату ТДС.

Наукова новизна отриманих результатів полягає в тому, що дістали подальшого розвитку моделі та методи структуризації та синтезу групових рішень в умовах нечіткої експертної інформації.

Практична цінність полягає в тому, що запропонований підхід, при синтезі групового рішення, дозволяє отримувати достовірніші результати комбінування ЕС різної структури за рахунок застосування правила PCR5, розподіляючи конфліктну масу впевненості на підмножини, залучені до локальних конфліктів. Цей підхід можна застосувати для агрегування як узгоджених, так і суперечливих ЕС.

Перспективи подальших досліджень полягають у дослідження можливості застосування правил комбінування ЕС при побудові ДР в умовах інтервальної невизначеності.

ПОДЯКИ

Робота виконана за підтримки іменної стипендії Верховної Ради України для молодих учених – докторів наук за 2023 рік.

ЛІТЕРАТУРА

1. Decision trees as a predictive model in digital marketing / [C. Páez-Quinde, W. Llerena-Llerena, F. Zúñiga-Vásquez, M. P. Silva], Ranganathan G., Allioui Y., Piramuthu S. (eds) // Soft Computing for Security Applications. ICSCS 2023. Advances in Intelligent Systems and Computing. – Springer, Singapore. – 2023. – Vol. 1449. – P. 403–414. DOI: 10.1007/978-981-99-3608-3_28
2. Overview of use of decision tree algorithms in machine learning / [A. Navada, A. N. Ansari, S. Patil, B. A. Sonkamble] // Control and System Graduate Research Colloquium(ICSGRC 2011): Graduate Research Colloquium, Shah Alam, Malaysia, 27–28 June 2011: proceedings. – Shah Alam : IEEE, 2011. – P. 37–42. DOI: 10.1109/ICSGRC.2011.5991826.
3. Rokach L. Data mining with decision trees. Theory and Applications / L. Rokach, O. Maimon. – London : World Scientific Publishing Co, 2008. – 264 p. DOI: 10.1142/9097.9
4. Bramer M. Avoiding overfitting of decision trees / M. Bramer. // Principles of data mining. Undergraduate topics in computer science. – London : Springer, 2013. – P. 121–136. DOI: 10.1007/978-1-4471-4884-5_9
5. Fuzzy decision trees in medical decision making support system / [V. Levashenko, P. Hrkut, A. Kovalenko, L. Kurmasheva] // Modern technologies of biomedical engineering: the 1st Internationals scientific and technical conference, Odesa, Ukraine, 25−27 May 2022: proceedings. Odesa, 2022. – P. 190–198.
6. Classification of cancer data: analyzing gene expression data using a fuzzy decision tree algorithm / [S. A. Ludwig, S. Picek, D. Jakobovic], Kahraman C., Topcu Y. I. (eds) // Operations research applications in health care management. – Cham : Springer. – 2018. – Vol. 262. – P. 327–347. DOI: 10.1007/978-3-319-65455-3_13
7. Zhang Z. Applications of the decision tree in business field / Z. Zhang // Economic Management and Cultural Industry (ICEMCI 2021): the 3rd International Conference, Guangzhou, China, 22–24 October 2021: proceedings. Atlantis Press International B. V., 2011. – P. 926–929. DOI: 10.2991/assehr.k.211209.151
8. Гофман Є. О. Еволюційний метод синтезу дерев рішень / Є. О. Гофман, А. О. Олійник, С. О. Субботін // Штучний інтелект. – 2011. – №2. – С. 6–14.
9. Fuzzy decision trees / [A. Altay, D. Cinar, Kahraman C., Kabak Ö. (eds)] // Fuzzy statistical decision-making: theory and applications. – 2016. – Vol. 343. – P. 221–261. DOI: 10.1007/978-3-319-39014-7_13
10. Janikow C. Z. Fuzzy decision trees: issues and methods / C. Z. Janikow // IEEE Transactions on Systems, Man and

DOI 10.15588/1607-3274-2024-2-11

Cybernetics. – 1998. – Vol. 28(1). – P. 1–14. DOI: 10.1109/3477.658573
11. Quinlan J. R. Induction on decision tree / J. R. Quinlan // Machine Learning. – 1986. – Vol. 1. – P. 81–106. DOI: 10.1007/ BF00116251
12. Kantarci S. A fuzzy ID3 induction for linguistic data sets / S. Kantarci, E. Nasibov // International Journal of Intelligent Systems. –2018. – Vol. 33. – P. 858–878. DOI: 10.1002/int.21971.
13. Quinlan J. R. C4.5: Programs for machine learning / J. R. Quinlan. – San Mateo: Morgan Kaufmann Publishers, 1993. – 312 p.
14. Classification and regression trees / [L. Breiman, J. H. Friedman, R. A. Olsen, C. J. Stone]. – California : Wadsworth & Brooks, 1984. – 368 p.
15. Klir G. J. Fuzzy sets and fuzzy logic: theory and applications / G. J. Klir , B. Yuan. – NJ, Prentice Hall; Upper Saddle River, 1995. – 592 p.
16. Peckol J. K. Introduction to fuzzy logic / J. K. Peckol. – Hoboken, NJ : Wiley, 2021. – 287 p.
17. Kondratenko Yu. Hesitant fuzzy information processing based on the generalized aggregation of resulting trapezoidal linguistic terms / Yu. Kondratenko, G. Kondratenko, I. Sidenko // ICT in Education, Research and Industrial Applications (ICTERI-2019): the 15th International Conference, Kherson, Ukraine, 12–15 June 2019: CEUR workshop proceedings. Integration, Harmonization and Knowledge Transfer. – 2019. – Vol. I. – P. 479–484.
18. Sancho-Royo A. Methods for the construction of membership functions / A. Sancho-Royo, J. Verdegay // International Journal of Intelligent Systems. – 1999. – Vol. 14. – P. 1213–1230. DOI: 10.1002/(SICI)1098-111X(199912)14:123.0.CO;2-5.
19. Smarandache F. Advances and applications of DSmT for information fusion / F. Smarandache, J. Dezert. – Vol. 1. – Rehoboth : American Research Press, 2004. – 760 p.
20. Bhattacharyya A. On a measure of divergence between two statistical populations defined by their probability distribution / A. Bhattacharyya // Bulletin of the Calcutta Mathematical Society. – 1943. – Vol. 35. – P. 99–110.
21. Cuzzolin F. A geometric approach to the theory of evidence / F. Cuzzolin // Transactions on Systems, Man, and Cybernetics (Part C: Applications and Reviews). – 2007. – Vol. 38(4). – P. 522–534. DOI: 10.1109/TSMCC.2008.919174
22. Jousselme A. L. A new distance between two bodies of evidence / A. L. Jousselme, D. Grenier, E. Boss´e // Information Fusion. – 2001. – Vol. 2. – P. 91–101. DOI: 10.1016/S1566-2535(01)00026-4
23. Tessem B. Approximations for efficient computation in the theory of evidence / B. Tessem // Artificial Intelligence. – 1993. – Vol. 61. – P. 315–329. DOI: 10.1016/0004-3702(93)90072-J
24. Beynon M. J. The Dempster–Shafer theory of evidence: an alternative approach to multicriteria decision modeling / M. J. Beynon , B. Curry , P. Morgan // Omega. – 2000. – Vol. 28(1). – P. 37–50.
25. Sentz K. Combination of evidence in Dempster-Shafer theory. Technical report SAND 2002-0835 / K. Sentz, S. Ferson. – Albuquerque : Sandia National Laboratories, 2002. – 94 p.
26. Dempster A. P. Upper and lower probabilities induced by a multi-valued mapping / A. P. Dempster // Annals of Mathematical Statistics. – 1967. – Vol. 38(2). – P. 325–339. DOI: 10.1214/aoms/1177698950
27. Shafer G. A mathematical theory of evidence / G. Shafer. – Princeton : Princeton University Press, 1976. – 297 p.
28. Davydenko Ye. O. Development of technique for structuring of group expert assessments under uncertainty and inconcistancy / Ye. O. Davydenko, A. V. Shved, N. V. Honcharova // Radio Electronics, Computer Science, Control. – 2023. – Vol. 30(4). – P. 30–38. DOI: 10.15588/1607-3274-2023-4-3

Received 19.02.2024.
Accepted 24.04.2024.

UDC 004.827:519.816

DEVELOPMENT OF TECHNIQUE FOR DETERMINING THE MEMBERSHIP FUNCTION VALUES ON THE BASIS OF GROUP EXPERT ASSESSMENT IN FUZZY DECISION TREE METHOD

Shved A. V. – Dr. Sc., Professor, Professor of Department of Software Engineering, Petro Mohyla Black Sea National University, Mykolayiv, Ukraine.

ABSTRACT

Context. Recently, fuzzy decision trees have become widely used in solving the classification problem. In the absence of objective information to construct the membership function that shows the degrees of belongingness of elements to tree nodes, the only way to obtain information is to involve experts. In the case of group decision making, the task of aggregation of experts' preferences in order to synthesize a group decision arises.

The object of the study is group expert preferences of the degree of belonging (membership function) of an element to a given class, attribute, which require structuring and aggregation in the process of construction and analysis of a fuzzy decision tree.

Objective. The purpose of the article is to develop a methodology for determining the membership degree of elements to a given class (attribute) based on the group expert assessment in the process of construction and analysis of fuzzy decision trees.

Method. The research methodology is based on the complex application of the mathematical apparatus of the theory of plausible and paradoxical reasoning and methods of fuzzy logic to solve the problem of aggregating fuzzy judgments of the classification attribute values in the process of construction and analysis of a fuzzy decision tree. The proposed approach uses the mechanism of combination of expert evidences (judgments), formed within the framework of the Dezert-Smarandache hybrid model, based on the PCR5 proportional conflict redistribution rule to construct a group solution.

Results. The issues of structuring fuzzy expert judgments are considered and the method of synthesis of group expert judgments regarding the values of membership degree of elements to classification attributes in the process of construction and analysis of fuzzy decision trees has been proposed.

DOI 10.15588/1607-3274-2024-2-11

Conclusions. The models and methods of structuring and synthesis of group decisions based on fuzzy expert information were further developed. In contrast to the existing expert methods for the construction of membership function in context of group decision making, the proposed approach allows synthesizing a group decision taking into account the varying degree of conflict mass in the process of combination of original expert evidenced. This approach allows to correctly aggregate both agreed and contradictory (conflicting) expert judgments.

KEYWORDS: the theory of plausible and paradoxical reasoning, fuzzy decision trees, proportional conflict redistribution rule.

REFERENCES

1. Páez-Quinde C., Llerena-Llerena W., Zúñiga-Vásquez F., Silva M. P. Ranganathan G., EL Allioui Y., Piramuthu S. (eds), Decision trees as a predictive model in digital marketing, *Soft Computing for Security Applications. ICSCS 2023. Advances in Intelligent Systems and Computing.* Springer, Singapore, 2023, Vol. 1449, pp 403–414. DOI: 10.1007/978-981-99-3608-3_28
2. Navada A., Ansari A. N., Patil S., Sonkamble B. A. Overview of use of decision tree algorithms in machine learning, *Control and System Graduate Research Colloquium* (ICSGRC 2011*) : Graduate Research Colloquium, Shah Alam, Malaysia, 27–28 June 2011: proceedings.* Shah Alam, IEEE, 2011, pp. 37–42. DOI: 10.1109/ICSGRC.2011.5991826.
3. Rokach L., Maimon O. Data mining with decision trees. Theory and Applications. London, World Scientific Publishing Co, 2008, 264 p. DOI: 10.1142/9097.9
4. Bramer M. Avoiding overfitting of decision trees. In: Principles of data mining, *Undergraduate topics in computer science.* London, Springer, 2013, pp. 121–136. DOI: 10.1007/978-1-4471-4884-5_9
5. Levashenko V., Hrkut P., Kovalenko A., Kurmasheva L. Fuzzy decision trees in medical decision making support system, *Modern technologies of biomedical engineering: the 1st Internationals scientific and technical conference, Odesa, Ukraine, 25−27 May 2022, proceedings*. Odesa, 2022, pp. 190–198.
6. Ludwig S. A., Picek S., Jakobovic D., Kahraman C., Topcu Y. I. (eds) Classification of cancer data: analyzing gene expression data using a fuzzy decision tree algorithm. Operations research applications in health care management. Cham, Springer, 2018, Vol. 262, pp. 327–347. DOI: 10.1007/978-3-319-65455-3_13
7. Zhang Z. Applications of the decision tree in business field, *Economic Management and Cultural Industry (ICEMCI 2021): the 3rd International Conference, Guangzhou, China, 22–24 October 2021: proceedings.* Atlantis Press International B. V., 2011. pp. 926–929. DOI: 10.2991/assehr.k.211209.151
8. Gofman E. A., Oliinyk A. A., Subbotin S. A. Evolutionary method of decision trees synthesis, *Artificial Intelligence*, 2011, №2, pp. 6–14.
9. Altay A., Cinar D. Fuzzy decision trees. In Kahraman C., Kabak Ö. (eds) Fuzzy statistical decision-making: theory and applications, 2016, Vol. 343, pp. 221–261. DOI: 10.1007/978-3-319-39014-7_13
10. Janikow C. Z. Fuzzy decision trees: issues and methods, *IEEE Transactions on Systems, Man and Cybernetics*, 1998, Vol. 28(1), pp. 1–14. DOI: 10.1109/3477.658573
11. Quinlan J. R. Induction on decision tree, *Machine Learning*, 1986, Vol. 1, pp. 81–106. DOI: 10.1007/ BF00116251
12. Kantarci S., Nasibov E. A fuzzy ID3 induction for linguistic data sets, *International Journal of Intelligent Systems*, 2018, Vol. 33, pp. 858–878. DOI: 10.1002/int.21971.
13. Quinlan J. R. C4.5: Programs for machine learning. San Mateo, Morgan Kaufmann Publishers, 1993, 312 p.
14. Breiman L., Friedman J. H., Olsen R. A., Stone C. J. Classification and regression trees. California, Wadsworth & Brooks, 1984. 368 p.
15. Klir G. J., Yuan B. Fuzzy sets and fuzzy logic: theory and applications. NJ, Prentice Hall; Upper Saddle River, 1995, 592 p.
16. Peckol J. K. Introduction to fuzzy logic. Hoboken, NJ, Wiley, 2021, 287 p.
17. Kondratenko Yu., Kondratenko G., Sidenko I. Hesitant fuzzy information processing based on the generalized aggregation of resulting trapezoidal linguistic terms, *ICT in Education, Research and Industrial Applications* (ICTERI-2019): the 15th International Conference, Kherson, Ukraine, 12–15 June 2019: CEUR workshop proceedings. Integration, Harmonization and Knowledge Transfer, 2019, Vol. I. P. 479–484.
18. Sancho-Royo A., Verdegay J. Methods for the construction of membership functions, *International Journal of Intelligent Systems*, 1999, Vol. 14, pp. 1213–1230. DOI: 10.1002/(SICI)1098-111X(199912)14:123.0.CO;2-5.
19. Smarandache F., Dezert J. Advances and applications of DSmT for information fusion. Rehoboth, American Research Press, 2004, Vol. 1, 760 p.
20. Bhattacharyya A. On a measure of divergence between two statistical populations defined by their probability distribution, *Bulletin of the Calcutta Mathematical Society*, 1943, Vol. 35, pp. 99–110.
21. Cuzzolin F. A geometric approach to the theory of evidence, *Transactions on Systems, Man, and Cybernetics*, Part C: Applications and Reviews, 2007, Vol. 38(4), pp. 522–534. DOI: 10.1109/TSMCC.2008.919174
22. Jousselme A. L., Grenier D., Boss´e E. A new distance between two bodies of evidence, *Information Fusion*, 2001, Vol. 2, pp. 91–101. DOI: 10.1016/S1566-2535(01)00026-4
23. Tessem B. Approximations for efficient computation in the theory of evidence, *Artificial Intelligence*, 1993, Vol. 61, pp. 315–329. DOI: 10.1016/0004-3702(93)90072-J
24. Beynon M. J., Curry B., Morgan P. The Dempster–Shafer theory of evidence: an alternative approach to multicriteria decision modeling, *Omega*, 2000, Vol. 28(1), pp. 37–50.
25. Sentz K., Ferson S. Combination of evidence in Dempster-Shafer theory. Technical report SAND 2002-0835. Albuquerque, Sandia National Laboratories, 2002, 94 p.
26. Dempster A. P. Upper and lower probabilities induced by a multi-valued mapping, *Annals of Mathematical Statistics*, 1967, Vol. 38(2), pp. 325–339. DOI: 10.1214/aoms/1177698950
27. Shafer G. A mathematical theory of evidence. Princeton: Princeton University Press, 1976, 297 p.
28. Davydenko Ye. O., Shved A. V., Honcharova N. V. Development of technique for structuring of group expert assessments under uncertainty and inconcistancy, *Radio Electronics, Computer Science, Control*, 2023, Vol. 30(4), pp. 30–38. DOI: 10.15588/1607-3274-2023-4-3

DOI 10.15588/1607-3274-2024-2-11

p-ISSN 1607-3274 Радіоелектроніка, інформатика, управління. 2024. № 2
e-ISSN 2313-688X Radio Electronics, Computer Science, Control. 2024. № 2

ПРОГРЕСИВНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

PROGRESSIVE INFORMATION TECHNOLOGIES

UDC 004.93, 004.8

METHOD OF IMPERATIVE VARIABLES FOR SEARCH AUTOMATION OF TEXTUAL CONTENT IN UNSTRUCTURED DOCUMENTS

Boiko V. O. – Assistant of the Department of Software Engineering, Khmelnytskyi National University, Khmelnytskyi, Ukraine.

ABSTRACT

Context. Currently, there are a lot of approaches that are used for textual search. Nowadays, methods such as pattern-matching and optical character recognition are highly used for retrieving preferred information from documents with proven effectiveness. However, they work with a common or predictive document structure, while unstructured documents are neglected. The problem – is automating the textual search in documents with unstructured content. The object of the study was to develop a method and implement it into an efficient model for searching the content in unstructured textual information.

Objective. The goal of the work is the implementation of a rule-based textual search method and a model for seeking and retrieving information from documents with unstructured text content.

Method. To achieve the purpose of the research, the method of rule-based textual search in heterogenous content was developed and applied in the appropriately designed model. It is based on natural language processing that has been improved in recent years along with a new generative artificial intelligence becoming more available.

Results. The method has been implemented in a designed model that represents a pattern or a framework of unstructured textual search for software engineers. The application programming interface has been implemented.

Conclusions. The conducted experiments have confirmed the proposed software's operability and allow recommendations for use in practice for solving the problems of textual search in unstructured documents. The prospects for further research may include the improvement of the performance using multithreading or parallelization for large textual documents along with the optimization approaches to minimize the impact of OpenAI application programming interface content processing limitations. Furthermore, additional investigation might incorporate extending the area of imperative variables usage in programming and software development.

KEYWORDS: textual search, unstructured text documents, natural language processing, rule-based search, generative artificial intelligence, imperative variables.

ABBREVIATIONS

OCR is an optical character recognition;
API is an application programming interface;
ZOCR is a zonal optical character recognition;
NLP is a natural language processing;
AI is an artificial intelligence;
GPT is a generative pre-trained transformer.

NOMENCLATURE

T is a set of unstructured text documents;
R is a set of search rules (prompts);
D is a description of a purpose for the model;
X is input data that consists of unstructured text documents, search rules, purpose descriptions, and sample document content;
Y' is output data that represents extracted data points from text documents;
f is a general representation of a function that performs data extraction based on input parameters;
Y is a sample response, ground truth;
$L(Y,Y')$ is a Cross-Entropy Loss function;
N is a number of variables or data points to be predicted in each example;
M is the size of the whole dataset;
$y'_{i,j}$ is a predicted value for the j-th data point in the i-th example;
$Y'_{i:i-1}$ is a previously generated set of tokens;
$P()$ is a probability distribution of the subsequent token;
θ is a model parameters;
$\nabla L(\theta)$ is a gradient of the loss function L with respect to the model parameters θ;
$g^{(t)}$ is a gradient of the loss function evaluated at time step t;
$m^{(t)}$ is a first-moment estimate at time step t;
β_1 is a decay rate for the first-moment estimate;
$v^{(t)}$ is a second-moment estimate at time step t;

DOI 10.15588/1607-3274-2024-2-12

β_2 is a decay rate for the second-moment estimate;

$\hat{m}^{(t)}$ is a correction of bias in the first-moment estimate at time step *t*;

$\hat{v}^{(t)}$ is a correction of bias in the second-moment estimate at time step *t*;

$\alpha^{(t)}$ is an adaptive learning rate;

ε is a constant to prevent dividing by zero;

$GPT()$ is a function that performs text generation based on input parameters using a GPT-3.5 Turbo model.

INTRODUCTION

Text searching is a widespread and basic operation for working with document content. The most popular text processing software such as Microsoft Word, PDF Reader, etc. incorporates the standard seeking algorithms into their search capabilities like a keyword or pattern-based search.

On the other hand, they are not able to work with pictures – search and retrieve information from them. In this case, the optical character recognition method could help find the appropriate text and additionally categorize it properly [1].

However, when unstructured documents are taken into account, the above-listed methods will not work, because, for keyword-based, pattern-matching search, or even OCR, the predefined document structure is required, otherwise, the results will be smooth and inaccurate.

The object of study is the process of textual search in documents with unstructured content.

The subject of study is the methods for searching and retrieving information from textual documents.

The known text search approaches and algorithms, described by authors and outlined as a part of different areas of implementation [2, 3] and [5, 6] are inappropriate and not suitable for unstructured textual document processing.

However, several studies [7–12] outline the approach based on NLP to seek appropriate data in documents related to specific areas, but these approaches are not described as a general method of searching data in unstructured documents.

The purpose of the work is to develop a method and incorporate it into an efficient and generic model for textual search in documents without a predefined structure that would be possible to use by software engineers as a framework.

1 PROBLEM STATEMENT

Suppose we have a set of unstructured text documents T, a list of search rules R, and a description D of a field that represents a user context where to find the appropriate information. Text information from documents, descriptions, and criteria are considered a set of tokens which means it could be a different type of textual content, such as a sequence of symbols, words, or sentences.

The task is to develop a method that will perform an accurate rule-based search to find an appropriate set of data points Y' in unstructured documents. The quality of response and performance should not depend on the number of rules and documents that should be processed. This can be represented by the following model:

$$\begin{aligned} X &= \{T, R, D\} \\ f &: X \to Y' \end{aligned}, \qquad (1)$$

where the function f from input X generates an output Y' which represents the found data.

Additionally, a generic search model should be designed and the method of rule-based search should be implemented in the model that will be used to construct a convenient API.

2 REVIEW OF THE LITERATURE

A standard keyword search is usually performed using algorithms like Rabin-Karp or Knuth-Morris-Pratt. These algorithms are often utilized to develop frameworks that detect plagiarism in text documents as described in the study [2]. However, they are not effective in rule-based search, as they can only identify specific patterns of text based on explicitly specified key phrases. Therefore, these algorithms are not suitable for tasks that require more advanced search techniques.

Pattern Matching Search, also known as Regex Search, is a powerful tool that allows for flexible string matching by describing complex patterns. It is widely supported across different programming languages, as it is built into text processing libraries. In a certain publication [3], the authors combined regular expressions with keyword searches to improve web search results. By using keywords as criteria or rules, fragments of information found through pattern matching can be considered as either the criteria value or as a result of the defined criteria. This method provides an effective criteria-based search.

The methods mentioned earlier are useful only if the documents contain text information. However, they cannot be employed for extracting text from images or documents that have only images with text (for instance, scanned copies of pages in PDF format). In such circumstances, the OCR technique serves as a viable alternative for seeking and extracting textual information.

In recent years, many services have emerged that provide the ability to extract textual information from images through API. One such service is Azure OCR, which has gained popularity due to its capability to recognize both printed and handwritten text from images and to distribute information based on the contextual understanding of the document [4]. Other services, including Google Cloud Vision, Amazon Textract, and Tesseract OCR, also offer similar functionality for extracting textual information from images. Furthermore, recent research studies have highlighted the significance of word processing via OCR for historical documents [5]. These studies have demon-

DOI 10.15588/1607-3274-2024-2-12

strated that post-processing techniques can be applied to improve the accuracy and reliability of the OCR results.

The effectiveness of the OCR method for document processing is dependent on the selection of relevant criteria for data extraction. In scenarios where the criteria yield only a small amount of data while a document is voluminous, the OCR approach becomes ineffective, and the processing of the document may require significant memory or technical solutions aimed at reducing the system load. To address this challenge, the Zonal OCR approach has been developed. Unlike the standard OCR, which processes the entire document, ZOCR narrows the areas of text recognition to specific fragments where data extraction is required, thus avoiding the need to process all the text information in the document. In the paper [6], the underlying principles of ZOCR's smart parsing of documents are described. Modern OCR services, discussed earlier, have ZOCR support, making them effective tools for zonal character recognition.

These methods for finding textual information are useful and effective when predefined patterns and criteria are present. Text search based on regular expressions can identify similar character sequences, while optical character recognition is able to recognize characters and categorize text into appropriate groups using automated algorithms.

Both methods have common issues that become apparent when modifications are made to the current implementation. In the first case, developers must incorporate pattern-matching logic based on new business requirements, which may entail defining new or modifying existing search criteria. In the case of regular expressions, some selection rules may not be implementable through Regex. Therefore, an additional search logic alongside the existing one may be required. When considering this issue in the context of optical character recognition, introducing new search criteria may raise the question of retraining the existing model. Instead, a standardized approach may involve scanning the entire document for textual information and applying a pattern-matching search, which again brings us back to the above problem.

In a study [7], a solution encountered in extracting complex text structures, tabular information, and text located in different places was proposed by using NLP. The authors indicated that the ZOCR method was insufficient in extracting such information, and thus, they utilized the spaCy library which provides linguistically complex models for searching for necessary text information. Recent studies have shown that NLP is an important and effective approach in solving problems and offers new opportunities for improving information retrieval processes in text documents, resulting in more accurate and complete results.

After analyzing recent research on NLP [7, 8], it can be concluded that this approach is significant and effective in overcoming the limitations found in previously analyzed methods. The use of NLP can provide more accurate and complete results, simplify the recognition of complex structures, and improve the quality of textual information retrieval. For example, in the study [9] authors implemented an NLP algorithm to extract the presence of social factors from clinical text, which is considered as an unstructured document. The paper [10] also shows the usage of rule-based NLP search for unstructured data in electronic health records. The publication [11] outlines the NLP text extraction from unstructured geoscience reports.

3 MATERIALS AND METHODS

Generative AI refers to a class of AI systems that are specifically designed to generate new and original content, rather than just analyzing existing data or making predictions based on learned patterns. These systems employ various techniques, such as machine learning and deep learning, to produce fresh content that is often comparable, if not identical, to what a human can produce.

The development of generative AI has been significantly advanced by the contribution made by the OpenAI company [12]. Among their notable accomplishments is the ChatGPT model, which has been continuously evolving. OpenAI has made interaction with GPT models accessible through the public OpenAI API, which is widely used in various applications, including chatbots, content creation, and natural language understanding tasks. According to research [13], ChatGPT showcases significant progress in the field of natural language processing and has the potential to revolutionize the way we interact with machines and process natural language data. The model is also capable of maintaining the context of conversations, enabling it to refer to previous messages to provide relevant responses.

The potential of ChatGPT in the field of natural language processing (NLP) and the search approach outlined in a publication [6] have paved the way for leveraging modern generative artificial intelligence (AI) capabilities in the process of rule-based search for textual information. Today NLP mechanisms and capabilities can enable a more efficient and effective search of unstructured textual data.

When communicating with ChatGPT, prompts are used to provide information – input data given to the model for generating responses and can be messages, questions, or any text that provides context or instruction. Users interact with ChatGPT by sending prompts, and the model generates text responses based on the input. It is important to note that the quality and relevance of the responses generated by ChatGPT depend on the clarity and specificity of the prompts. A prompt is a crucial component in interacting with the context-forming model. In the paper [14], the main approaches to prompt engineering, which is the process of forming and correcting prompts, are considered and highlighted. Therefore, by using the correct approaches to form accurate AI queries, it is possible to search for information more effectively. The clearer and more specific the prompt is, the better the search results will be.

Using GPT models can be effective for searching and extracting text from text documents, including unstruc-

DOI 10.15588/1607-3274-2024-2-12

tured ones. Prompts are used to retrieve specific textual information. Correction of output results can be done by providing sample results. To distinguish the extracted information, prompts with specific example results can be marked by a unique identifier or name. All these parts can be represented as variables, that are denoted as rules in formula (1) and are a part of model input data *X*. Since prompts are often used in an imperative form, the variables can be called imperative. Fig. 1. illustrates the general structure of an imperative variable.

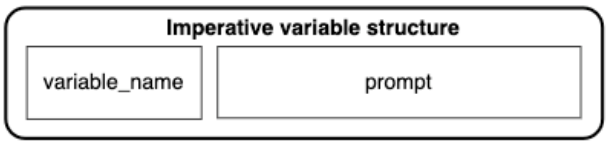


Figure 1 – Imperative variable structure

These variables contain prompts that can represent different rules for filtering, searching data, and even descriptions of how to format the output result which are saved in result variables. To make a model better understand what kind of data it should extract from documents, the sample response for each imperative variable should be defined. This type of data can be called a sample variable as it represents an example of an output for the appropriate imperative variable. The structure of a sample variable is shown in Fig. 2.

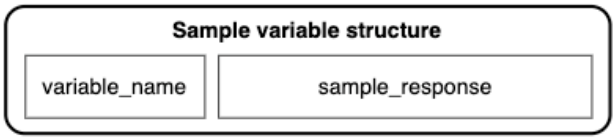


Figure 2 – Sample variable structure

After defining imperative and sample variables, there is a need to set up the chat model, which is described in this instruction [15]. Additionally, a chat assistant should have contextual information that briefly describes an area in which a user works. Imperative variables are included in the user-side messages along with the sample responses for assistance. For more accurate results the sample document can be provided. This document contains an example content to show the assistant what kind of documents it will deal with in case of user-provided documents. These settings are demonstrated in Fig 3.

```
[
    {"role": "system", "content": "Assistant is a language model trained to
    <purpose_description>."},
    {"role": "user", "content": "Can you help with parsing information from a text
    document if I give you the data points? "},
    {"role": "assistant", "content": "Yes. Please provide me with the data points."},
    {"role": "user", "content":<list_of_imperative_variables>},
    {"role": "assistant", "content": "Sure. Please provide me with the text document"},
    {"role": "user", "content": <sample_document_content>},
    {"role": "assistant", "content": <list_of_sample_variables>},
    {"role": "user", "content": "Are you ready for another document?"},
    {"role": "assistant", "content": "Yes, please provide the text document, and I will
    extract the required data points."},
    {"role": "user", "content":<user_document_content>}
]
```

Figure 3 – Messages to set up the assistant

In GPT models during the training stage, several common operations are typically performed, including the specification of a loss function and its minimization through optimization algorithms. In the case of an unstructured document text search task, the loss function can be defined to measure the discrepancy between the model's predictions – the search results and the ground truth relevant information that the model is expected to retrieve from the documents. In the case of text generation tasks the Cross-Entropy Loss function can be used. It is represented by the following formula (2):

$$L(Y,Y') = -\sum_{i=1}^{N} \ln(P(y'_i \mid X, Y'_{i:i-1})). \quad (2)$$

It's important to note that function (2) works only with a single sequence of data, but the more effective way will be enhancing the existing function with the possibility of batch processing, so multiple sequences can be used simultaneously. In the context of batch processing, where multiple sequences are processed simultaneously, the loss function computes the discrepancy between the model's predictions and the ground truth for all sequences in the dataset. To obtain a representative measure of the average discrepancy per sequence in the batch, 1/*M* a coefficient is included. This factor ensures that the loss value is normalized by the number of sequences. Updated formula looks like this (3):

$$L(Y,Y') = -\frac{1}{M}\sum_{j=1}^{M}\sum_{i=1}^{N} \ln(P(y'_{i,j} \mid X, Y'_{i:i-1})). \quad (3)$$

In (2) the double summation is used since there is a need for batch processing to sum over all tokens in the output sequence Y' and all possible tokens in the vocabulary (the inner summation) to compute the overall loss. This ensures that the discrepancy is considered for each token in the predicted sequence compared to all possible tokens in the vocabulary [16].

After the loss function is defined there is a need to optimize it, because the primary goal during training is to make the model learn to perform more accurate predictions or generate more relevant outputs. Several optimization techniques can be used, but the most effective is the adaptive moment estimation algorithm (Adam) and its variations that are used in GPT models. Adam optimization is performed in several steps [17]. Firstly, the gradient function should be defined. For the current case, it can be represented by the following formula (4):

$$\nabla L(\theta) = -\frac{1}{M}\sum_{j=1}^{M}\sum_{i=1}^{N} \frac{\partial}{\partial\theta} \ln(P(y_{i,j} \mid X, y_{i:i-1})). \quad (4)$$

It computes the partial derivative of the logarithm of the predicted probability of each token in the output sequence given the input and previous tokens, summed over all training examples (documents) and tokens within each

DOI 10.15588/1607-3274-2024-2-12

document. The gradient function with respect to the time step is represented by the following formula (5):

$$g^{(t)} = \nabla L(\theta^{(t-1)}) . \quad (5)$$

Then, update the first (6) and second (7) moment estimation happens:

$$m^{(t)} = \beta_1 m^{(t-1)} + (1-\beta_1) g^{(t)} , \quad (6)$$

$$v^{(t)} = \beta_2 v^{(t-1)} + (1-\beta_2)(g^{(t)} \circ g^{(t)}) . \quad (7)$$

After that, a bias correction for each updated moment estimate is computed (8, 9):

$$\hat{m}^{(t)} = \frac{m^{(t)}}{1-\beta_1{}^t} , \quad (8)$$

$$\hat{v}^{(t)} = \frac{v^{(t)}}{1-\beta_2{}^t} . \quad (9)$$

The final stage is to update the model parameters based on calculated adaptive learning rate and bias-corrected moment estimates (10):

$$\theta^{(t)} = \theta^{(t-1)} - \alpha^{(t)} \frac{\hat{m}^{(t)}}{\sqrt{\hat{v}^{(t)}} + \varepsilon} . \quad (10)$$

The algorithm which includes operations (4–10) is performed several times. The loop works until either the model converges or the maximum number of iterations is reached, whichever comes first.

The mentioned algorithms used in the model training process today are incorporated into GPT language models, like GPT-3.5 Turbo from OpenAI. However, this model does not disclose its optimization techniques, but according to several types of research, the methods shown in this paper are used by some GPT models. Instead of building a custom model that can last long and take a big amount of computing resources to maintain training and deployment processes, the most stable GPT-3.5 Turbo model can be used effectively for tasks such as a rule-based search. Thus, the final formula for getting the search result Y' using the GPT-3.5 Turbo model can be represented as the following (11):

$$Y' = GPT(X) . \quad (11)$$

The imperative variables method can be applied to different documents and to make it generic the appropriate search model has been developed. According to this model, imperative variables along with sample responses are saved in data storage. This model allows us to build custom and flexible templates based on imperative variables for different areas and apply them for rule-based textual search in documents, including unstructured ones. Templates consist of imperative and sample variables with one sample document. Fig 4 illustrates this model which is represented by a sequence diagram. This is a generic approach to serve the NLP-based textual search. Users can define their templates according to their business area, specify the purpose description, and manage templates that contain imperative variables. Additionally, the model can be used to implement the appropriate API.

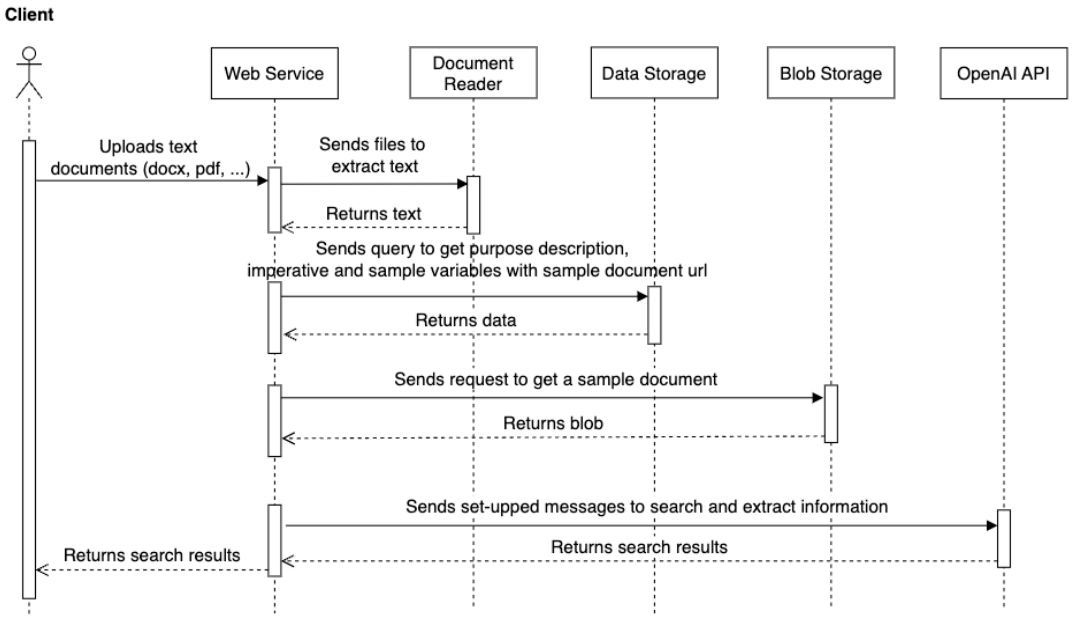


Figure 4 – Sequence diagram of a search model using the imperative variables method

DOI 10.15588/1607-3274-2024-2-12

Compared to the pattern-matching search, the imperative variables search model has several advantages:

1) The model requires only properly described prompts using natural language and sample responses while pattern-matching uses Regular Expression syntax to seek data that is not well-known to common users.

2) Along with a description of what the user needs to find, it can be formally extended by the flexible set of additional requirements, for example, of how it should be formatted in the output or conditions of what to do if the required data is not found without any additional programmed logic.

However, with a large number of imperative variables or large documents, a search can be slower, thus the model needs to be optimized using multithreading or parallelization mechanisms which is a part of further studies.

4 EXPERIMENTS

The model has been implemented to show how the method works. Since it is generic, any area is suitable for this model. So, the Shipping area was selected to conduct the experiments.

According to the model – several imperative and response variables were created in the MySQL database. Also, to properly set up the Open AI assistant, the purpose description was initialized.

For a convenient view data was converted to CSV format. The area information along with imperative and sample variables are listed in Figure 5.

Imperative and sample variables

variable_name	prompt	sample
document_id	Referred to as the bill of lading number, sea waybill number, or a similar abbreviation	AAABBB123456
shipper	This is the party declared as shippers or exporters of the goods, not the carrier the document is produced by	Mattel, Inc
description	Description of the goods being shipped	Fisher-Price
vessel	A ship that is transporing the goods	MSC TUXPAN
scac	The SCAC code is a 4 letter uppercase character string. If the SCAC code is not found, take the first 4 digits of the document_id	CMAU
container_num	The number of container where goods are	CMAU1234000
shipper_ref	Reference number associated with the shipper	1234560000
net	Net weight of the goods that are being shipped by the vessel	1,100.150 KG

Area

area_name	purpose_description	sample_url
Shipping	parse shipping documents	/samples/shipping.pdf

Figure 5 – Variables for text extraction

For better results, there was created a sample shipping document which contains example data for the GPT model. It is saved on Azure blob storage.

To conduct the experiments .NET environment was used and for simplicity there was developed a console application that accepts text documents and on output generates the CSV file with a response.

There are several steps were performed to parse the documents:

1) Application field information and variables are retrieved from the MySQL database.

2) A sample document is downloaded from the Azure Blob storage.

3) The messages are set up and sent to the OpenAI API for processing.

4) The result is obtained in a JSON format and saved to a CSV file.

The search model was implemented in ASP.NET-based Imperative Variable Search Web API which is a general point to access the rule-based search. The list of implemented endpoints is illustrated in Fig. 6.

Figure 6 – Imperative Variable Search API endpoints

Endpoints accept user requests and perform the main functionality of the API. A user first needs to register an account, then create an area with an appropriate name and purpose, populate the set of imperative variables along with sample responses that are related to the area, upload a sample file, and then upload files to perform the rule-based search and obtain the results. The API has a convenient way of interacting using Swagger and can be integrated into different business solutions.

5 RESULTS

15 unstructured shipping documents were selected and processed (18 KB each) with 8 variables. No parsing errors occurred. There are conducted 5 attempts of execution to determine the average time. Time calculation includes retrieving imperative variables and other information from the database, extracting textual content from documents, and batch-sending requests to the OpenAI API. The result of processing documents was saved into a CSV file and illustrated in Fig. 7.

According to time measurement results the average time spent on requests to OpenAI API and the time of overall program execution is approximately 3 seconds which can be considered as an acceptable result (Fig. 8).

DOI 10.15588/1607-3274-2024-2-12

However, OpenAI API has a rate limit that depends on the Tier subscription [18]. The experiment was conducted on Tier 1 which has a limit of 60000 tokens that can be sent per 1 minute, which also keeps the limit of the number of documents that can be processed per this time. Thus, the queue should be used in case of a large number of documents. Additionally, the content length has limits too – for GPT-3.5 Turbo model is 16385 tokens [19], thus larger documents should be split into smaller parts before processing.

Shipping Documents Search Result

document_id	shipper	description	vessel	scac	container_num	shipper_ref	net
XYZUH820572	XYZ Trading Co.	Product XYZ 500ML	XYZ ATLANTIC	MSCU	MSCU9876541	987654321	1,000.00 KG
MEDUUH820571	LMN Corporation	Nautical Nuts and Bolts	MSC ATLANTIC	MSCA	MSCA8907092	987654321	10,500.00 KG
ABCUH820575	ABC Exporters Inc.	Product 500ML	ABC ATLANTIC	HLCU	HLCU9876567	987654321	5,000.00 KG
MEDUUH820571	ABC Trading Co.	Oceanic Organic Coffee	MSC OCEANIA	COSU	COSU9876578	1234567890	3,000.00 KG
MEDUUH820571	ABC Trading Co.	Seafarer's Seafood Mix	MSC OCEANIA	MSCU	MSCU9876521	1234567890	8,000.00 KG
GLHUH820576	Globe Exporters Inc.	Wave Rider Surfboards	GLOBE EXPRESS	ONEY	ONEY9876543	1234567890	5,000.00 KG
XYZUH820573	ABC Trading Co.	Captain's Choice Whiskey	XYZ ATLANTIC	ONEY	ONEY9876545	987654321	8,000.00 KG
MEDUUH820571	ABC Trading Co.	Harbor Lights Candles	MSC OCEANIA	ZIMU	ZIMU4560701	1234567890	2,100.00 KG
MEDUUH820571	ABC Trading Co.	Oceanic Organic Coffee	MSC OCEANIA	ZIMU	ZIMU9876543	1234567890	2,400.00 KG
GLHUH820576	Globe Exporters Inc.	Shipshape Sunglasses	GLOBE EXPRESS	GLHU	MSCU9871234	1234567890	1,000.00 KG
MEDUUH820571	ABC Trading Co.	Coastal Crafts Art Supplies	MSC OCEANIA	MAEU	MAEU9876433	1234567890	10,000.00 KG
MEDUUH820571	ABC Trading Co.	Harbor Lights Candles	MSC OCEANIA	MAEU	MAEU9876501	1234567890	10,500.00 KG
GLHUH820576	Globe Exporters Inc.	Wave Rider Surfboards	GLOBE EXPRESS	MSCZ	MSCZ9876541	1234567890	2,000.00 KG
MEDUUH820571	ABC Trading Co.	Coastal Crafts Art Supplies	MSC OCEANIA	MSCZ	MSCZ9876511	1234567890	3,000.00 KG
MEDUUH820571	ABC Trading Co.	Tidebreaker T-shirts	MSC OCEANIA	COSU	COSU9876512	1234567890	1,000.00 KG

Figure 7 – Search results for 15 shipping documents

```
Documents: 15
Average document size: 18KB
Imperative variables: 8
Attempt #1. OpenAI API requests Time elapsed: 00:00:02.9789661
Attempt #1. All program Time elapsed: 00:00:03.7524120


Attempt #2. OpenAI API requests Time elapsed: 00:00:04.3081889
Attempt #2. All program Time elapsed: 00:00:04.6543972


Attempt #3. OpenAI API requests Time elapsed: 00:00:02.2930125
Attempt #3. All program Time elapsed: 00:00:02.7082040


Attempt #4. OpenAI API requests Time elapsed: 00:00:03.1995830
Attempt #4. All program Time elapsed: 00:00:03.4447443


Attempt #5. OpenAI API requests Time elapsed: 00:00:04.2532939
Attempt #5. All program Time elapsed: 00:00:04.4986658
```

Figure 8 – Execution time measurement

6 DISCUSSION

The method of imperative variables generally shows applicable results. Compared to research that was performed and resulted in [7–11] the developed method has an easier implementation, so the API that can be built on this model, will be developed without any extra spending time for developing custom NLP models, as it was done and outlined in those studies. Therefore, custom-trained NLP models are often trained on a set of data that is related to a specific area. This approach has advantages in that this model better interacts and produces more accurate results since it is trained according to the specific field. However, the disadvantage could be the absence of enough flexibility, so these models will need to be additionally trained in case when they are used in another field.

The significant advantage of the method is that it uses a modern and well-trained GPT model and an open-source API for processing queries. According to experiment results this method is relatively fast and produces accurate search results. Also, this method is flexible and has relatively small-time expenses.

However, since it is based on OpenAI GPT models which have access to facilities based on a chosen subscription, API has several limitations that lead to the inability to parse large sets of data. Due to them, there is a need to perform several optimizations. For example, if the model deals with a large document, it can be divided into smaller parts which are acceptable by OpenAI API.

This method can be extended by combining other methods that are used to narrow the search range, like ZOCR. With these optimizations, OpenAI API will not be overloaded, but it will not be suitable for all unstructured documents since specific cases can take place when the document contains the desired textual information in different places. Overall, the combination of advanced language models with OCR technologies represents a promising direction for improving document processing and information retrieval tasks. By leveraging the strengths of both approaches, it's possible to create more robust and versatile solutions capable of handling a wide range of document formats and sources.

DOI 10.15588/1607-3274-2024-2-12

Despite these limitations, the method remains a powerful tool for natural language processing tasks, offering a balance between performance and accessibility. Continued improvements in the underlying GPT models and enhancements in the API capabilities may further mitigate these limitations in the future.

The method's integration with existing systems and workflows is relatively straightforward, because of its API-oriented model. This allows developers to seamlessly incorporate its capabilities into their applications without significant overhead.

CONCLUSIONS

The generic rule-based unstructured data search method was developed and incorporated into an API-oriented model.

The scientific novelty of the obtained results is that the generic imperative variables method based on OpenAI GPT models is firstly proposed. It outlines the approach of a rule-based search in unstructured documents based on GPT models. The flexible field-independent search model is designed based on the method. This allows to automate finding the information in documents with no predefined structure, build requests, and criteria in different forms for search and form the desired output results.

The practical significance of the obtained results is that the method is able to automate information retrieval tasks, without the need for predefined structures or explicit rules, making it highly adaptable to diverse use cases. The developed flexible model enables organizations to streamline their document processing workflows, improve efficiency, and extract valuable insights from unstructured textual data. Implemented Web API allows users to build custom templates to perform a rule-based search.

Prospects for further research are to study the optimization approaches to minimize the impact of OpenAI API limitations and decrease the execution time. Additionally, further investigation could involve expanding the usage of imperative variables in programming and software development.

REFERENCES

1. Dutta H., Gupta A. PNRank: Unsupervised ranking of person name entities from noisy OCR text, *Decision support systems*, 2021, P. 113662.
2. Kumar V., Chinmay B., Varsha N. A framework for document plagiarism detection using Rabin Karp method, *International Journal of Innovative Research in Technology and Managemen*, 2021, Vol. 5, pp. 18–19.
3. Onyenwe I. et al. Developing Smart Web-Search using Regex, *International Journal on Natural Language Computing*, 2022,Vol. 11, No. 3, pp. 25–30.
4. OCR – optical character recognition – azure AI services [Electronic resource], *Microsoft Learn: Build skills that open doors in your career*. Mode of access: https://learn.microsoft.com/en-us/azure/ai-services/computer-vision/overview-ocr (date of access: 23.03.2024). Title from screen.
5. Drobac S., Lindén K. Optical character recognition with neural networks and post-correction with finite state methods, *International journal on document analysis and recognition (IJDAR)*, 2020, Vol. 23, No. 4, pp. 279–295.
6. Deshmukh M., Maheshwari S. Free form document based extraction using ML, *International journal of science and research (IJSR)*, 2019, Vol. 8, P. 1.
7. Kwabena A. E. et al. An automated method for developing search strategies for systematic review using natural language processing (NLP), *MethodsX*, 2022, P. 101935.
8. Just J. Natural language processing for innovation search – Reviewing an emerging non-human innovation intermediary, *Technovation*, 2024, Vol. 129, P. 102883.
9. Allen K. S. et al. Natural language processing-driven state machines to extract social factors from unstructured clinical documentation, *JAMIA open*, 2023, Vol. 6, No. 2.
10. Li I. et al. Neural natural language processing for unstructured data in electronic health records: A review, *Computer science review*, 2022, Vol. 46, P. 100511.
11. Qiu Q. et al. Automatic spatiotemporal and semantic information extraction from unstructured geoscience reports using text mining techniques, *Earth science informatics*, 2020, Vol. 13, No. 4, pp. 1393–1410.
12. Research [Electronic resource], *OpenAI*. Mode of access: https://openai.com/research/overview (date of access: 24.03.2024). Title from screen.
13. Koubaa A. et al. Exploring ChatGPT capabilities and limitations: A critical review of the NLP game changer. Riyadh. Preprints, 2023, 29 p. (Preprint / Prince Sultan University; 2023030438).
14. Ekin S. Prompt Engineering For ChatGPT: A Quick Guide To Techniques, Tips, And Best Practices. Texas City: TechRxiv, 2023, 12 p. (Preprint / Texas A&M University; 22683919).
15. Chat completions API [Electronic resource]. Mode of access: https://platform.openai.com/docs/guides/text-generation/chat-completions-api (date of access: 26.03.2024). – Title from screen.
16. Lee M. A mathematical investigation of hallucination and creativity in GPT models, *Mathematics*, 2023, Vol. 11, No. 10, P. 2320.
17. Kingma D. P., Ba J. Adam: A Method for Stochastic Optimization, *3rd International Conference for Learning Representations*, San Diego, 7–9 May 2015.
18. Usage tiers [Electronic resource]. Mode of access: https://platform.openai.com/docs/guides/rate-limits/usage-tiers?context=tier-one (date of access: 26.03.2024). Title from screen.
19. GPT-3.5 Turbo [Electronic resource]. Mode of access: https://platform.openai.com/docs/models/gpt-3-5-turbo (date of access: 26.03.2024). – Title from screen.

Received 04.03.2024.
Accepted 24.04.2024.

УДК 004.93, 004.8

МЕТОД ІМПЕРАТИВНИХ ЗМІННИХ ДЛЯ АВТОМАТИЗАЦІЇ ПОШУКУ ТЕКСТОВОЇ ІНФОРМАЦІЇ У НЕСТРУКТУРОВАНИХ ДОКУМЕНТАХ

Бойко В. О. – асистент кафедри інженерії програмного забезпечення Хмельницького національного університету, Хмельницький, Україна.

DOI 10.15588/1607-3274-2024-2-12

АНОТАЦІЯ

Актуальність. На сьогодні існує багато підходів для виконання ефективного текстового пошуку. Для отримання знаходження та вилучення фрагментів інформації з документів широко використовуються такі методи, як зіставлення з шаблоном і оптичне розпізнавання символів. Однак вони працюють із чітко визначеною структурою документа, тоді як неструктуровані документи не можуть бути оброблені такими методами. А тому проблема полягає в автоматизації текстового пошуку в документах з неструктурованим вмістом. Метою дослідження було розробити метод та реалізувати ефективну модель пошуку вмісту в неструктурованій текстовій інформації.

Мета роботи – реалізація методу та моделі текстового пошуку на основі правил для отримання інформації з документів з неструктурованим текстовим вмістом.

Метод. Для досягнення мети дослідження розроблено та застосовано у відповідній моделі метод критеріального текстового пошуку для знаходження інформації у різнорідному текстовому вмісті. Він заснований на обробці природної мови, яка була вдосконалена в останні роки разом із новим генеративним штучним інтелектом, який стає все більш доступним та продуктивним.

Результати. Метод реалізовано в розробленій моделі, яка представляє шаблон або структуру неструктурованого текстового пошуку для розробників програмного забезпечення. Розроблено прикладний програмний інтерфейс для взаємодії з моделлю.

Висновки. Проведені експерименти у вигляді реалізованого програмного забезпечення підтвердили працездатність запропонованого методу та доводять практичність його використання для вирішення задач текстового пошуку в неструктурованих документах. Перспективи подальших досліджень можуть включати покращення продуктивності за допомогою багатопотоковості або паралелізації для великих текстових документів, а також розробка підходів до оптимізації методу для мінімізації впливу обмежень обробки контенту прикладного програмного інтерфейсу OpenAI. Крім того, додаткові дослідження можуть включати розширення області використання імперативних змінних у програмуванні та розробці програмного забезпечення.

КЛЮЧОВІ СЛОВА: текстовий пошук, неструктуровані текстові документи, обробка природної мови, пошук на основі правил, генеративний штучний інтелект, імперативні змінні.

ЛІТЕРАТУРА

1. Dutta H. PNRank: Unsupervised ranking of person name entities from noisy OCR text / Haimonti Dutta, Aayushee Gupta // Decision support systems. – 2021. – P. 113662.
2. Kumar V. A framework for document plagiarism detection using Rabin Karp method / Vivek Kumar, Bhatt Chinmay, Namdeo Varsha // International Journal of Innovative Research in Technology and Managemen. – 2021. – Vol. 5. – P. 18–19.
3. Developing Smart Web-Search using Regex / Ikechukwu Onyenwe et al. // International Journal on Natural Language Computing. – 2022. – Vol. 11, No. 3. – P. 25–30.
4. OCR – optical character recognition – azure AI services [Electronic resource] // Microsoft Learn: Build skills that open doors in your career. – Mode of access: https://learn.microsoft.com/en-us/azure/ai-services/computer-vision/overview-ocr (date of access: 23.03.2024). – Title from screen.
5. Drobac S. Optical character recognition with neural networks and post-correction with finite state methods / Senka Drobac, Krister Lindén // International journal on document analysis and recognition (IJDAR). – 2020. – Vol. 23, No. 4. – P. 279–295.
6. Deshmukh M. Free form document based extraction using ML / Mona Deshmukh, Shruti Maheshwari // International journal of science and research (IJSR). – 2019. – Vol. 8. – P. 1.
7. An automated method for developing search strategies for systematic review using natural language processing (NLP) / Antwi Effah Kwabena [et al.] // MethodsX. – 2022. – P. 101935.
8. Just J. Natural language processing for innovation search – Reviewing an emerging non-human innovation intermediary / Julian Just // Technovation. – 2024. – Vol. 129. – P. 102883.
9. Natural language processing-driven state machines to extract social factors from unstructured clinical documentation / Katie S. Allen et al. // JAMIA open. – 2023. – Vol. 6, No. 2.
10. Neural natural language processing for unstructured data in electronic health records: A review / Irene Li et al. // Computer science review. – 2022. – Vol. 46. – P. 100511.
11. Automatic spatiotemporal and semantic information extraction from unstructured geoscience reports using text mining techniques / Qinjun Qiu et al. // Earth science informatics. – 2020. – Vol. 13, No. 4. – P. 1393–1410.
12. Research [Electronic resource] // OpenAI. – Mode of access: https://openai.com/research/overview (date of access: 24.03.2024). – Title from screen.
13. Exploring ChatGPT capabilities and limitations: A critical review of the NLP game changer / Anis Koubaa et al. – Riyadh : Preprints, 2023. – 29 p. – (Preprint / Prince Sultan University; 2023030438).
14. Ekin S. Prompt Engineering For ChatGPT: A Quick Guide To Techniques, Tips, And Best Practices / Sabit Ekin. – Texas City: TechRxiv, 2023. – 12 p. – (Preprint / Texas A&M University; 22683919).
15. Chat completions API [Electronic resource]. – Mode of access: https://platform.openai.com/docs/guides/text-generation/chat-completions-api (date of access: 26.03.2024). – Title from screen.
16. Lee M. A mathematical investigation of hallucination and creativity in GPT models / Minhyeok Lee // Mathematics. – 2023. – Vol. 11, no. 10. – P. 2320.
17. Kingma D. P. Adam: A Method for Stochastic Optimization / Diederik P. Kingma, Jimmy Ba // 3rd International Conference for Learning Representations: International Conference, San Diego, 7–9 May 2015.
18. Usage tiers [Electronic resource]. – Mode of access: https://platform.openai.com/docs/guides/rate-limits/usage-tiers?context=tier-one (date of access: 26.03.2024). – Title from screen.
19. GPT-3.5 Turbo [Electronic resource]. – Mode of access: https://platform.openai.com/docs/models/gpt-3-5-turbo (date of access: 26.03.2024). – Title from screen.

DOI 10.15588/1607-3274-2024-2-12

УДК 004.9

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ РОЗПІЗНАВАННЯ ПРОПАГАНДИ, ФЕЙКІВ ТА ДЕЗІНФОРМАЦІЇ У ТЕКСТОВОМУ КОНТЕНТІ НА ОСНОВІ МЕТОДІВ NLP ТА МАШИННОГО НАВЧАННЯ

Висоцька В. А. – д-р техн. наук, доцент, доцент кафедри «Інформаційні системи та мережі», Національний університет «Львівська політехніка», Львів, Україна.

АНОТАЦІЯ

Актуальність. Дослідження спрямоване на застосування штучного інтелекту для розроблення та вдосконалення засобів кіберборотьби, зокрема для боротьби з дезінформацією, фейками та пропагандою в Інтернет-просторі, виявлення джерел дезінформації та неавтентичної поведінки (боти) скоординованих груп. Реалізація проекту сприятиме вирішенню важливого та актуального у наш час питання інформаційної маніпуляції у медіа, адже для ефективної боротьби із викривленням та дезінформацією необхідно отримати ефективний інструмент розпізнання цих явищ у текстових даних для вироблення подальшої стратегії запобігання розповсюдження таких даних.

Метою дослідження є розробка інформаційної технології для автоматичного розпізнавання політичної пропаганди у текстових даних, яка побудована на основі машинного навчання з учителем та реалізована за допомогою методів опрацювання природної мови.

Метод. Розпізнання наявності пропаганди відбуватиметься на двох рівнях: на загальному рівні, тобто рівні документу, та на рівні окремих речень. Для реалізації проекту використано такі методи конструювання ознак, як статистичний показник TF-IDF, модель векторизації «Торба слів», розмічування частин мови, моделі word2vec для отримання векторних представлень слів, а також розпізнавання тригерних слів (підсилюючі слова, абсолютні займенники та «блискучі» слова). У якості основного алгоритму моделювання використана логістична регресія.

Результати. Розроблено моделі машинного навчання для розпізнавання пропаганди, фейків та дезінформації на рівні документу (статті) та на рівні речень. Обидві оцінки моделі є задовільними, проте модель для розпізнання пропаганди на рівні документу впоралася в майже 1,2 разів краще (на 20%).

Висновки. Створені моделі показує відмінні результати розпізнавання пропаганди, фейків та дезінформації у текстовому контенті на основі методів NLP та машинного навчання. Аналіз вихідних даних показав, що моделі розпізнання пропаганди на рівні документу (статті) вдалося коректно класифікувати 6097 не пропагандистських статей та 694 статті пропагандистського характеру. 123 пропагандистські статті та 285 не пропагандистських статей були класифіковані невірно. Отримана оцінка моделі: 0,9433254618697041. Модель розпізнання пропаганди на рівні речень успішно класифікувала 1917 не пропагандистських статей та 205 пропагандистських статей, проте 585 пропагандистських статей та 146 не пропагандистських статей були класифіковані невірно. Оцінка моделі становить: 0,7437784787942516.

КЛЮЧОВІ СЛОВА: дезінформація, фейк, пропаганда, лінгвістичний аналіз, опрацювання природньої мови, машинне навчання, кіберборотьба, штучний інтелект, семантичний аналіз, інформаційна безпека.

АБРЕВІАТУРА

ЗМІ – засоби масової інформації;
ІПСО – інформаційно-психологічна операція;
ІС – інтелектуальна система;
ІТ – інформаційна технологія;
ПЗ – програмне забезпечення;
ПО – предметна область;
СД – сховище даних;
IDF – Inverse Document Frequency;
ML – machine learning;
NLP – Natural Language Processing;
nPMI – Normalized pointwise mutual information;
SVM – Support Vector Machine;
TF – Term Frequency.

НОМЕНКЛАТУРА

S – система розпізнавання пропаганди;
I – множина вхідних даних;
O – множина вихідних даних;
R – основні правила опрацювання вхідних даних;
U – параметри опрацювання вхідних даних;
L_R – метод машинного навчання;
α – оператор скачування вхідних даних;
β – оператор опрацювання вхідних даних;
γ – оператор аналізу статей на основі ML;
μ – оператор ідентифікації тематичних статей;
χ – оператор формування датасету статей;
ω – оператор маркування статті;
λ – оператор прийняття рішення;
i_1 – множина даних із Інтернет-джерел;
i_2 – сховище даних публікацій;
i_3 – словники слів-маркерів пропаганди;
i_4 – множина тематичних ключових слів фейків;
o_1 – періодичні запити на збір публікацій;
o_2 – результат застосування NLP;
o_3 – результат застосування ML;
r_1 – правила збору даних з Інтернет-джерел;
r_2 – правила NLP текстового контенту;
r_3 – правила ML для розпізнавання пропаганди;
r_4 – правила маркування статті як пропаганди;
u_1 – множина умов збору статей в Інтернет-джерелах;
u_2 – множина вимог фільтрування датасету від шуму;
u_3 – множина умов опрацювання датасету статей;
u_4 – множина умов ML для розпізнавання фейку;
u_5 – множина вимог формування висновків.

DOI 10.15588/1607-3274-2024-2-13

ВСТУП

Дезінформація визначається як «фактично невірна інформація, яка не підтверджена доказами». Дезінформація в Інтернет є актуальною та життєво важливою проблемою, особливо в сферах, пов'язаних з війною в Україні. Така інформація, отримана з соціальних медіа, включаючи тематичні онлайн-спільноти, впливає на результати формування громадської думки, керування настроями суспільства та, відповідно, на хід війни в цілому. Занепокоєння з приводу дезінформації зросло із збільшенням кількості запитів на відповідну інформацію в Інтернет, зокрема, в ЗМІ та соціальних мережах. Відсутність захисних механізмів під час обговорень в онлайн-спільнотах сприяє поширенню та зміцненню дезінформації, фейків та пропаганди. Існуюча література здебільшого зосереджена на виявленні фальшивих оглядів і фейкових новин. Однак у літературі бракує комплексної теоретичної основи, розробленої для виявлення дезінформації, особливо в контексті онлайн-спільноти. Враховуючи величезний обсяг дезінформації про війну в Україні, що поширюється в відповідних онлайн-спільнотах, існує необхідність розробити ефективну модель автоматичного виявлення потоку дезінформації для подальшої ідентифікації неавтентичної поведінки скоординованих груп людей/ботів-розповсюджувачів.

Метою дослідження є розроблення інформаційної технології виявлення дезінформації для підвищення рівня інформаційної безпеки держави шляхом розроблення математичних моделей, методів та засобів кіберборотьби з дезінформацією. Зокрема, це сприятиме для автоматичного виявлення джерел дезінформації та неавтентичної поведінки (боти) скоординованих груп в Інтернет на основі стилістичного аналізу та лінгвістичного опрацювання тексту фейків та пропаганди, особливостей їх розповсюдження та репостів на основі ML-методів.

Розробка методів та засобів моніторингу та виявлення дезінформації в Інтернет вимагає розв'язку відповідних задач, зокрема:

– лінгвістичне опрацювання дезінформації для виявлення спільних характерних ознак пропаганди;

– розпізнання пропаганди на рівні статті;

– розпізнання пропаганди на рівні речення;

– тренування моделей для формування прогнозів на основі тестової вибірки;

– розроблення модулів ІС для аналізу текстових потоків контенту для виявлення пропаганди;

– експериментальна апробація розробленої ІТ розпізнавання пропаганди, фейків та дезінформації у текстовому контенті на основі методів NLP та ML.

Наукова новизна полягає у розробленні методів:

– стилістичного аналізу та лінгвістичного опрацювання дезінформації для виявлення спільних характерних ознак фейків одного авторського колективу основі методів опрацювання природньої мови та штучного інтелекту, лінгвістичного аналізу повідомлень, класифікації/кластеризації тексту тощо для виявлення лінгвістичних ознак деструктивного та маніпулятивного спроб впливу на читача;

– виявлення потенційно подібних за стилістикою дезінформації для формування множини потенційних авторів та учасників розповсюдження пропаганди на основі збору/моніторингу/виявлення/класифікації інформаційних загроз в Інтернет-просторі.

Практична новизна полягає у розробленні ІС виявлення пропаганди, а також експериментальна апробація, збір/опрацювання/аналіз отриманих результатів для розрахунку точності/ефективності функціонування на основі реалізації модулів ПЗ як:

– модуль інтелектуального пошуку, збору, маркування, лінгвістичного аналізу та класифікації інформаційних повідомлень для подальшого формування множини потенційних фейків, а також моніторингу, керування, виявлення та відстеження даних інформаційних загроз на основі ML;

– модуль стилістичного аналізу множини фейків для ідентифікації подібних за стилем для одного авторського колективу з подальшим їх класифікацією (людина/бот) на основі методів ML та NLP.

Проект спрямований на застосування штучного інтелекту для розроблення та вдосконалення засобів кіберборотьби, зокрема для боротьби з дезінформацією в Інтернет, а саме для автоматичного виявлення джерел дезінформації та неавтентичної поведінки (боти) скоординованих груп. Необхідно дослідити явище політичної пропаганди у новинних медіа, розпізнати наявність пропаганди у текстових даних. Необхідно також розробити алгоритм підготовки та виокремлення ознак текстових даних, а також побудувати модель машинного навчання, котра розпізнаватиме наявність політичної пропаганди у текстах за допомогою цих ознак. Об'єкт дослідження процеси пошуку, виявлення та класифікації політичної пропаганди, фейків та дезінформації у медіа, зокрема у ЗМІ в Інтернет-середовищі. Предмет дослідження – це методи та засоби розпізнання пропаганди, фейків та дезінформації у текстових даних. Дослідження сконцентроване на розробці системи розпізнання пропаганди, фейків та дезінформації на основі машинного навчання через опрацювання природної мови як на рівні речення, так і на рівні документу.

1 ПОСТАНОВКА ПРОБЛЕМИ

Зростання темпів розповсюдження дезінформації в ЗМІ, зокрема в Інтернет, під час інформаційнішої війни вже давно викликає занепокоєння суспільства, оскільки поширення такої дезінформації має негативний вплив на населення як споживача цього контенту та відповідно хід самої війни. Зазвичай виявлення тематичної онлайн-дезінформації ПО ґрунтується на лінгвістичних особливостях змісту текстового контенту публікацій (статей). Але вони множаться та розповсюджуються швидше, ніж їх ідентифікують та блокують. Тому виявлення джерел подібного контенту, потенційних авторів, механізмів

DOI 10.15588/1607-3274-2024-2-13

розповсюдження, зокрема аналіз та ідентифікація поведінки потенційних генераторів фейків є задачею першочерговою для вдосконалення засобів кіберборотьби з дезінформацією на просторах Інтернету. А це базується на результатах точного та оперативного виявлення стилістично подібного тексту в публікаціях пропаганди та фейків ПО.

Систему розпізнавання пропаганди, фейків та дезінформації у текстовому контенті на основі методів NLP та машинного навчання подамо як:

$$S = < I, O, R, U, L_R, \alpha, \beta, \gamma >, \quad (1)$$

де $I = \{i_1, i_2, i_3, i_4\}$, $O = \{o_1, o_2, o_3\}$, $R = \{r_1, r_2, r_3, r_4\}$, $U = \{u_1, u_2, u_3, u_2, u_3\}$.

Основними процесами моделі аналізу текстового контенту статей із Інтернет-джерел для розпізнавання пропаганди, фейків та дезінформації є «Збір статей для формування датасету», «NLP текстового контенту статей для виділення лінгвістичних ознак», «Машине навчання для розпізнавання пропаганди» та «Формування висновків наявності пропаганди».

Процес «Збір статей для формування датасету» опишемо суперпозицією:

$$C_{AU} = \mu \circ \beta \circ \alpha, \quad (2)$$

$$C_{AU} = \mu(\beta(\alpha(i_1, i_2, i_4), r_1, u_1), u_2). \quad (3)$$

Особливості онлайн-дезінформації можна класифікувати на два рівні: центральний(включаючи особливості теми) і периферійний (включаючи лінгвістичні особливості, особливості настроїв і особливості поведінки користувачів). Необхідно знайти особливості поведінки, щоб відобразити характеристики взаємодії користувачів: початок обговорення, залучення до взаємодії, сфера впливу, посередництво у відносинах та інформаційна незалежність. Тому процес «NLP текстового контенту статей для виділення лінгвістичних ознак» опишемо суперпозицією:

$$C_{CU} = \chi \circ \beta \circ \alpha, \quad (4)$$

$$C_{CU} = \chi(\beta(\alpha(C_{AU}, i_2, i_3, i_4), r_1, u_3), r_2). \quad (5)$$

Щоб побудувати моделі та методи ідентифікації дезінформації в Інтернеті, багато дослідників присвятили себе виявленню особливостей дезінформації. Дезінформацію в соціальних мережах можна розглядати як повідомлення, які публікуються, щоб переконати інших користувачів. Щоб виявити ефективні функції виявлення дезінформації в онлайн-спільнотах, необхідно використали модель, яка зможе допомогти зрозуміти, як дезінформація в Інтернет, зокрема в соціальних мережах та онлайн-спільнотах переконує користувачів. Користувачі зазвичай будують ставлення до повідомлення як центральним, так і периферійним маршрутом. У центральному маршруті користувачі ретельно перевіряють якість і силу інформації; тоді як у периферійному маршруті користувачі більше дбають про поверхневі фактори, такі як репутація джерела, візуальна привабливість і презентація. Окрім змісту повідомлення, деяка вторинна інформація (наприклад, кількість лайків і зірочок) суттєво підвищує валідність і надійність повідомлень. Тому функції центрального рівня повідомлень переконують користувачів на основі вмісту повідомлень, тоді як функції периферійного рівня переконують користувачів через вплив авторів повідомлень. Найкращими функціями для виявлення дезінформації в соціальних мережах можуть бути ті, які розглядають особливості користувача, повідомлення, теми та поведінки користувача.

Процес «Машине навчання для розпізнавання пропаганди» опишемо як:

$$C_{UL} = \omega \circ \gamma \circ \beta \circ \alpha, \quad (6)$$

$$C_{UL} = \omega(\gamma(\beta(\alpha(C_{CU}, L_R, i_2), i_3), u_4), r_3). \quad (7)$$

Створення моделі виявлення дезінформації, яка об’єднує функції центрального рівня (зокрема особливості теми) та функції периферійного рівня (зокрема лінгвістичні особливості, особливості настрою та особливості поведінки користувачів), потребує подальших досліджень. На основі цих функцій необхідно оцінювати їхню здатність автоматично відрізняти дезінформацію від правдивої в межах тематичної онлайн-спільноти за допомогою різних методів машинного навчання.

Процес «Формування висновків наявності пропаганди» опишемо як:

$$C_{US} = \lambda \circ \gamma \circ \beta \circ \alpha, \quad (8)$$

$$C_{US} = \lambda(\gamma(\beta(\alpha(C_{US}, i_2), i_4), u_5), r_4). \quad (9)$$

Розроблена система швидкої ідентифікації джерел дезінформації має базуватися на аналізі неавтентичної поведінки учасників розповсюдження фейків. Результати не лише мають продемонстрували ефективність поведінкових особливостей у виявленні дезінформації, але й запропонували як методологічний, так і теоретичний внесок у виявлення дезінформації з точки зору інтеграції особливостей повідомлень, а також особливостей авторів повідомлень.

На фоні інформаційної війни витрачається багато ресурсів та часу на оперативний збір контенту, його опрацювання та аналіз, а також генерування рішень/висновків щодо його наповнення. На це також впливає мова публікацій, при перекладі якої суттєво/частково спотворюється зміст. ІС не зможе повністю замінити діяльність фахівців кібербезпеки та кіберборотьби. Але вона буде допоміжним інструментом для оперативного формування відповідних датасетів/корпусів фейкового контенту та їх джерел, стилістичного та лінгвістичного аналізі тексту дезінформації для формування інформаційного портрету авторів, пошук авторів та розповсюджувачів через аналіз неавтентичної поведінки та результатів аналізу стилю написання контенту, а також реагування на динаміку змін або локальні зміни в

DOI 10.15588/1607-3274-2024-2-13

контентому потоці, маркуючи відповідний контент як ймовірносно фейковий.

2 АНАЛІЗ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

Онлайн ЗМІ та соціальні мережі дозволяють швидко обмінюватися інформацією, в тому числі і дезінформацією як цілеспрямовано, так і випадково/хаотично. Поряд з основною перевагою як організація швидкого доступу для всіх бажаючих до оперативної та актуальної інформації, онлайн медіа часто використовують для поширення навмисно оманливого контенту як фейків та пропаганди про конкретні події, людей або організацій, в тому числі уряди [1]. Останнім часом яскравими прикладами розповсюдження дезінформації є спроби російського уряду контролювати інформацію під час війни в Україні з 2014 року, наприклад, авіакатастрофа MH17 [2]. Паралельно на багато онлайн-інформації накладається регіональна цензура на певних територіальних регіонах із-за політичних, економічних, соціальних, релігійних та інших чинників, наприклад, для контролю/управління думкою людей цього регіону, наприклад на окупованих територіях росії для контролю майбутніх виборців бункерного президента [3]. Загубитися та зорієнтуватися в цій масі потоку контенту з протилежними фактами та причинами подій/явищ пересічній людині легко [4]. Контролювати, що показувати/скривати (накладати цензуру) серед Інтернет-контенту пересічному користувачу в демократичних державах є неетично, незаконно та недоцільно без прямих доказів щодо наявності дезінформації/ фейку/ пропаганди для цілеспрямованого порушення інформаційної безпеки організації/країни [5–6]. Це один із перших кроків переходу до тоталітаризму. А надавати інформацію, наприклад, журналістам про можливий тематичний фейк для проведення журналістичного розслідування або попередження пересічного читача про можливість наявності в цьому контенті/ресурсі дезінформації є з одного боку підтримкою свободи слова, з іншого надання можливості людині обирати чому вірити. Це дає змогу отримувати розуміння подій та орієнтування в потоці інформації для вирішення буденних задач і корегування бізнес-стратегій тощо.

Політична пропаганда є спрямованим та навмисним поширенням інформації, метою якого є вплив на громадську думку суспільства на користь певної громадської позиції чи спільної справи. Пропаганда може відбуватися як у формі дезінформації, тобто фабрикування недостовірних та фальшивих новин, так і використовувати більш складні та комплексні методики. Пропаганда, фейки та дезінформація зазвичай генерується, формується та розповсюджується за допомогою ЗМІ, є дотичною до тих чи інших політичних подій – передвиборна кампанія, фінансова криза тощо. Отже, у деяких випадках, для розпізнання пропаганди необхідно знати контекст політичного клімату у світі.

Значне та масове розповсюдження фейків, пропаганди та дезінформації на фоні війни в Україні без систематичного та ґрунтовного аналізу ймовірно впливає на формування думки суспільства та керує нею, а також призводить до панічних настроїв серед відповідного регіону/верства населення, значно впливає на корегування стратегій/планів державних органів, соціальних служб, бізнесу, тощо. Блокуванням дезінформації та джерел її розповсюдження, а також ідентифікацій потенційних авторів на основі аналізу неавтентичної поведінки зазвичай є функціональним обов'язками уповноважених органів, особливо під час інформаційної війни. Але вона настільки зараз швидко та оперативно генерується/розповсюджується на основі застосування сучасних інформаційних технологій та штучного інтелекту, що справитися з цією задачею на 100% ніхто не спроможний без використання нових методів та засобів на базі машинного навчання.

Для повного аналізу всього нового/старого контенту не вистачить ресурсів. Та і поки буде проведений системний аналіз даних, сама дезінформація стане застарілою. А ось швидке формування/модифікування/поповнення баз/сховищ даних маркованого контенту як блокований/ неблокований в певному регіоні, відсортованого за відповідними метриками (час, тема, регіон блокування, мова тощо) від актуального/релевантного до менш актуального для подальшого аналізу методами/технологіями NLP/ML значно пришвидшить процес орієнтування серед хаосу нової інформації в Інтернет. Визначення теми/причини блокування контенту (накладання цензури) на певному регіоні дозволить покращити якість ідентифікації фейків/пропаганди/дезінформації відповідної тематики. Тому актуальним та необхідним є розроблення ІС автоматичного виявлення джерел дезінформації та неавтентичної поведінки користувачів чатів в кібернетичному просторі. ІС має бути реалізована на нових принципах інформаційної безпеки (моніторинг даних, виявлення загроз, прогнозування), що дасть змогу ідентифікувати, моніторити, повідомляти про рівень загрози та прогнозувати кіберзагрози, а також ступінь ймовірнісного інформаційно-психологічного впливу на громадську думку. З огляду на це, такий проект є релевантним, актуальним, перспективним і своєчасним для збільшення ступеня інформаційної безпеки держави на основі ідентифікації, моніторингу, прогнозування та аналізу загроз у кіберпросторі України.

Проблема політичної пропаганди стає усе більш актуальною за рахунок того, що інформація стає усе більш доступною, поширюється практика діджиталізації суспільних медіа, продукувати та редагувати новини стає усе простіше, зростає вплив соціальних мереж. Наприклад, дослідження поширення пропаганди за допомогою соціальних

DOI 10.15588/1607-3274-2024-2-13

мереж, підтверджує, що пропаганда, розповсюджена таким чином, поширюється швидше та на більш широкий демографічний спектр, а також є більш стійкою до розпізнання за рахунок використання підтверджувального упередження [7]. Окрім цього, результати нещодавнього дослідження Science Magazine показали, що плітки та фейкові новини розповсюджуються приблизно у шість разів швидше, ніж достовірна та правдива інформація [8]. Це вказує на те, що явище політичної пропаганди не лише згубно впливає на сучасний політичний клімат, але й частково формує його. На даний момент більшість проектів, пов'язаних із розпізнанням пропаганди, виконуються за допомогою статистичних досліджень із залученням спеціалістів, проте, за рахунок швидкого поширення пропаганди за допомогою онлайн-медіа, є доволі неефективним та дорогим з точки зору використання ресурсів. Саме тому побудова ефективної модель ML для оптимізації цього процесу є як ніколи актуальною, особливо враховуючи той факт, що на сьогоднішній день ML-системи, базовані на NLP, набувають усе більшої популярності як у академічному, так і у прикладному середовищі науки про дані.

Для успішного аналізу та опрацювання природного тексту, необхідно зазначити певні ознаки пропаганди, котрі використовують при класифікації тексту. Для цього необхідно проаналізувати, яку саме структуру текст повинен мати для того, аби бути маркованим як ймовірно пропагандистський. Основні методи поширення пропаганди є наступними [9]:

– Відволікаючий маневр. Презентація недоречного матеріалу, котрий не має відношення до обговорюваного питання у тексті.

– Викривлення позиції. Заміна подібною, але не аналогічною, позицією.

– Whataboutism. Позиція опонента дискредитується шляхом звинувачення його у лицемірстві без безпосередньої аргументації.

– Причинне спрощення. Виділення та презентація лише одної гіпотетичної причини певного явища, коли таких причин є декілька.

– Навмисна розпливчатість, збиття з пантелику. Використання навмисно абстрактних термінів та слів таким чином, щоб інтерпретація сказаного не була єдиною та очевидною.

– Апелювання до авторитету. Ствердження, що заявка є вірною, бо чинний авторитет/експерт її підтримує, зазвичай без подання жодних доказів.

– Чорно-біла оманливість. Презентація двох альтернативних варіантів або точок зору як єдиних можливих, навіть якщо існують інші варіанти.

– Навішування ярликів. Спосіб, коли джерело пропаганди надає явищу, проти котрого виступає, негативних зміст, зазвичай апелюючи до того, чого цільова аудиторія боїться, що ненавидить.

– Навантажена мова. Використання певних фраз та слів з сильним емоційним підтекстом (позитивним або негативним) для впливу на цільову аудиторію.

– Перебільшення або мінімізація. Певне явище репрезентовано або у надмірному вигляді, або ж як щось менш важливе, ніж є насправді.

– Розмахування прапором. Маніпуляція патріотичними поглядами/почуттями цільової аудиторії для виправдання/поширення явища/ідеї.

– Сумнів. Ставлення під сумнів авторитетність та надійність певної людини або явища.

– Апелювання до страху або упередження. Спроба сприяти підтримці певної ідеї шляхом вселення страху та тривоги, або ж апелювання до певних соціальних упереджень цільової аудиторії.

– Слогани. Використання коротких ударних фраз, для навішування ярликів або стереотипізації через апелювання до емоцій цільової аудиторії.

– Кліше без суті. Слова/фрази для перешкоджання аргументованому обговоренню ситуації та критичному мисленню.

– Загальна платформа. Спроба переконати цільову аудиторію приєднатися до справи та прийняті певне рішення як всі інші або більшість.

– Повторення. Повторення одного посилу декілька разів для зомбування цільової аудиторії.

Пропаганда є комплексним комунікативним явищем, котре використовує різноманітні методи та підходи для досягнення своєї мети. Задача розпізнання пропаганди у текстах не є новою та у цій сфері проведено багато досліджень. Для аналізу обрано декілька аналогів, різних за методами дослідження та моделювання.

– Framing and Agenda-setting in Russian News: a Computational Analysis of Intricate Political Strategies. Дослідження проведено колективом вчених із університету Карнегі Мелон, університету Хайфа та Стенфордського університету у 2018 році [10]. Основна мета роботи – дослідити «фреймінг» у новинних джерелах росії протягом економічно несприятливого становища у країні та те, як це пов'язано із висвітленням новин, що стосуються США, у російських ЗМІ (бо більшість із них керуються державними органами та знаходяться під їх безпосереднім впливом). Був встановлений сильний негативний кореляційний зв'язок між економічної ситуацією у росії та кількістю новин, що фокусуються на висвітленні подій у США. Наступним кроком досліджено те, у якому контексті та з яким фреймом ці новини подаються. Фрейм у даному контексті визначає те, яким чином (із якими конотаціями) висвітлюється та чи інша новина у медіа. Потім висунута гіпотеза про те, що кореляційний зв'язок є причинно спрямованим, і що несприятлива економічна ситуація у росії є безпосереднім чинником, котрий викликає посилену увагу до США у державних ЗМІ. За допомогою імовірнісних методів (причинність за Грейнджером та лінійної регресійної моделі) встановлено, що причинний зв'язок між цими двома змінними є наявним у наборі обраних даних. Далі за допомогою імовірнісного методу Баєса сформовано так звані лексикони фрейму, з

DOI 10.15588/1607-3274-2024-2-13

урахуванням конкретного слова та фрейму, до якого воно належить. Для кожного фрейму обрано 250 слів, котрі мають найбільшу імовірність зустрітися у відповідному контексті. У якості початкового, базового рішення, обрано логістичну регресію та модель «торба слів». Для того, аби визначити, які фрейми стосуються США, обчислено nPMI. Виділено п'ять фреймів, котрі надалі використовуються для дослідження. У дослідженні також використовується Structure Topic Modelling для того, аби концептуалізувати статті на тому рівні, на якому б їх сприймав читач. Таким чином, досліджено метод пропаганди, націлений на відволікання публіки від скрутного стану внутрішніх справ у країні за допомогою перекладання уваги на зовнішню політику.

– The Use of Supervised Learning Algorithms in Political Communication and Media Studies: Locating Frames in the Press. Дослідження фокусується на дослідженні фреймінгу у сучасних медіа [11] та побудовано на наборі даних із іспанських новинних джерел за 2015 рік, зокрема увагу сфокусовано на висвітленні кризи біженців. Обрано два фрейми, згідно з якими проводилася класифікація – human rights та security. Основний алгоритм ML, котрий використовується у дослідженні для побудови моделі – метод SVM, який застосовують для розв'язку задач класифікації та побудований на припущенні, що дані є лінійно розподілені таким чином, що можна знайти таку гіперплощину, котра могла б ефективно розділити їх один від одного. Однак, лінійне ядро моделі не показало задовільних результатів, тому в остаточній версії алгоритму використано метод опорних векторів із радіальним ядром. Для видалення семантично незначних слів із корпусу даних використано алгоритм IDF.

– Фейкогриз – це інструмент для розпізнання пропаганди, створений українською платформою дослідження даних Texty.org [12–13]. Система побудована у вигляді додатку для веб-браузера, а також у якості чат-боту для платформи Telegram. У якості вхідних даних Фейкогриз приймає посилання на новину і, використовуючи веб-скрейпінг, визначає, чи у тексті, поданому за джерелом, є наявною маніпуляція або пропаганда. Модель побудована за принципом трансферного навчання, що означає, що використовується модель загального призначення, котра, можливо, не була натренована для виконання специфічного завдання, проте згодом відбувається відповідне до задачі налаштування гіперпараметрів моделі [14–15]. Замість того, щоб фокусуватися на проблемі класифікації та витрачати ресурси на позначення даних відповідними класами, Фейкогриз побудований на алгоритмі кластеризації, котрий автоматично визначає клас даних, котрі отримує.

3 МАТЕРІАЛИ ТА МЕТОДИ

Оскільки розпізнавання пропаганди у текстових даних [16–18] відбуватиметься на двох рівнях – на рівні документу та на рівні речення, обрано два окремі набори даних для кожної задачі.

– Розпізнання пропаганди на рівні документу. Набір даних для розпізнавання пропаганди для цієї задачі складається із 35993 статей (включно із заголовками) англійською мовою, кожна із яких промаркована як «пропаганда» або «не-пропаганда». Також в наборі присутній унікальний ідентифікатор для кожної із статей. Дані подані як текстовий файл, у якому текст статті є відділеним від категорії та ідентифікатору знаками табуляції. Після завантаження файлу, видалення атрибуту ідентифікатору та перетворення даних до формату pandas.DataFrame (рис. 1).

	article	label
0	Et tu, Rhody? A recent editorial in the Provi...	non-propaganda
1	A recent post in The Farmington Mirror — our t...	non-propaganda
2	President Donald Trump, as he often does while...	non-propaganda
3	February is Black History Month, and nothing l...	non-propaganda
4	The snow was so heavy, whipped up by gusting w...	non-propaganda
5	Four months after the Sandy Hook School shooti...	non-propaganda
6	The first major newspaper article about Donald...	non-propaganda
7	For three years, starting in 2008, New York ar...	non-propaganda
8	President Donald Trump's tumultuous administra...	non-propaganda
9	With Hartford on edge about the future of Aetn...	non-propaganda
10	An employee at a Hibachi Express in Florida ha...	non-propaganda
11	With the toll of the carnage from the country'...	non-propaganda
12	The State Department's point-man on North Kore...	non-propaganda
13	The Trump Organization announced Monday that i...	non-propaganda
14	Aer Lingus' service from Bradley International...	non-propaganda
15	For its show "Constellations," which ends its ...	non-propaganda
16	The Corporation for Public Broadcasting (CPB) ...	non-propaganda
17	All five members of New Britain's state legisl...	non-propaganda
18	The leader and second in command of a credit-c...	non-propaganda

Рисунок 1 – Датасет для розпізнання на рівні статті

– Розпізнання пропаганди на рівні речень. Набір даних для цього типу задачі містить у собі близько 450 англомовних статей (включно із заголовками), розбитих на окремі речення. Кожне речення марковане як «пропагандистське» або «не пропагандистське». Також набір даних містить у собі унікальний ідентифікатор для кожної статті, а також унікальний ідентифікатор для речення у межах статті, до якої воно належить. Дані подано як набір текстових файлів, окремий для кожної статті. Атрибути даних також зберігаються окремо. Загалом набір даних налічує 15168 речення (рис. 2). Після завантаження кожної колекції файлів, конкатенуємо їх та формуємо єдиний pandas.DataFrame, усуваючи із набору даних унікальні ідентифікатори статей/речень. Оскільки обидва набори даних є промарковані за двома класами, під час реалізації проекту вирішуватиметься задача бінарної класифікації. Існує багато відомих методів вирішення цієї задачі,

DOI 10.15588/1607-3274-2024-2-13

включно із алгоритмами нейронних мереж, проте заради економії обчислювальних ресурсів зупинимося на класичних методах ML із вчителем. Одними із найбільш відомих та ефективних моделей є SVM, наївний класифікатор Баєса та логістична регресія. Розглянемо кожен із них.

	sentence	label
0	US bloggers banned from entering UK	non-propaganda
2	Two prominent US bloggers have been banned fro...	non-propaganda
4	Pamela Geller and Robert Spencer co-founded an...	propaganda
6	They were due to speak at an English Defence L...	non-propaganda
8	A government spokesman said individuals whose ...	non-propaganda
...	...	...
15164	This is a Moon of Alabama fundraiser week.	non-propaganda
15165	No one pays me to write these blog posts.	non-propaganda
15166	If you appreciated this one, or any of the 7,0...	non-propaganda
15167	Posted by b on November 29, 2018 at 10:23 AM |...	non-propaganda
15168	Comments	non-propaganda

14263 rows × 2 columns

Рисунок 2 – Датасет для розпізнання на рівні речення

– Метод SVM – це модель, призначена для бінарної класифікації, зокрема, для класифікації текстових даних (рис. 3) із використанням неімовірнісного лінійного бінарного класифікатору. Модель SVM є поданням зразків як точок у просторі, де зразки з окремих категорій розділено найбільш оптимальною гіперплощиною. Модель може мати декілька видів ядр, проте найчастіше використовується із лінійним ядром. Для того, аби класична модель SVM із лінійним ядром показувала хороші результати, дані повинні бути лінійно розділеними. Працює ефективно у тому випадку, коли між даними різних класів є чіткий розподіл, дані є багатовимірними та тоді, коли кількість вимірів є більшою за загальну кількість екземплярів даних. Проте не призначена для роботи із наборами даних великого обсягу та не є ефективною у випадку, коли у наборі даних є багато «шуму», а також тоді, коли кількість екземплярів даних одного класу перевищує кількість екземплярів іншого класу, тобто дані повинні бути збалансованими. Як бачимо на рис. 3, оскільки наші набори даних не є збалансованими, лінійна модель SVM погано впоралась із своєю задачею – багато зразків пропагандистських текстів були класифіковані як не пропагандистські.

– Наївний класифікатор Баєса для визначення ймовірності приналежності екземпляру до одного з класів, приймаючи гіпотезу незалежності змінних (рис. 4). Наївний класифікатор Баєса є не надто чутливим до відсутності певних значень атрибутів у наборі, швидше працює у тому випадку, коли розмір тренувальної вибірка є відносно великим. Проте приймає гіпотезу незалежності даних, тому може бути неефективним у тому випадку, якщо вони пов’язані між собою, а також є дуже чутливим до форми вхідних даних. Як бачимо на рис. 4, оскільки текстові дані мають багато шуму, наївний класифікатор Баєса також погано впорався із своєю задачею.

– Логістична регресія побудована на основі лінійної регресії, проте, на відміну від неї, логістична регресія призначена для задачі класифікації (рис. 5).

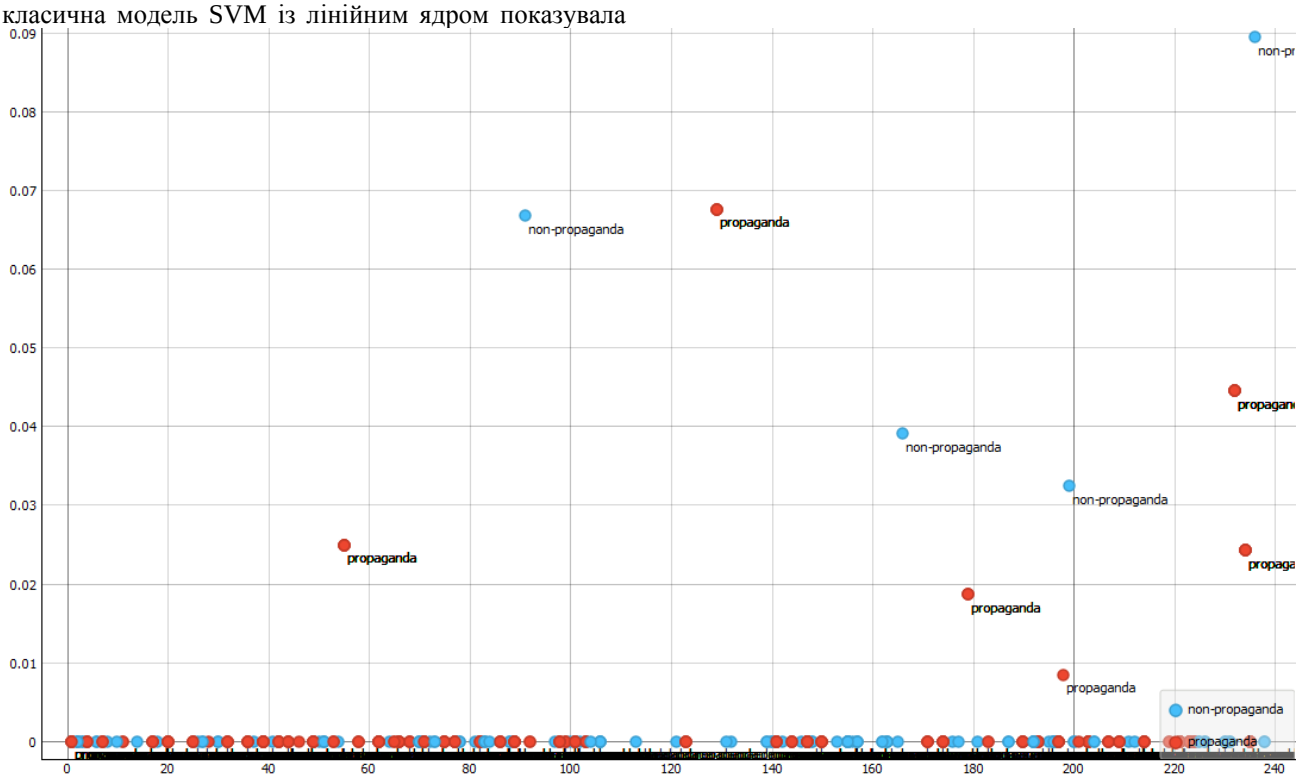


Рисунок 3 – Візуалізація роботи алгоритму SVM

DOI 10.15588/1607-3274-2024-2-13

Прогнозування імовірності належності змінної до того чи іншого класу визначається шляхом порівняння значення із логістичною кривою. Модель логістичної регресії не є чутливою до перенавчання у тому випадку, коли набір даних не є багатовимірним, проте навіть у такому випадку можна використати алгоритм регуляризації.

Зупиняємо вибір на моделі логістичної регресії, оскільки не можемо гарантувати незалежність змінних одна від одної для текстових даних (вимога наївного класифікатору Баєса), а також відсутність шуму (може викликати неефективність у роботі SVM). До того ж, наші дані не збалансовані з точки зору розподілу за класами. Оскільки сирий текст сам по собі не має жодних ознак та атрибутів та не є придатним для використання у моделі ML, необхідно також визначити методи вилучення ознак.

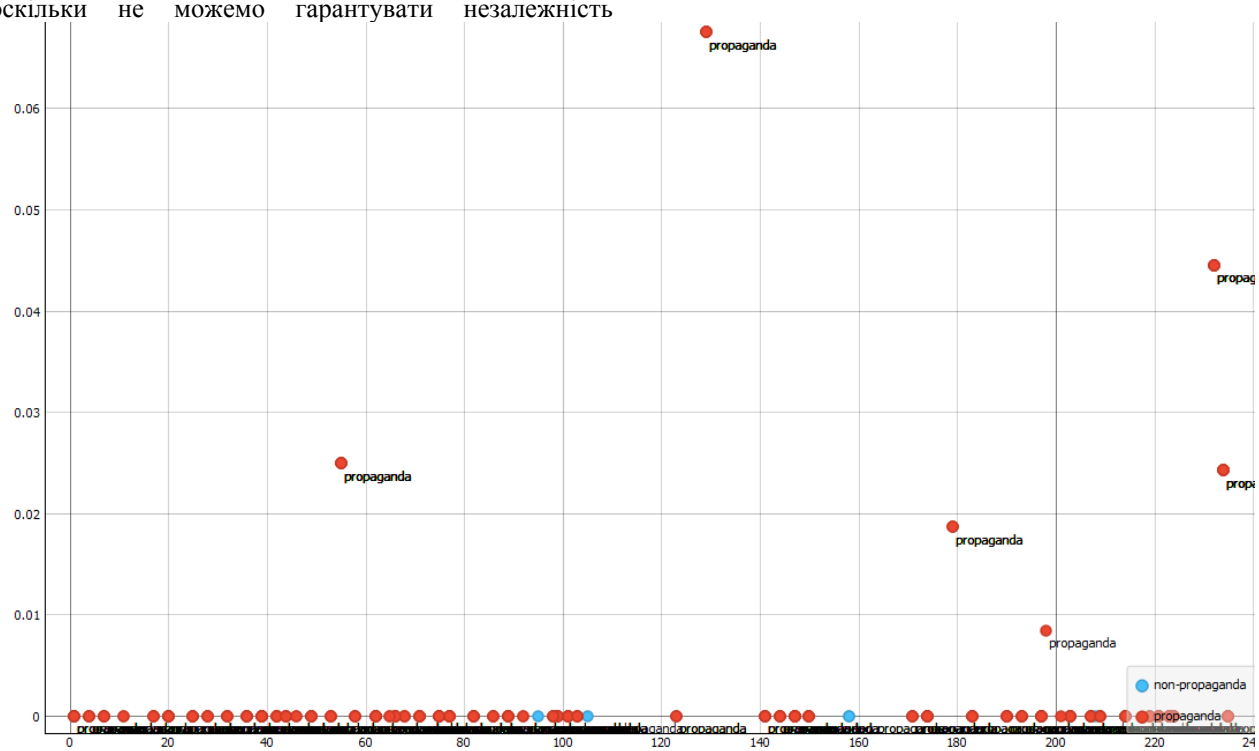


Рисунок 4 – Візуалізація роботи алгоритму наївного Баєса

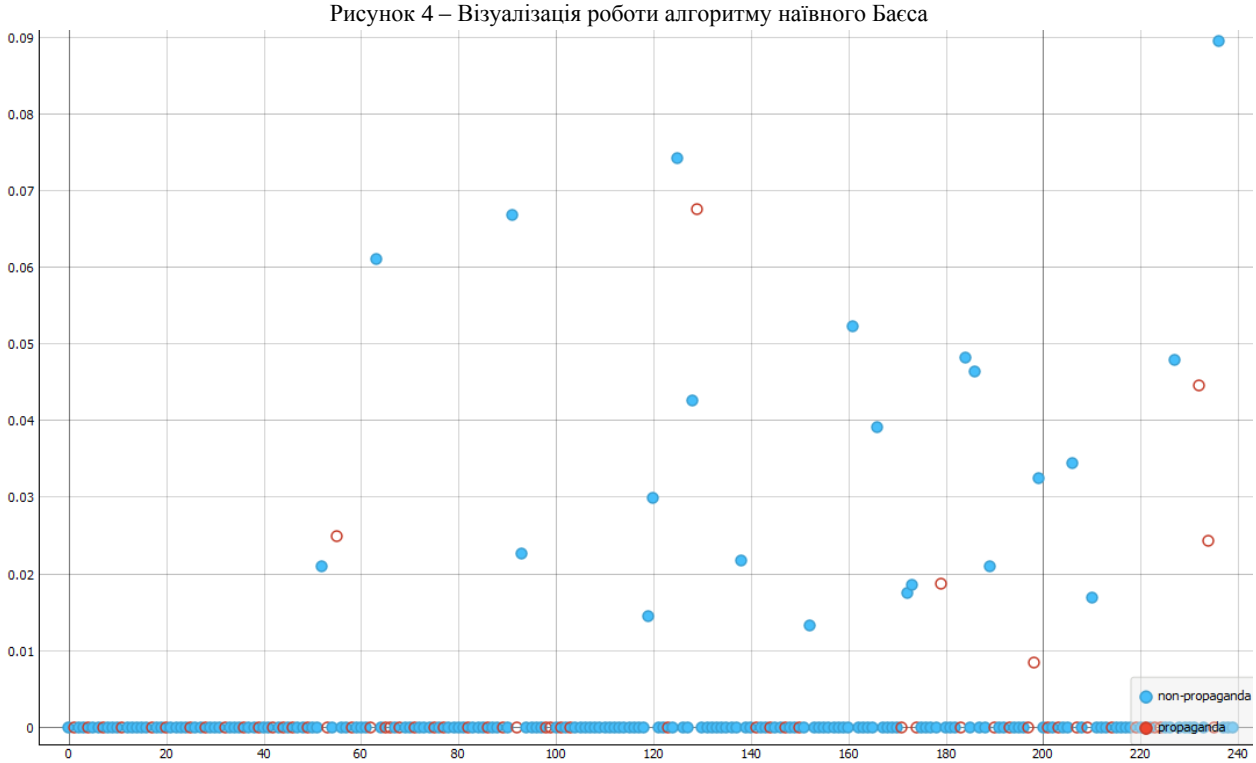


Рисунок 5 – Візуалізація роботи моделі логістичної регресії

DOI 10.15588/1607-3274-2024-2-13

Перелік методів є наступним:

– Векторизація текстових даних за моделлю «Торба слів». Кожному слову (терму) у корпусі присвоюється певне число, а текст перетворюється на вектор розмірністю N, де N – це загальна кількість слів у корпусі, у якому значення кожного елемента дорівнює частоті терму.

– TF-IDF трансформація на основі оцінки важливості слів у контексті статті/речення, що є частиною колекції статей/речень.

– Розмічування частин мови. Форматування текстових даних у вигляді «%слово%_%частина мови, до якої належить слово%_%лема слова%».

– Використання Word2Vec моделі для вбудовування слів. Використання неглибокої двошарової нейронної мережі для векторизації слів із одночасним зменшенням кількості вимірів.

4 ЕКСПЕРИМЕНТИ

Основою методології дослідження пропонуємо синтезовану технологію на основі методів штучного інтелекту, комп'ютерної лінгвістики, машинного навчання, інтелектуального аналізу даних, статистичної обробки даних, теорії систем та системного аналізу, комп'ютерного та імітаційного моделювання тощо. Проблема складається з двох основних складових – визначення множини інформації як фейкової та основі неї знайти джерела та проаналізувати неавтентичну поведінку учасників.

Принцип функціонування ІС автоматичного виявлення джерел дезінформації та неавтентичної поведінки користувачів чатів:

Етап 1. Визначення множини інформації як фейкової:

Крок 1.1. Збір та інтеграція контенту відповідної мови з відповідних ресурсів у СД.

Крок 1.2. Перевірка чи блокований контент з конкретного ресурсу в певному регіоні.

Крок 1.3. Маркування кожного контенту як блокований/неблокований на певному регіоні з відповідним додаткових метриками (час, ресурс, частота появи блокованих/неблокованих дублів, наявність в назві/дайджесті/анотації відповідних маркованих слів, наприклад власні назви, тощо)

Крок 1.4. Формування проміжної бази маркових відсортованих даних.

Крок 1.5. Застосування до топового контенту NLP методів для розрахунку потенційності фейку і/або теми як причини блокування контенту на певному регіоні на основі словників та множини метрик. Схема NLP процесу визначення теми контенту:

1.5.1. Визначення множини ключових слів відповідного контенту та множини наявних слів-маркерів (власних назв, абревіатур, топ-слів відповідної теми тощо). Визначення якщо можливо теми контенту (метод класифікації тексту).

1.5.2. Якщо за ключовими словами складно визначити тему – ідентифікація стійких словосполучень. Визначення якщо можливо тему контенту

1.5.3. Якщо за стійкими ключовими словами складно визначити тему – проведення семантичного аналізу та побудова онтології. Визначення якщо можливо тему контенту.

1.5.4. Якщо за результатами семантичного аналізу це зробити неможливо – відповідно маркувати та передати в список для роботи модератора контенту

1.5.5. При визначеній темі якщо контент маркований як блокований перевірити з списком тем блокованих на цьому регіоні раніше тем. Якщо немає – поновити список. Якщо є поновити кількість блоків цієї теми як цензури в конкретному регіоні.

Крок 1.6. Застосування технологій ML для покращення аналізу/маркування/ NLP даних. Попередньо тренування моделей ML на перевіреному тренувальному датасеті.

Крок 1.7. Формування моделей/шаблонів потенційних фейків для поновлення списку метрик сортування маркованого контенту на кроці 1.3 та метрик/словників для NLP.

Крок 1.8. Постійне поновлення проміжної бази маркових відсортованих даних та переведення в архів застарілого контенту.

Крок 1.9. Поновлення тренувального датасету для вдосконалення моделей ML. Загальна схема процесу навчання та тренування модуля аналізу дезінформації:

Конвеєр 1.9.1. Попередньо марковані дані → NLP → ML → Моделі/шаблони/метрики

Конвеєр 1.9.2. Вхідні нові дані → Маркування даних (блоковані/неблоковані) → NLP → ML → Маркування контенту (фейк/не фейк) або знаходження потенційної причини блокування (не фейк, але саме ця подія/тема є забороною на певному регіоні для пересічної аудиторії). Загальна схема ІС розпізнавання пропаганди подана на рис. 6. Процес розпізнання пропаганди на рівні статті подано на рис. 7, а на рівні речення – на рис. 8.

Етап 2. Ідентифікація джерел та аналіз неавтентичної поведінки учасників

Крок 2.1. Створення моделі виявлення дезінформації, яка об'єднує функції центрального рівня (зокрема особливості теми) та функції периферійного рівня (зокрема лінгвістичні особливості, особливості настрою та особливості поведінки користувачів),

Крок 2.2. Оцінювання здатності функцій центрального та периферійного рівня автоматично відрізняти дезінформацію від правдивої в межах тематичної онлайн-спільноти за допомогою різних методів машинного навчання.

DOI 10.15588/1607-3274-2024-2-13

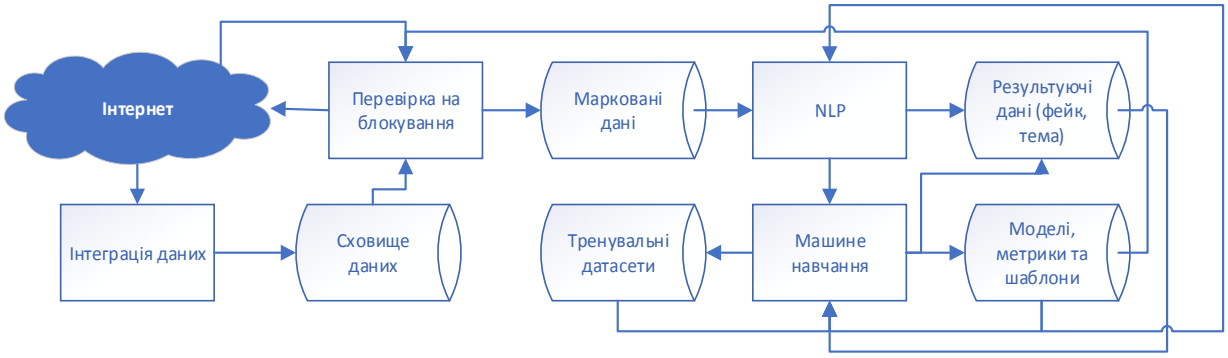


Рисунок 6 – Загальна схема системи розпізнавання пропаганди

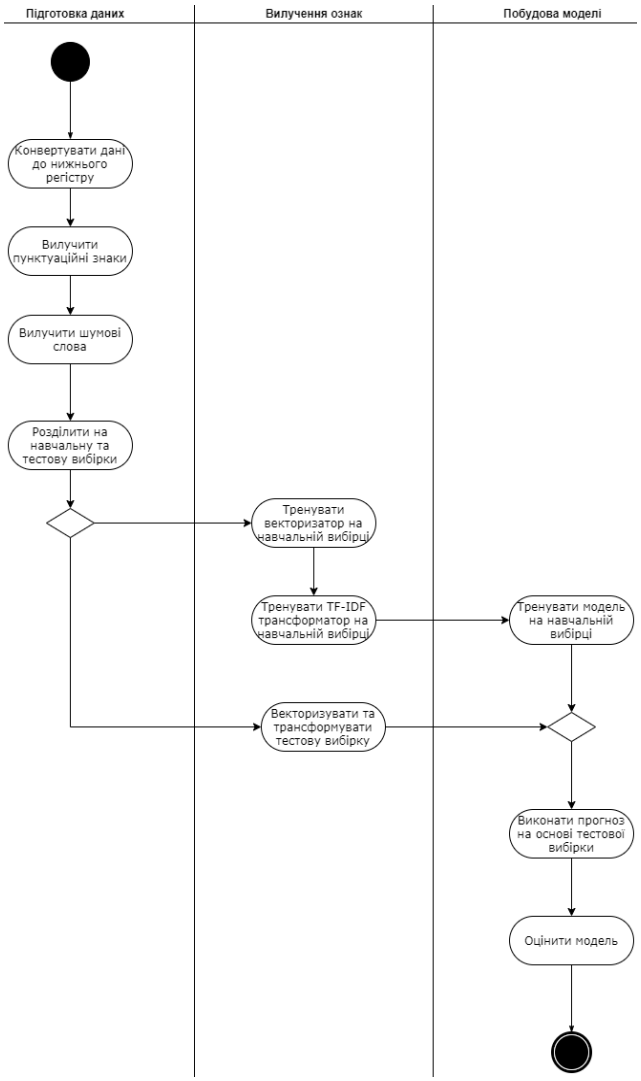


Рисунок 7 – Розпізнання пропаганди на рівні статті

Крок 2.3. Інтелектуальний пошук фейків на основі машинного навчання.

Крок 2.4. Знаходження множини стилістично подібних фейків для одного автора.

Крок 2.5. Знаходження першоджерел фейку на основні аналізу графу розповсюдження.

Крок 2.6. Аналіз поведінки автора/колективу/бота за тривалий проміжок часу для формування множини основних характерних поведінкових рис.

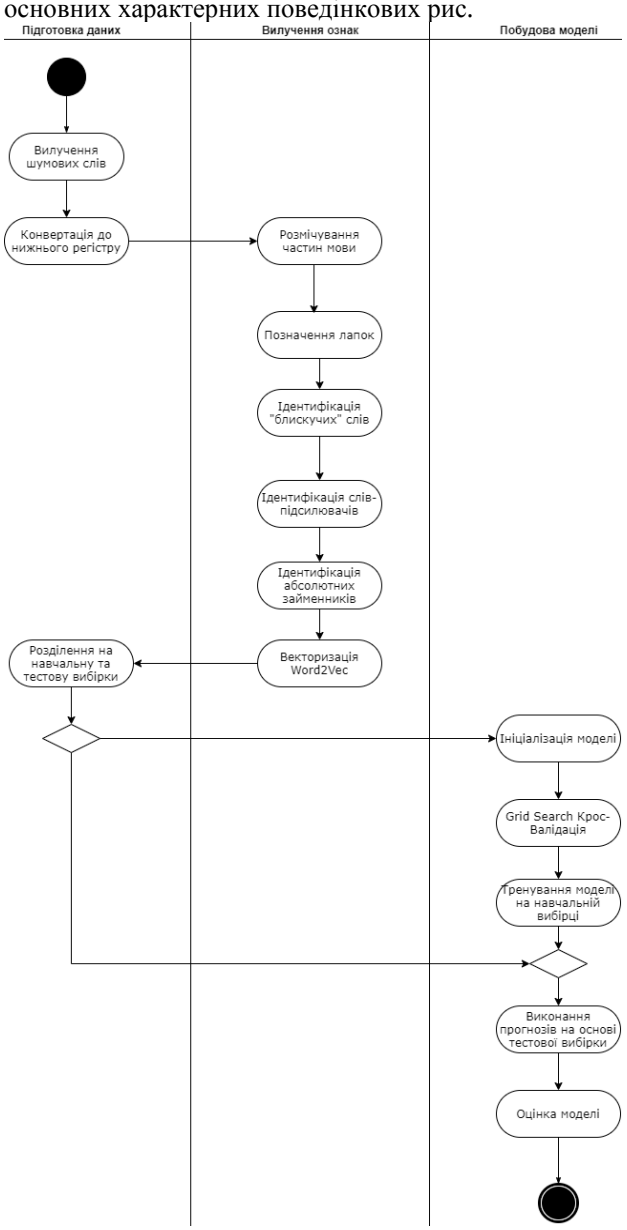


Рисунок 8 – Розпізнання пропаганди на рівні речень

DOI 10.15588/1607-3274-2024-2-13

Крок 2.7. Знаходження інших фейків автора за його стилем написання та поведінки.

Крок 2.8. Формування портрету поведінки автора та моделі передбачення поведінки.

Крок 2.9. На основі аналізу інформаційних портретів різних авторів формувати прогнози розвитку та розповсюдження фейків (частота, густина, тематика), наприклад для ІПСО.

Розпізнання пропаганди на рівні статті (рис. 7):

– Підготовка даних – очищення даних, усунення непотрібних слів та атрибутів (шумові слова та символи пунктуації), конвертування даних до нижнього регістру, розділення даних на навчальну та тестову вибірки;

– Вилучення ознак – виокремлення атрибутів із текстових даних – ініціалізація векторизатора та TF-IDF трансформатора, їх тренування та відповідні перетворення для начальної та тестової вибірок;

– Побудова моделі – ініціалізація моделі, тренування, виконання прогнозів на основі тестової вибірки та оцінка ефективності роботи моделі.

Розпізнання пропаганди на рівні речення (рис. 8):

– Підготовка даних – вилучення шумових слів та конвертування до нижнього регістру (знаки пунктуації знадобляться нам для ідентифікації певних ознак пропаганди), а також розділення даних на навчальну та тестову вибірки;

– Вилучення ознак – розмічення частин мови, позначення лапок, ідентифікація так званих «блискучих» слів, слів-підсилювачів та абсолютних займенників, а також векторизація за допомогою нейронної мережі Word2Vec;

– Побудова моделі – ініціалізація моделі, виконання Grid Search крос-валідації, тренування моделі та виконання прогнозів через тестову вибірку.

5 РЕЗУЛЬТАТИ

Зупиняємо вибір на мові Python за рахунок більш простої майбутньої інтеграції, а також більшої кількості обчислювальних можливостей. Зокрема, використовуватимемо наступні бібліотеки Python: scikit-learn (для побудови моделі, виокремлення ознак та застосування метрик); pandas, numpy (для збереження та маніпуляції даними); spacy (для розмічування частин мови); nltk (для видалення шумових слів); genism (для використання Word2Vec моделі); seaborn, matplotlib (для візуалізації).

– Розпізнання пропаганди на рівні статті (рис. 9). Переглянемо розподіл даними за їх класами для того, аби зрозуміти рівень їх збалансованості.

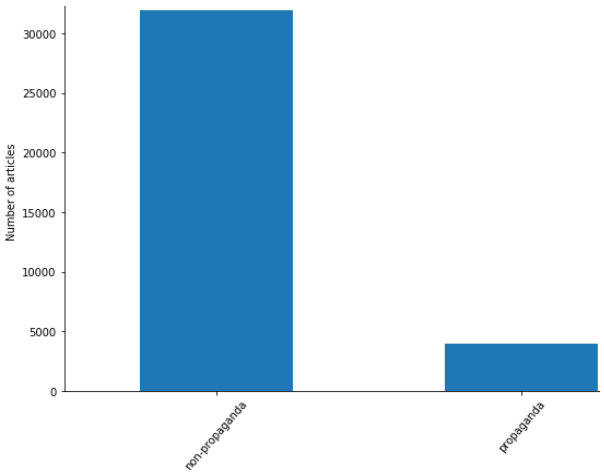


Рисунок 9 – Графік розподілу даних для розпізнання пропаганди на рівні документу за класами

Як бачимо, дані не є збалансованими та у наборі суттєво переважають статті не пропагандистського характеру. А саме, набір даних містить 31972 статті, марковані як «не пропаганда» та 4021 статті, марковані як «пропаганда». Перед тим, як розпочати процес виділення ознак, необхідно виконати деякі операції для очистки та підготовки даних. В першу чергу, необхідно виконати операцію конвертування кожного слова у наборі даних до нижнього регістру для того, аби під час процесу векторизації два ідентичних слова, котрі починаються із різного регістру літер, не враховувалися у якості окремих токенів. Для цього виконуємо наступне перетворення:

```
data['article'] = data['article'].apply(lambda
x: " ".join(x.lower() for x in x.split()))
```

Далі вилучаємо із набору даних пунктуаційні знаки, оскільки на даному рівні вирішення задачі пунктуація не є інформативною ознакою, проте під час процесу векторизації буде рахуватися у якості окремого токену, що може призвести до формування зайвого шуму у даних. Виконуємо перетворення:

```
data['article'] =
data['article'].str.replace('[^\w\s]','')
```

Вилучаємо шумові слова (не несуть змістовного навантаження). Для цього використаємо вбудований корпус шумових слів бібліотеки nltk та у циклі вилучимо їх із статей.

```
stop = nltk.stopwords.words('english')
data['article'] = data['article'].apply(lambda
x: " ".join(x for x in x.split() if x not in
stop))
```

Після цього можемо розділити дані на навчальну та тестову вибірки.

```
X = data['article']
y = data['label']
X_train, X_test, y_train, y_test =
train_test_split(X, y, test_size=0.2,
random_state=42)
```


DOI 10.15588/1607-3274-2024-2-13

Вилучення ознак. Виконуємо векторизацію тексту. Використаємо клас CountVectorizer із scikit-learn.

```
vectorizer = CountVectorizer(analyzer='word',
token_pattern=r'\w{1,}', ngram_range=(1,2),
strip_accents='unicode', min_df=3, max_df=0.5)
X_train = vectorizer.fit_transform(X_train)
X_test = vectorizer.transform(X_test)
```

У якості вихідного результату методи fit та transform екземпляру класу CountVectorizer продукують розріджену матрицю, розмірність якої дорівнює кількості унікальних слів у тексті. Переконаємося у тому, що після трансформації кількість атрибутів навчальної та тестової вибірки співпадають (рис. 10).

```
: X_train.shape
: (28794, 605048)

: X_test.shape
: (7199, 605048)
```

Рисунок 10 – Перевірка розмірностей навчальної та тестової вибірок даних на рівні документу

Далі виконаємо TF-IDF трансформацію. Використовуємо клас TfidfTransformer із scikit-learn.

```
transformer = TfidfTransformer(use_idf=True,
smooth_idf = True)
X_train = transformer.fit_transform(X_train)
X_test = transformer.transform(X_test)
```

Перетворення відбується для попередньо сформованої розрідженої матриці векторизованих текстових даних. Побудова моделі.

```
LogisticRegression(C=1.0,
    class_weight='balanced',
    dual=False, fit_intercept=True,
    intercept_scaling=1,
    l1_ratio=None, max_iter=100,
    multi_class='auto', n_jobs=None,
    penalty='12', random_state=None,
    solver='lbfgs', tol=0.0001,
    verbose=0, warm_stert=False)
```

Для моделювання використаємо клас LogisticRegression із бібліотеки scikit-learn.

```
model = LogisticRegression(penalty='12',
class_weight='balanced', solver='lbfgs')
model.fit(X_train, y_train)
```

Серед параметрів вказуємо penalty='l2', тобто для регуляризації модель використовуватиме метод гребеневої регресії, а solver='lbfgs' означає, що для оптимізації модель використовуватиме алгоритм Бройдена-Флетчера-Гольдфарба-Шанно із обмеженим використанням пам'яті.

Підготовка даних. Побудуємо графік розподілу даних за категоріями для оцінки рівня незбалансованості даних (рис. 11). Набір даних не є збалансованим та у ньому знову переважують речення не пропагандистського характеру.

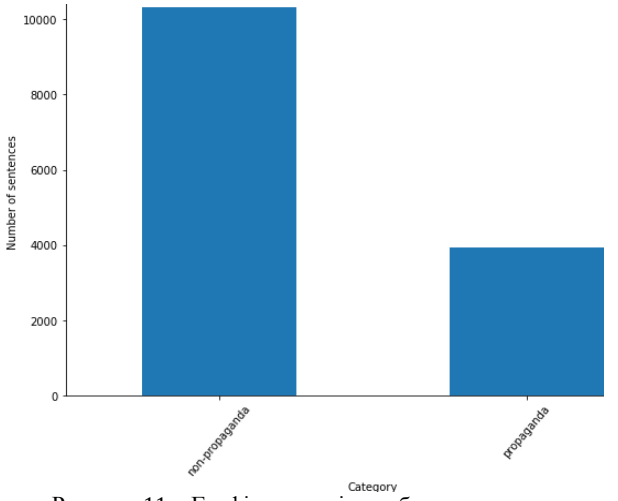


Рисунок 11 – Графік розподілу набору даних для розпізнання пропаганди а рівні речення за класами

Далі повторюємо процес видалення шумових слів із набору даних. Після цього знову конвертуємо текстові дані до нижнього регістру.

```
stop = stopwords.words('english')
data['sentence'] = data['sentence'].apply(lambda
x: " ".join(x for x in x.split() if x not in
stop))
data['sentence'] = data['sentence'].apply(lambda
x: " ".join(x.lower() for x in x.split()))
```

На даному рівні задачі не видаляємо пунктуаційні знаки із речень, оскільки вони знадобляться на етапі виділення ознак. Виконуємо розмічення частин мови. Для цієї операції використовуємо бібліотеку spacy.

```
nlp = spacy.load('en')
def tag(sentence):
                global nlp
                doc = nlp(sentence)
                return "
".join([f'{x.text}_{x.tag_}_{x.lemma_}' for x in
doc])
sentences_pos = copy.deepcopy(data['sentence'])
tagged = pos_tagging(sentences_pos)
data['tagged'] = tagged
```

Маркуємо кожне речення у наборі даних за наявністю у ньому фігурних лапок. Це дозволяє відслідкувати техніку пропаганди як «Апеляція до авторитету». Ініціалізуємо відповідну функцію.

```
def get_quotations(sentences):
    result = []
    for sentence in sentences:
        match = 1 if '"' in sentence else 0
        result.append(match)
    return np.array(result).reshape(-1, 1)
```

Далі перевіряємо кожне речення на наявність так званих «блискучих слів». Прикладами таких слів є «патріотизм», «свобода», «сила», «ідея» тощо. Виконуємо цю операцію для ідентифікації таких методики пропаганди, як «Розмахування прапором» та «Слоган». Для цього обчислюємо кількість співпадінь між словами у речення та словами у лексиконі «блискучих слів» та розділяємо це значення на загальну кількість слів у речення для того, щоб нормалізувати коефіцієнт. Формуємо відповідний

DOI 10.15588/1607-3274-2024-2-13

лексикон у вигляді текстового файлу із «блискучими словами» та ініціалізуємо відповідну функцію.

```
def get_glitter(tagged):
    filename = 'glitter_words.txt'
    glitters = []
    append = glitters.append
    with open(os.path.join(LEXICONS_PATH,
filename), encoding='utf-8') as f:
        for line in f.readlines():
            append(line.replace('\n', ''))
    result = []
    for sentence in tagged:
        words = 0
        matches = 0
        for wline in sentence.split():
            try:
                w, t, l = wline.split("_")
            except:
                continue
            w = w.lower()
            l = l.lower()
            words+=1
            if l in glitters or w in glitters:
                matches+=1
        if words == 0:
            result.append(0)
        else:
            result.append(matches/words)
    return np.array(result).reshape(-1, 1)
```

Використовуємо аналогічний підхід для пошуку у реченнях слова-підсилювачів («неймовірно», дуже», «абсолютно», «тощо) та абсолютних займенників («усі», «ніхто» тощо). Намагаємося відповідно ідентифікувати таки методики пропаганди як «Загальна платформа» та «Навантажена мова». Формуємо аналогічні лексикони та іцініалізуємо відповідні функції. У якості останнього кроку, векторизуємо текстові дані за допомогою моделі Word2Vec. Для цієї задачі використовуємо заздалегідь натреновану на наборі даних із соціальної мережі Twitter модель, котра має 200 вимірів. Використовуємо бібліотеку gensim.

```
w2v_file = os.path.join(WORD2VEC_PATH,
'twitter.27B.200d.txt')
w2v_model =
KeyedVectors.load_word2vec_format(w2v_file,
binary=False)
def w2v_vectorize(tagged):
    global w2v_model
    X = []
    ndims = 200
    for sentence in tagged:
        words = []
        for wline in sentence.split():
            try:
                w, t, l = wline.split("_")
            except:
                continue
            words.append(w)
        row_data = np.mean([w2v_model[w] for w
in words if w in w2v_model] or
                           [np.zeros(ndims)],
axis=0).tolist()
        X.append(row_data)
    X = np.array(X)
    X_std = (X - X.min(axis=0)) / (X.max(axis=0)
- X.min(axis=0))
    X_scaled = X_std * (1 - 0) + 0
    return X_scaled
```

Формуємо новий DataFrame на основі уже існуючого з використанням усіх описаних операцій.

```
word2vec_features =
w2v_vectorize(data['tagged'])
word2vec_columns = [f'dim{x}' for x in
range(200)]
glitter_words = get_glitter(data['tagged'])
quotations = get_quotations(data['sentence'])
intensifiers = get_intensifiers(data['tagged'])
absolutes = get_absolutes(data['tagged'])
X = pd.DataFrame(word2vec_features,
columns=[word2vec_columns])
X['quotations'] = quotations
X['glitter_words'] = glitter_words
X['intensifiers'] = intensifiers
X['absolutes'] = absolutes
y = data['label']
```

Розділяємо датасет на навчальну/тестову вибірки.

```
X_train, X_test, y_train, y_test =
train_test_split(X, y, test_size=0.2,
random_state=42)
```

Використовуємо алгоритм крос-валідації Grid Search для пошуку найкращих параметрів, ініціалізуємо та тренуємо модель.

```
lr_model = LogisticRegression()
penalty = ['l1', 'l2']
C = np.logspace(0, 4, 10)
hyperparameters = dict(C=C, penalty=penalty)
clf = GridSearchCV(lr_model, hyperparameters,
refit='f1', cv=5)
best_model = clf.fit(X_train, y_train)
```

Отримані найкращі параметри моделі є наступними: penalty=’l2’, C=7.74.

6 ОБГОВОРЕННЯ

Проаналізуємо та оцінимо роботу моделі розпізнання пропаганди на рівні статті. Для цього виконуємо прогнози на основі тестової вибірки. Побудуємо матрицю помилок (рис. 12). Матрицю помилок можемо інтерпретувати наступним чином: моделі вдалося коректно класифікувати 6097 не пропагандистських статей та 694 статті пропагандистського характеру. 123 пропагандистські статті та 285 не пропагандистських статей були класифіковані невірно. Отримана оцінка моделі: 0.9433254618697041.

```
n = len(conf_matrix)
plt.figure(figsize = (n, n))
sns.heatmap(conf_matrix, annot=True)
plt.title("Confusion Matrix")
plt.ylabel("Actual Value")
plt.xlabel("Predicted Value")
plt.show()
```

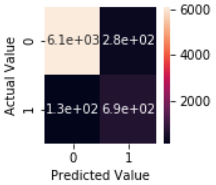


Рисунок 12 – Побудована матриця помилок для моделі розпізнання пропаганди на рівні статті

Після цього виконаємо аналогічні дії для моделі розпізнання пропаганди на рівні речень. Отже, знову виконуємо прогнозування на основі тестової вибірки. Будуємо аналогічну матрицю помилок (рис. 13).

DOI 10.15588/1607-3274-2024-2-13

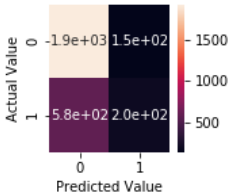


Рисунок 13 – Матриця помилок на рівні речень

Матриця помилок може бути інтерпретована наступним чином: модель успішно класифікувала 1917 не пропагандистських статей та 205 пропагандистських статей, проте 585 пропагандистських статей та 146 не пропагандистських статей були класифіковані невірно. Оцінка моделі становить: 0.7437784787942516. Обидві оцінки є задовільними, проте модель для розпізнання пропаганди на рівні документу впоралася краще. До того, враховуючи незбалансованість обидвох наборів даних бачимо, що друга модель (для розпізнання пропаганди на рівні речень) невірно класифікувала більше екземплярів даних, котрі були марковані як пропагандистські. Це може свідчити про те, що модель є недостатньо специфічною, тобто у майбутніх розробках необхідно використати складніший, більш комплексний алгоритм та вилучити більше специфічних ознак для того, аби мати можливість точніше ідентифікувати інші методики пропаганди.

ВИСНОВКИ

На етапах опрацювання дезінформації пропонується новий метод аналізу пропаганди для ідентифікації ознак та зміни динаміки поведінки скоординованих груп на основі машинного навчання.. Впровадження отриманих результатів дозволить суттєво скоротити час на прийняття найбільш адекватного рішення щодо впровадження заходів боротьби з дезінформацією стосовно ідентифікованих скоординованих груп генерування, дезінформації фейків і пропаганди.

Під час виконання роботи реалізовано дві моделі для розпізнання пропаганди у текстових даних – на рівні статті та на рівні речення. Для цього розв’язано задачу бінарної класифікації тексту. Обидві моделі побудовані на основі логістичної регресії, у процесі підготовки даних та виокремлення ознак застосовано такі методи, як векторизація за моделлю «Торба слів», TF-IDF векторизація, розмічування частин мови, вбудовування слів за допомогою двошарової нейронної мережі Word2Vec, а також ручні методи виокремлення ознак, котрі націлені на ідентифікацію конкретних методик політичної пропаганди у текстах. Проаналізовано аналоги розроблюваного проекту, досліджено ПО (застосування пропаганди у ЗМІ та основі методики її продукування). Програмну реалізацію виконано за допомогою Python, із використанням бібліотек scikit-learn, pandas, numpy, spacy, nltk, genism, matplotlib, seaborn. Отримана оцінка моделі для розпізнання пропаганди на рівні статті: 0.9433254618697041, а на рівні речень: 0.7437784787942516.

Очікувані результати виконання проєкту:

– вперше розроблені основи та основні принципи синтезованої інформаційної технології автоматичного виявлення джерел дезінформації та неавтентичної поведінки користувачів чатів, що дозволить своєчасно виявляти деструктивні і підозрілі спільноти в різних соціальних мережах, визначати їх лідерів і кураторів, виявляти інформаційні загрози в повідомленнях користувачів, попереджати поширення фейковий і шкідливої інформації.

– вперше розроблено метод стилістичного аналіз та лінгвістичного опрацювання дезінформації для формування інформаційного портрету автора/бота генерування текстового контенту як частини параметрів пошуку як подібного авторського контенту, так і шляхів розповсюдження.

– вироблені критерії та параметри неавтентичної поведінки користувачів чатів для формування інформаційних портретів потенційних розповсюджувачів дезінформації та виявлення маршрутів та механізмів розповсюдження, частоти генерування фейків, тематики та ключові слова, характерні для відповідної групи.

NLP процесу визначення контенту як фейку/не феку є складним процесом, так як дуже залежить не лише від швидкості/якості попередньо зібраного/ інтегрованого та опрацьованого контенту (блокований/неблокований на певному регіоні, теми контенту) але від ефективно підібраної моделі машинного навчання на тренувальних датасетах. Зазвичай фейк не блокується. Мета його створення – по швидше розповсюдити як у всьому світі, так і на тих регіонах, де зазвичай правдива інформація (не фейк) потенційна може блокуватися (не гарантовано). Якщо не фейк інформація заблокована на певній території, а протилежна інформація (фейк) з цієї території розповсюджується – то шанс ідентифікувати фейк збільшується. Якщо і нефейк не заблокований та фейк паралельно вільно розповсюджується, тут методи NLP не допоможуть. Вони лише можуть промаркувати дві множини протилежними поясненнями щодо події/явища. І лише при додаткових статистичних дослідженнях можна ідентифікувати як множина з фейками, а яка ні. Складність ще полягає в самій мові контенту, зокрема в українській. Для порівняння з англомовним контентом українська/російська мови є досить складними для автоматичного опрацювання, особливо при аналізі семантики та розбудові онтології. Стандартні та традиційні методи, які застосовують для опрацювання англомовних тестів, не підходять для опрацювання слов’янських мов, в тому числі для виявлення дезінформації та особливостей стилістики авторів-генераторів фейків та пропаганди. Аналогічно крім того що неавтентична поведінка користувачів чатів як людей і ботів відрізняється, так і відрізняється людей з різною вмотивованістю (віра в

DOI 10.15588/1607-3274-2024-2-13

пропаганду, робота за гроші, просто одна з видів вандалізму та так би мовити дозвілля), національністю, освітою, статтю, ментальністю, рівнем знання мови тексту, ступенем віри, інтелектом, тощо. Це все значно впливає на процес визначення критеріїв поведінки різних спільнот та в межах однієї спільноти, що в свою чергу значно впливає на формування інформаційного портрету неавтентичної поведінки користувачів різних чатів (те що властиве для пропагандиста-мусульманина, суттєво відрізняється для представника рашки або днр/лнр).

Обґрунтування практичної цінності запланованих результатів проєкту для економіки та суспільства.

– Зменшення обсягів дезінформації, фейків та пропаганди та частоти/регулярності публікації за рахунок відстежування стилістично подібного контенту та маршрутів розповсюдження.

– Зменшення негативного впливу дезінформації на настрої суспільства та зменшення ступеня керування громадською думкою через розповсюдження пропаганди в інформаційній війні. Наприклад пригнічення психіки молоді (у т.ч. порушення психіки, доведення до летальних наслідків), спонукання їх до асоціальної поведінки, формування груп громадської непокори чи агресивної поведінки за сфабрикованими приводами, аналіз соціальних наслідків кібератак тощо.

– Зменшення ціни пошуку, ідентифікації та блокування дезінформації, авторів/цільових розповсюджувачів та джерел.

Розробка вищеописаних методів спрямована на виявлення загроз, стороннього втручання (атак) на ранніх стадіях, класифікацію загроз за видами та подальше протистояння кожному виду загроз.

ЛІТЕРАТУРА

1. Zhao Y. Detecting health misinformation in online health communities: Incorporating behavioral features into machine learning based approaches / Y. Zhao, J. Da, J. Yan // Information Processing & Management. – 2021. – Vol. 58(1). – P. 102390. DOI: 10.1016/j.ipm.2020.102390
2. Hartmann M. Mapping (dis-)information flow about the MH17 plane crash / M. Hartmann, Y. Golovchenko, I. Augenstein, // arXiv. – Access mode: https://arxiv.org/abs/1910.01363.
3. Prokipchuk O. Ukrainian Language Tweets Analysis Technology for Public Opinion Dynamics Change Prediction Based on Machine Learning / O. Prokipchuk, V. Vysotska // Radio Electronics, Computer Science, Control. – 2023. – Vol. 2(2023). – P. 103–116. DOI: 10.15588/1607-3274-2023-2-11
4. Ahmed S. Classification of Censored Tweets in Chinese Language using XLNet / S. Ahmed, A. Kumar // Fourth Workshop on NLP for Internet Freedom: Censorship, Disinformation, and Propaganda :, Association for Computational Linguistics, Online, 2021 : proceedings. – Online: ACL, 2021. – P. 136–139. DOI: 10.18653/v1/2021.nlp4if-1.21
5. NLP Tool for Extracting Relevant Information from Criminal Reports or Fakes/Propaganda Content / [V. Vysotska, S. Mazepa, L. Chyrun et al.] // Computer Sciences and Information Technologies : 17th International Conference, Lviv, 2022, November. – Lviv: IEEE, 2021. – P. 93–98. DOI: 10.1109/CSIT56902.2022.10000563
6. Oliinyk V. A. Propaganda Detection in Text Data Based on NLP and Machine Learning / [V. A. Oliinyk, V. Vysotska, Y. Burov et al.] // CEUR Workshop Proceedings. – 2020. – Vol. 2631. – P. 132–144.
7. Bjola C. Propaganda in the digital age / C. Bjola // Global Affairs. – 2017. – Vol. 3(3). – P. 189–191. DOI: 10.1080/23340460.2017.1427694
8. Vosoughi S. The spread of true and false news online / S. Vosoughi, D. Roy, S. Aral // Science. – 2018. – Vol. 359(6380). – P. 1146–1151. DOI: 10.1126/science.aap9559
9. Propaganda Definitions. – Access mode: https://propaganda.qcri.org/annotations/definitions.html
10. Framing and Agenda-setting in Russian News: a Computational Analysis of Intricate Political Strategies / [A. Field, D. Kliger, S. Wintner et al.] // arXiv. – Access mode: – https://arxiv.org/abs/1808.09386
11. Garcia-Marín J. The Use of Supervised Learning Algorithms in Political Communication and Media Studies: Locating Frames in the Press / J. Garcia-Marín, A. Calatrava // Pamplona. – 2018. – Vol. 31(3). – P. 175–188. DOI: 10.15581/003.31.3.175-188
12. nginx. – Access mode: https://fgz.texty.org/
13. texty.org.ua. How Texty detects and makes sense of manipulative news. – Access mode: https://medium.com/@texty.org.ua/how-texty-detects-and-makes-sense-of-manipulative-news-1f43d33936eb
14. Hein V. Propaganda detection in Russian and American news coverage about the war in Ukraine through text classification / V. Hein // Diploma Thesis, Technische Universität Wien. – 2023. DOI: 10.34726/hss.2023.104640
15. Ceușan I. F. European Union policies and strategies to counter Russian propaganda and disinformation / I. F. Ceușan // L'Europe Unie. – 2023. – Vol. 19(19). – P. 113–122.
16. Perdoor S. Fake News Detection with LSTM and NLP – ProRew1 / S. Perdoor. – Access mode: https://www.kaggle.com/code/superrajdoor/fake-news-detection-with-lstm-and-nlp-prorew1/input //
17. Duratnir İ. Fake News Detection with NLP and LSTM / İ. Duratnir. – Access mode: https://www.kaggle.com/code/ilaydadu/fake-news-detection-with-nlp-and-lstm
18. propaganda-detection-our-data. – Access mode: https://www.kaggle.com/datasets/vladimirsydor/propaganda-detection-our-data

Accepted 09.02.2024.
Received 27.04.2024.

UDC 004.9

INFORMATION TECHNOLOGY FOR RECOGNIZING PROPAGANDA, FAKES AND DISINFORMATION IN TEXTUAL CONTENT BASED ON NLP AND MACHINE LEARNING METHODS

Vysotska V. – Dr. Sc., Associate Professor of Information Systems and Networks Department, Lviv Polytechnic National University, Lviv, Ukraine.

DOI 10.15588/1607-3274-2024-2-13

ABSTRACT

Context. The research is aimed at the application of artificial intelligence for the development and improvement of means of cyber warfare, in particular for combating disinformation, fakes and propaganda in the Internet space, identifying sources of disinformation and inauthentic behavior (bots) of coordinated groups. The implementation of the project will contribute to solving the important and currently relevant issue of information manipulation in the media, because in order to effectively fight against distortion and disinformation, it is necessary to obtain an effective tool for recognizing these phenomena in textual data in order to develop a further strategy to prevent the spread of such data.

Objective of the study is to develop or automatic recognition of political propaganda in textual data, which is built on the basis of machine learning with a teacher and implemented using natural language processing methods.

Method. Recognition of the presence of propaganda will occur at two levels: at the general level, that is, at the level of the document, and at the level of individual sentences. To implement the project, such feature construction methods as the TF-IDF statistical indicator, the "Bag of Words" vectorization model, the marking of parts of speech, the word2vec model for obtaining vector representations of words, as well as the recognition of trigger words (reinforcing words, absolute pronouns and "shiny" words). Logistic regression was used as the main modeling algorithm.

Results. Machine learning models have been developed to recognize propaganda, fakes and disinformation at the document (article) and sentence level. Both model scores are satisfactory, but the model for document-level propaganda recognition performed almost 1.2 times better (by 20%).

Conclusions. The created model shows excellent results in recognizing propaganda, fakes and disinformation in textual content based on NLP and machine learning methods. The analysis of the raw data showed that the propaganda recognition model at the document (article) level was able to correctly classify 6097 non-propaganda articles and 694 propaganda articles. 123 propaganda articles and 285 non-propaganda articles were misclassified. The obtained estimate of the model: 0.9433254618697041. The sentence-level propaganda recognition model successfully classified 205 propaganda articles and 1917 non-propaganda articles. The model score is: 0.7437784787942516 (but 731 articles were incorrectly classified).

KEYWORDS: disinformation, fake, propaganda, linguistic analysis, natural language processing, machine learning, cyber warfare, artificial intelligence, semantic analysis, information security.

REFERENCES

1. Zhao Y., Da J., Yan J. Detecting health misinformation in online health communities: Incorporating behavioral features into machine learning based approaches, *Information Processing & Management*, 2021, Vol. 58(1), P. 102390. DOI: 10.1016/j.ipm.2020.102390
2. Hartmann M., Golovchenko Y., Augenstein I. Mapping (dis-)information flow about the MH17 plane crash, *arXiv*. Access mode: https://arxiv.org/abs/1910.01363.
3. Prokipchuk O., Vysotska V. Ukrainian Language Tweets Analysis Technology for Public Opinion Dynamics Change Prediction Based on Machine Learning, *Radio Electronics, Computer Science, Control*, 2023, Vol. 2(2023), pp. 103–116. DOI: 10.15588/1607-3274-2023-2-11
4. Ahmed S., Kumar A. Classification of Censored Tweets in Chinese Language using XLNet, *Fourth Workshop on NLP for Internet Freedom. Censorship, Disinformation, and Propaganda, Association for Computational Linguistics, Online, 2021, proceedings. Online: ACL*, 2021, pp. 136–139. DOI: 10.18653/v1/2021.nlp4if-1.21
5. Vysotska V., Mazepa S., Chyrun L., Brodyak O., Shakleina I., Schuchmann V. NLP Tool for Extracting Relevant Information from Criminal Reports or Fakes/Propaganda Content, *Computer Sciences and Information Technologies : 17th International Conference*, Lviv, 2022, November. Lviv, IEEE, 2021, pp. 93–98. DOI: 10.1109/CSIT56902.2022.10000563
6. Oliinyk V. A., Vysotska V., Burov Y., Mykich K., Fernandes V. B. Propaganda Detection in Text Data Based on NLP and Machine Learning, *CEUR Workshop Proceedings*, 2020, Vol. 2631, pp. 132–144.
7. Bjola C. Propaganda in the digital age, *Global Affairs*, 2017, Vol. 3(3), pp. 189–191. DOI: 10.1080/23340460.2017.1427694
8. Vosoughi S., Roy D., Aral S. The spread of true and false news online, *Science*, 2018, Vol. 359(6380), pp. 1146–1151. DOI: 10.1126/science.aap9559
9. Propaganda Definitions. Access mode: https://propaganda.qcri.org/annotations/definitions.html
10. Field A. Kliger D., Wintner S., Pan J., Jurafsky D., Tsvetkov Y. Framing and Agenda-setting in Russian News: a Computational Analysis of Intricate Political Strategies, *arXiv*. Access mode: https://arxiv.org/abs/1808.09386
11. Garcia-Marín J., Calatrava A. The Use of Supervised Learning Algorithms in Political Communication and Media Studies: Locating Frames in the Press, *Pamplona*, 2018, Vol. 31(3), pp. 175–188. DOI: 10.15581/003.31.3.175-188
12. nginx. – Access mode: https://fgz.texty.org/
13. texty.org.ua. How Texty detects and makes sense of manipulative news. Access mode: https://medium.com/@texty.org.ua/how-texty-detects-and-makes-sense-of-manipulative-news-1f43d33936eb
14. Hein V. Propaganda detection in Russian and American news coverage about the war in Ukraine through text classification, *Diploma Thesis*, Technische Universität Wien, 2023. DOI: 10.34726/hss.2023.104640
15. Ceușan I. F. European Union policies and strategies to counter Russian propaganda and disinformation, *L'Europe Unie*, 2023, Vol. 19(19), pp. 113–122.
16. Perdoor S. Fake News Detection with LSTM and NLP – ProRew1. Access mode: https://www.kaggle.com/code/superrajdoor/fake-news-detection-with-lstm-and-nlp-prorew1/input //
17. Duratnir İ. Fake News Detection with NLP and LSTM / İ. Duratnir. Access mode: https://www.kaggle.com/code/ilaydadu/fake-news-detection-with-nlp-and-lstm
18. propaganda-detection-our-data. Access mode: https://www.kaggle.com/datasets/vladimirsydor/propaganda-detection-our-data

DOI 10.15588/1607-3274-2024-2-13

UDC 004.415

METHOD AUTOMATED CLASS CONVERSION FOR COMPOSITION IMPLEMENTATION

Kungurtsev O. B. – PhD, Professor, Professor of the Software Engineering Department, Odessa Polytechnic National University, Odessa, Ukraine.

Bondar V. R. – Student of the Software Engineering Department, Odessa Polytechnic National University, Odessa, Ukraine.

Gratilova K. O. – Student of the Software Engineering Department, Odessa Polytechnic National University, Odessa, Ukraine.

Novikova N. O. – PhD, Associate Professor of the Department of Technical Cybernetics and Information Technologies named after professor R. V. Merct, Odessa National Maritime University, Odessa, Ukraine.

ABSTRACT

Context. Using the composition relation is one of the most effective and commonly used ways to specialize classes in object-oriented programming.

Objective. Problems arise when “redundant” attributes are detected in an inner class, which are not necessary for solving the tasks of a specialized class. To work with such attributes, the inner class has corresponding program methods, whose usage not only does not solve the tasks of the specialized class, but can lead to errors in its work. The purpose of this work is to remove “redundant” attributes from the inner class, as well as all methods of the class directly or indirectly (through other methods) using these attributes.

Method. A mathematical model of the inner class was developed, which allowed us to identify “redundant” elements of the class. The method of internal class transformation is proposed, which, based on the analysis of the class code, provides the developer with information to make a decision about “redundant” attributes, and then in the automated mode gradually removes and transforms the class elements.

Result. To approbate the proposed solutions, a software product Composition Converter was developed. Experiments were carried out to compare the conversion of classes in “manual” and automated modes. The results showed a multiple reduction of conversion time in the automated mode.

Conclusions. The proposed method of automated transformation of the inner class according to the tasks of the outer class when implementing composition allows to significantly reduce the time or the number of errors when editing the code of the inner class. The method can be used for various object-oriented languages.

KEYWORDS: object-oriented programming, classes, composition, syntactic analysis, class transformation.

ABBREVIATIONS

OOP – object-oriented programming.

NOMENCLATURE

attrName is a attribute identifier;

attrType is a attribute type;

cHead is a class header;

cHead1C2 is a new name of the inner class, reflecting the use in the outer class;

cName is a class name;

cName1 is a parent class name for cName (can be empty);

destr is a class destructor (if provided by the programming language);

fName is a method name;

mArgs is a set of method arguments;

mAttr is a set of class attributes;

mAttr1 is a set of attributes of class C1;

mAttr` is a a subset of the mAttr set containing redundant attributes;

mConstr is a set of class constructors;

mFunc is a set of ordinary methods of the class;

mMeth is a set of class methods;

mMeth1 is a set of methods of class C1;

mMeth11 is a methods of class C1 that are independent of mAttr`;

mMethR is a set of edited methods that have become independent of mAttr`;

mOperand is a set of operands:

mOperator is a set of method operators;

retType is a type of return value (empty for constructors and destructor).

INTRODUCTION

In object-oriented programming (OOP) there are two main ways of creating specialized classes based on existing ones – inheritance and extending the functionality of some class by using another class as an object attribute [1, 2]. Let us call the specialized class an outer class and the class of the included object an inner class. The object control of the inner class by an object of the outer class can be full and partial. In the first case the connection between the classes is called composition, and in the second case – aggregation. To implement

DOI 10.15588/1607-3274-2024-2-14

composition, the outer and inner classes must have the following relations [3]:

– the inner class is a part of the outer class;

– the inner class can belong to only one outer class;

– the inner class (object) is controlled by the outer class (object);

– the inner class (object) does not know about the existence of the outer class (object).

Aggregation involves sharing of the inner class by several outer classes. In this case, conflicts of interests of outer classes may arise.

In practice, the use of composition is observed much more frequently than the use of aggregation. This work solves the problems associated with the composition usage. Composition has two significant advantages over inheritance [3]:

– allows adding additional functionality to the outer class with minimal changes in its structure;

– significantly reduces debugging time of the outer class, because the inner class is already ready-to-work.

The notion of a "ready-to-work class" requires explanation. If specializing the outer class by inheritance is understood as continuing to work on that class, then connecting the inner class involves searching for a suitable class from some library. By definition, the inner class in the vast majority of cases was not created for use in a particular outer class. To find a suitable inner class, candidates that provide the required functionality are considered. In many cases, a suitable candidate for the inner class has functionality beyond the required one. "Redundant functionality" consists of the existence of "redundant" methods and attributes. For example, in order to assign a bus to a driver to perform a trip, we can enter the attribute "Bus" into the class "Driver", which is a class. The "Bus" class may have many attributes and methods that model its engine, electrical system, running gear, repair information, etc., while the composition needs only the brand, registration number, number of seats, and possibly a few more attributes and corresponding methods. In case of the presence of "redundant" structural units in the inner class, the following problems arise [4]:

– when initializing a "redundant attribute", information is needed that is not defined by the task, which the outer class solves. This may be a source of initialisation errors;

– when working with an object of an inner class, it is possible to use methods directly or indirectly, which do not solve the tasks of the outer class, but introduce errors in their solution;

– methods that are "useful" from the point of view of tasks solved by the outer class may perform some actions on "redundant" attributes, which may also cause errors.

Thus, there is a problem of identifying, removing or "neutralizing" redundant attributes and methods in a class that is chosen as an inner class during composition.

According to the above problem, the following research tasks have been formulated:

– create a model of the inner class;

– develop a method to identify and remove "redundant" attributes and methods of the inner class, as well as to correct methods dependent on the deleted class elements.

1 PROBLEM STATEMENT

Suppose there is some program class c=<cHead, mAttr, mMeth>. When using this class as an object of another class *c2* (composition), a subset of attributes *mAttr`* turned out to be redundant. It is necessary to perform the transformation

$c \Rightarrow c1$,

where $c1 = < cHead1C2, mAttr1, mMeth1 >$.

Wherein $mAttr1 = mAttr \cap mAttr'$,

$mMeth1 = mMeth11 \cap mMethR$,

where $mMeth11 \in mMeth$,

$mMeth' = F(mAttr') \Rightarrow mMethR \neq F(mAttr')$.

2 REVIEW OF THE LITERATURE

Composition in programming languages is analyzed and applied at different levels. An attempt to develop a general approach to composition is made in work [5]. From our point of view, the recommendation to compose models for composition according to specific conditions is useful in this work.

In work [6], composition is considered at the level of language constructs of various domain-oriented languages. Of interest is a framework that allows creating a language from known constructs for a new subject area. Some principles of framework construction may find application to the present study.

In work [7] the principle of composition is applied at the level of individual operators and in [8] at the level of individual expressions, but it is also actually about making changes to programming language constructs rather than to program elements.

The conditions under which class-level composition has advantages compared to inheritance are described in sufficient detail in the literature [3,9], but the authors do not analyze the problems arising in its implementation.

In work [4] the composition problems are formulated, but the model is not developed. Therefore, the proposed solution is applicable only for a special case.

In work [10] it is proposed to allocate key classes for software understanding. This idea is relevant in the realization of composition if we represent "useful" attributes as key attributes. The authors did not formulate the task of any work with "redundant" classes (elements).

The issue of identifying and analyzing the effectiveness of class attributes is considered in work [11]. However, the study concerns only the attributes, the choice of which corresponds to the purpose of class creation, whereas in the conditions of composition the initial purpose of class application can be slightly changed.

In work [12], a class model is proposed, which represents its functionality quite completely, but does not provide for changes in the class.

DOI 10.15588/1607-3274-2024-2-14

For the task of finding inheritance relations, an appropriate class model was created in [13]. The model provides class transformation by redistributing methods and attributes between classes, which is also applicable for this work, but does not allow identifying and removing "redundant" attributes and methods.

Class transformation is based on the extraction of certain constructs. Syntactic code analysis is considered in works [14, 15], where the main focus is on parser performance and creation of new convolution algorithms, whereas for this work the main requirement is the extraction of only certain code constructs.

A number of approaches to static code analysis [16] are applicable in the conditions of this work when "redundant" elements are used together with "useful" ones within one operator. An interesting proposal regarding combining static code analysis with object-oriented structure extraction is made in [17], but the authors do not offer an acceptable practical implementation of their project.

The work [18] shows the role of refactoring on the quality of object-oriented code. Accepting the recommendations of the authors of the work, the present study envisages not only checking the code for a given functionality, but also for compliance with design patterns [19].

3 MATERIALS AND METHODS

Class model.

Let's represent the class as a tuple:

$$c = <cHead, mAttr, mMeth>. \quad (1)$$

Let's represent the class header as a tuple:

$$cHead = <cName, cName1>. \quad (2)$$

Let's represent each attribute from the set *mAttr* as:

$$Attr = <attrName, attrType>. \quad (3)$$

Let's represent the set of methods as a tuple:

$$mMeth = <mFunc, mConstr, destr> . \quad (4)$$

Any element from mMeth has the form

$$mMeth_i = < fName_i, mArgs_i, retTipe_i, mOperator_i >. \quad (5)$$

Any operator is represented as a set of operands (variables, constants, function calls)

$$operator = mOperator.$$

Method of class transformation by removing redundant elements.

Initial data: some class *C*, which contains redundant attributes and methods from the point of view of its usage in composition.

Let's consider the case when the composite class is not inherited from another class.

First step. Let's analyze the set of attributes *mAttr* and form on its base the set of "redundant" attributes *mAttr'* and "useful" attributes *mAttr1*.

$$mAttr1 = mAttr \cap mAttr'.$$

Second step. Let's select from the set of all *mMeth* methods a subset of *mMeth'* methods, which use only attributes from the *mAttr'* set and do not use other methods of the same class (constructors and destructor are not analyzed yet).

$$mMeth' = \{ meth_i \mid mAttr1_k \notin_a meth_i \wedge meth_l \notin_m meth_i \}, i = 1, |mMeth|; k = 1, |mAttr'|; l = 1, |mMeth|,$$

where $\in_a$ and $\notin_a$ designate the use (non-use) of an attribute in a method; $\in_m$, $\notin_m$ – use (non-use) of other methods in this method.

Let's form a set of *mMeth1* methods that remain in the class:

$$mMeth1 = mMeth \cap mMeth'.$$

Third step. Let's select methods from the set *mMeth1* that do not use attributes from the set *mAttr1* and methods from the set *mMeth1*:

$$mMeth' = \{ meth_i \mid mAttr1_j \notin_a meth_i \wedge mMeth1_k \notin_m meth_i \}, i = 1, |mMeth1|; j = 1, |mAttr1|; k = 1, |mMeth1|.$$

Form the set of methods that remain in the class:

$$mMeth2 = mMeth1 \cap mMeth'.$$

Fourth step. Let's select from the set *mMeth2* a subset of methods that require editing (*mMethForAdjustment*). This category includes methods that contain "redundant" attributes and methods along with "useful" attributes and methods.

$$mMethForAdjustment = \{ meth_i \mid \exists\, mAttr'_j \in_a meth_i \vee \exists\, meth'_k \in_m meth_i \}, i = 1, |mMeth2|; k = 1, |mMeth'|; j = 1, |mAttr'|.$$

Fifth step. From each constructor the elements associated with redundant attributes are removed.

Let's represent the set of constructors in the form

$$mConstr = \{ constr_i \}, i = 1, |mConstr|.$$

Let's represent each constructor as a set of arguments and operators:

$$constr = \{ < mArgs, mOperator > \}.$$

The constructor operators should include not only operators in the body of the constructor, but also elements of the initialization list.

Operators that do not use the "useful" attributes *mAttr1* are defined

$$mOperator' = \{ operator_j \mid mAttr1_k \notin_{op} operator_j \}, j = 1, |mOperator|; k = 1, |mAttr1|,$$

where the relation $\notin_{op}$ – designates non-use of the attribute in the body of the operator.

DOI 10.15588/1607-3274-2024-2-14

A new set of constructor operators is created

$$mOperator1 = mOperator \cap mOperator'.$$

Arguments that are used to initialize only redundant attributes are defined

$$mArgs' = \{args_j \mid args_j \notin_{op} mOperator1\}, \quad j = 1, |mArgs|.$$

A new set of the constructor arguments is created

$$mArgs1 = mArgs \cap mArgs'.$$

Let's select from the set of remaining constructor *mOperator1* operators a subset of operators that require correction (*mOperatorForAdjustment*). Operators that contain "redundant" attributes and arguments along with "useful" attributes and arguments should be included in this category.

$$mOperatorForAdjustment = \{ operator_j \mid operator_j \in mOperator1 \wedge (\exists\, mAttr'_p \in_{op} operator_j)\}, j = 1, |mOperator1|; p = 1, |mAttr'|; l = 1, |mArgs'|.$$

Sixth step (only for programming languages that use destructors). The elements associated with redundant attributes are removed from the destructor.

Let's represent the destructor as a set of operators

$$destr = mOperator.$$

Operators that do not use attributes from set *mMeth1* are defined

$$mOperator' = \{ operator_j \mid mAttr1_l \notin_{op} operator_l \}, \quad j = 1, |mOperator|; l = 1, |mAttr1|.$$

A new set of destructor operators is created

$$mOperator1 = mOperator \cap mOperator'$$

Seventh step. The class C1 is formed based on *mAttr1*, *mMeth2*, transformed constructors and destructor.

$$c1 = <cHead1C2, mAttr1, mMeth1>,$$

where the new name *cHead1C2* indicates the modification of the original class C to conform to the requirements of class C2.

The case when an aggregate class is an inheritor of another class.

Option 1. There is a code of a parent class.

First step. The method proposed above is applied to the parent class.

Second step. The method proposed above is applied to the generated class.

Option 2. The code of the parent class is inaccessible.

First step. The "redundant" attributes introduced in the inherited class are determined. For them, the method proposed above is applied without performing the fifth, sixth and seventh steps.

Second step. The "redundant" attributes of the parent class are determined.

Third step. Methods that use only "redundant" attributes of the parent class are defined. Such methods should be made "neutral" depending on the context of their use. That is, such that their call does not lead to any changes in the context of their use.

A method of the parent class with a private method can be overridden. Then the call of the corresponding method of the parent class will be possible only when referring to the parent class.

The **fourth and subsequent steps** are performed according to the method proposed earlier.

4 EXPERIMENTS

In accordance with the proposed model and method of inner class transformation, a grammar is proposed that allows to extract from the class code the attribute description, the description of a regular method, the description of a constructor, the description of a destructor, an operator, an identifier and a method argument. In order to shorten the record of a number of rules widely used in grammars of programming languages, some right parts of definitions are omitted or replaced by an ellipsis

Grammar for highlighting necessary code elements:

```
$ class = {specifier} class class_name {specifier class_name} "{" {/ (description | operator /} "}"
$ description = type description_list";"
$ description list = description_item | description list "," description_item
$ type = standard_type | user_type
$ user_type = class_name | structure_name
$ standard_type = int | float | double | char |
        boolean | .....
$ identifier = ( letter | "_") { letter | number | "_"}
$ class_name = identifier
$ method_name = identifier
$ method = { specifier } method_name "(" argument list ")" "{" { (description | operator /} } "}"
$ any_sequence_of_characters_without_";"
=.....
$ operator= any_sequence_of_characters_without_; ";" | "{" operator "}"
```

The Composition Converter software product was created to implement the developed method. The scheme of the software product operation is shown in Fig.1. The Analyzer module allows you to select elements of the inner class in accordance with the given grammar. Command line compilers are used to check the correctness of the code of the edited methods. The scheme shows the sequential transformation of the original inner class $C \rightarrow C\_1 \rightarrow C\_2 \rightarrow C\_3$ by removing "redundant" methods and attributes, as well as editing methods for which "mechanical" removal of elements is impossible.

Fig. 2 shows a window with a list of attributes of the inner class, where the programmer can indicate redundant attributes.

DOI 10.15588/1607-3274-2024-2-14

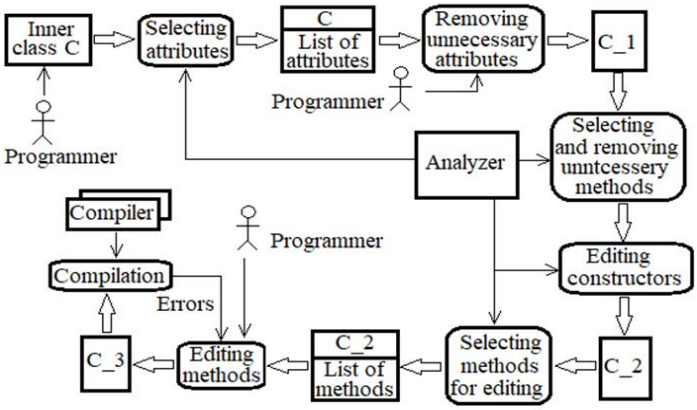


Figure 1 – Scheme of Composition Converter work

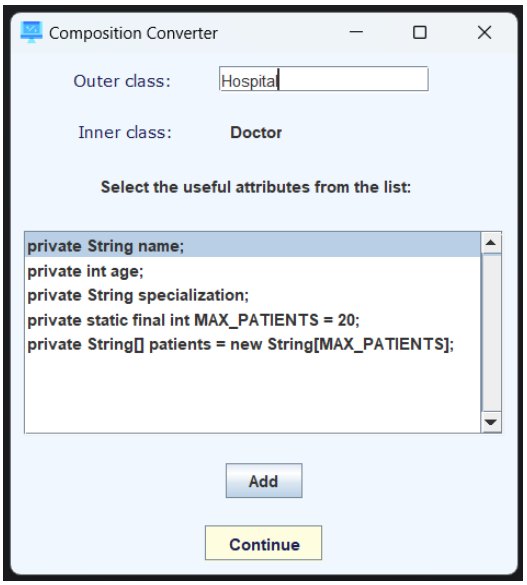


Figure 2 – Attribute selection window

5 RESULTS

A series of experiments were conducted to approbate the results of the study. The purpose of conducting the experiments was:

1. Verification of the quality of the program work.
2. Evaluation of the effectiveness of the proposed method.

In accordance with the first purpose, it was checked:

– identification of all cases of redundant constructions usage;

– deleting and automatic editing of constructions that do not require programmer intervention;

– providing the programmer with all constructions that require editing.

In accordance with the second purpose, it was determined:

– time for automated inner class transformation;

– time for “manual” inner class transformation.

For the study, 6 classes were developed from the subject areas “transport” and “health”. The number of attributes in the classes was 10, 20 and 30. The number of methods was 2–3 per attribute. 12 students from among the equally successful students of OO-programming subject were involved in the experiments. Each student performed conversion of 3 classes in “manual” mode and other 3 classes in automated mode using Composition Converter. Lists of “redundant” attributes were reported immediately before the experiments were performed.

No errors were found in the program operation at the stages of deletion and automatic transformation of class elements. Errors were observed when editing methods selected by the program. Errors in “manual” mode were observed at all stages.

In the “manual” mode, the time for class conversion ranged from 8 to 30 minutes. In the automated mode, the main time was spent on editing the methods allocated by the program and ranged from 2 to 10 minutes.

Fig. 3 shows the averaged data of the experimental results in the form of a graph, where *totalN* – is the total number of attributes, *unnecN* – is the number of redundant attributes, *mt* – is the time of “manual” class transformation, *at* – is the time of automated class transformation.

DOI 10.15588/1607-3274-2024-2-14

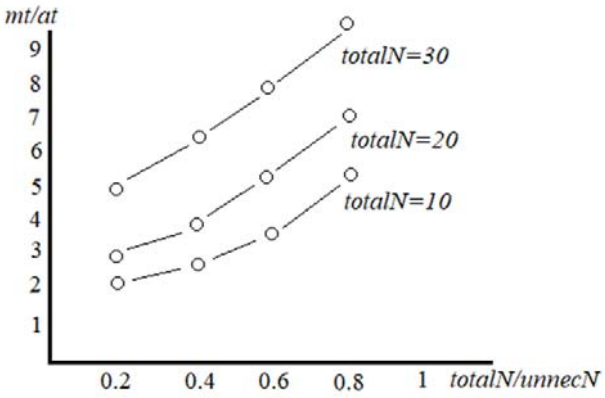


Figure 3 – Comparison of manual and automatic class transformation

6 DISCUSSION

Fig. 3 shows that the proposed transformation method proves to be effective for sufficiently large classes (10 or more attributes).

Errors during the "manual" class transformation were observed, but they were not counted because the result of the transformation was a valid code. Thus, the errors increased the transformation time.

The method leaves the programmer with the responsibility to edit functions (class methods) that use "useful" attributes along with "redundant" attributes. In general, it is extremely difficult to automate such editing, since it is determined by the tasks solved by the outer class and about which we have no information at the time of the research. Therefore, it was decided to limit to selecting such class methods and operators that use "redundant" attributes and loading them into the editor.

The proposed model and method are universal for most object-oriented programming languages with a high level of typing. However, the developed software product so far supports only Java and C++ languages.

The quality of editing class methods by the programmer is checked only for correct syntax, for which purpose command line compilers were connected to Composition Converter.

CONCLUSIONS

It is shown that the use of program classes as attributes of other classes in the implementation of composition is associated with significant problems caused by the presence of "redundant" attributes and "redundant" methods, the removal of which is a nontrivial task.

A mathematical model of the inner class is proposed, which allows us to consider a program class from the point of view of using its attributes, making it possible to formalize operations on class transformation.

A method is developed that allows automating the process of transforming an inner class, as a result of which all "redundant" attributes are removed from it, and all methods that use them are removed or edited.

Software has been created, which implements the proposed method of inner class transformation.

Approbation of the proposed solutions has shown their efficiency in the form of multiple reduction of time for class transformation (up to 10 times) in comparison with the existing technology (taking into account the time for error correction).

ACKNOWLEDGEMENTS

The work is supported by the state budget scientific research project of the Software Engineering Department, Odessa Polytechnic National University "Models, methods and tools of software engineering" (state registration number 0116U004528) and by of the Department of Technical Cybernetics and Information Technologies named after professor R.V. Merct, Odessa National Maritime University "Computer systems and information technology for the solution of applied problems" (state registration number 0123U101986).

We thank Alina I. Vitnova, who directly participated in the implementation of the class model.

REFERENCES

1. Forouzan B. A., Gilberg R. C++ Programming: An Object-Oriented Approach. McGraw-Hill Education, 2019, 960 p. https://www.booksfree.org/wp-content/uploads/2022/02/C-Programming-An-Object-Oriented-Approach-Behrouz-Forouzan.pdf
2. Lee G. Modern Programming: Object Oriented Programming and Best Practices. Packt Publishing, 2019, 266 p.
3. Kanjilal J. Composition vs. inheritance in OOP and C# [Electronic resource], InfoWord, 2023. Access mode: https://www.infoworld.com/article/3699129/composition-vs-inheritance-in-oop-and-c-sharp.html
4. Kungurtsev O., Bondar V., Gratilova K. Tranforming Classes for Composition Implementation, *Modern research in science and education: The 2nd International scientific and practical conference, Chicago, USA, 12–14 October 2023: proceedings.* Chicago, BoScience Publisher, 2023, pp. 143–148. ISBN 978-1-73981-123-5
5. Talcott C., Heinrich R., Duran F. et al. Composition of Languages, Models, and Analyses. New York, Springer, 2021, 311 p.
6. Kihlman L. Framework for Composition of Domain Specific Languages and the Effect of Composition on Re-use of Translation Rules: abstract of the dissertation … doctor of

DOI 10.15588/1607-3274-2024-2-14

philosophy in computer science. Essex, University of Essex, 2021, 69 p.
7. Pfeiffer J., Rumpe B., Schmalzing D. et al. Composition operators for modeling languages: A literature review, *Journal of Computer Languages*, 2023, Vol. 76, P. 101226
8. Zhang W., Sun Y., Oliveira B. C. Compositional Programming, *ACM Transaction on Programming Lanquages and Systems*, 2021, Vol. 43, pp. 1–61 https://doi.org/10.1145/3460228
9. Nero R. Java inheritance vs. composition: How to choose [Electronic resource], InfoWord, 2020. Access mode: https://www.infoworld.com/article/3409071/java-challenger-7-debugging-java-inheritance.html
10. Wang L., Du X., Jiang B. et al. KEADA: Identifying Key Classes in Software Systems Using Dynamic Analysis and Entropy-Based Metrics, *PubMed*, 2022,Vol. 24, № 5, P. 652 DOI: 10.3390/e24050652
11. Rashidi H., Azadi F. On Attributes of Objects in Object-Oriented Software Analysis, *International Journal of Industrial Engineering & Production Research*, 2019, Vol. 30, pp. 341–352. DOI: 10.22068/ijiepr.30.3.341
12. Kungurtsev O., Novikova N., Reshetnyak M. et al. Method for defining conceptual classes in the description of use cases, *Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments*. Vilga, 6 November 2019, proceedings, SPIE P. 11176 doi: 10.1117/12.2537070
13. Kungurtsev O. B., Vytnova A. I. Determination of inheritance relations and restructuring of software class models in the process of developing information systems, *Radio Electronics, Computer Science, Control*, 2022, № 4(63), pp. 98–107.
14. Slivnik B., Mernik M. On Parsing Programming Languages with Turing-Complete Parser, *Mathematics*, 2023, Vol. 11, Issue 7. https://doi.org/10.3390/math11071594
15. Slivnik B. Context-sensitive parsing for programming languages, *Journal of Computer Languages*, 2022, Vol. 73, P. 101172. https://doi.org/10.1016/j.cola.2022.101172
16. Sudheer N., Hrushikesava S. Different Approach Analysis for Static Code in Software Development, *International Journal of Computer Sciences and Engineering*, 2016, Vol. 4 (9), pp. 111–118.
17. Wojszczyk R., Hapka A., Królikowski T. Performance analysis of extracting object structure from source code, *27th International Conference on Knowledge Based and Intelligent Information and Engineering Sytems (KES 2023), 2023 : proceedings*, *Procedia Computer Science*, 2023, Vol. 225, pp. 4065–4073. https://doi.org/10.1016/j.procs.2023.10.402
18. Kaur S., Singh P. How does object-oriented code refactoring influence software quality? Research landscape and challenges, *Journal of Systems and Software*, 2019, Vol. 157, P. 110394. https://doi.org/10.1016/j.jss.2019.110394
19. Wedyan F., Abufakher S. Impact of design patterns on software quality: a systematic literature review, *IET Software*, 2020, Vol. 14, Issue 1, pp. 1–17. https://doi.org/10.1049/iet-sen.2018.5446

Received 22.02.2024.
Accepted 02.04.2024.

УДК 004.415

МЕТОД АВТОМАТИЗОВАНОГО ПЕРЕТВОРЕННЯ КЛАСІВ ДЛЯ РЕАЛІЗАЦІЇ КОМПОЗИЦІЇ

Кунгурцев О. Б. – канд. техн. наук, професор кафедри Інженерії програмного забезпечення Національного університету «Одеська політехніка», м. Одеса, Україна.

Бондар В. Р. – студентка кафедри Інженерії програмного забезпечення Національного університету «Одеська політехніка», м. Одеса, Україна.

Гратілова К. О. – студентка кафедри Інженерії програмного забезпечення Національного університету «Одеська політехніка», м. Одеса, Україна.

Новікова Н. О. – канд. техн. наук, доцент кафедри Технічна кібернетика й інформаційні технології ім. професора Р. В. Меркта Одеського національного морського університету, м. Одеса, Україна.

АНОТАЦІЯ

Актуальність. Використання відношення композиції – один із найефективніших і найчастіше використовуваних способів спеціалізації класів в об’єктно-орієнтованому програмуванні.

Мета роботи. Проблеми виникають при виявленні у внутрішньому класі зайвих атрибутів, які не потрібні для вирішення завдань спеціалізованого класу. Для роботи з такими атрибутами внутрішній клас має відповідні програмні методи, використання яких не тільки не вирішує завдання спеціалізованого класу, але й може призводити до помилок у його роботі. Метою роботи є видалення із внутрішнього класу «зайвих» атрибутів, і навіть всіх методів класу, які безпосередньо чи опосередковано (через інші методи) використовують ці атрибути.

Метод. Розроблено математичну модель внутрішнього класу, яка дозволила виділити «зайві» елементи класу. Запропоновано метод перетворення внутрішнього класу, який на основі аналізу коду класу надає розробнику інформацію для ухвалення рішення про «зайві» атрибути, а потім в автоматизованому режимі поетапно видаляє та перетворює елементи класу.

Результати. Для апробації запропонованих рішень розроблено програмний продукт Composition Converter. Проведено експерименти для порівняння перетворення класів у «ручному» та автоматизованому режимах. Результати показали багаторазове скорочення часу перетворення у автоматизованому режимі.

Висновки. Запропонований метод автоматизованого перетворення внутрішнього класу відповідно до завдань зовнішнього класу при реалізації композиції дозволяє суттєво скоротити час або кількість помилок при редагуванні коду внутрішнього класу. Метод може бути використаний для різних об’єктно-орієнтованих мов.

КЛЮЧОВІ СЛОВА: об’єктно-орієнтоване програмування, класи, композиція, синтаксичний аналіз, перетворення класу.

DOI 10.15588/1607-3274-2024-2-14

ЛІТЕРАТУРА

1. Forouzan B. A. C++ Programming: An Object-Oriented Approach / B. A. Forouzan, R. Gilberg. – McGraw-Hill Education, 2019. – 960 p. https://www.booksfree.org/wp-content/uploads/2022/02/C-Programming-An-Object-Oriented-Approach-Behrouz-Forouzan.pdf
2. Lee G. Modern Programming: Object Oriented Programming and Best Practices. / G. Lee. – Packt Publishing, 2019. – 266 p.
3. Kanjilal J. Composition vs. inheritance in OOP and C# [Electronic resource] / J. Kanjilal. – InfoWord, 2023. – Access mode: https://www.infoworld.com/article/3699129/composition-vs-inheritance-in-oop-and-c-sharp.html
4. Kungurtsev O. Tranforming Classes for Composition Implementation / O. Kungurtsev, V. Bondar, K. Gratilova // Modern research in science and education: The 2nd International scientific and practical conference, Chicago, USA, 12–14 October 2023: proceedings. – Chicago : BoScience Publisher, 2023. – P. 143–148. ISBN 978-1-73981-123-5
5. Composition of Languages, Models, and Analyses / [C. Talcott, R. Heinrich, F. Duran et al.]. – New York : Springer, 2021. – 311 p.
6. Kihlman L. Framework for Composition of Domain Specific Languages and the Effect of Composition on Re-use of Translation Rules: abstract of the dissertation … doctor of philosophy in computer science / L. Kihlman. – Essex: University of Essex, 2021. – 69 p.
7. Composition operators for modeling languages: A literature review / [J. Pfeiffer, B. Rumpe, D. Schmalzing et al.] // Journal of Computer Languages. – 2023. – Vol. 76. – P. 101226
8. Zhang W. Compositional Programming / W. Zhang, Y. Sun, B. C. Oliveira // ACM Transaction on Programming Lanquages and Systems. – 2021. – Vol. 43. – P. 1–61. https://doi.org/10.1145/3460228
9. Nero R. Java inheritance vs. composition: How to choose [Electronic resource] / R. Nero. – InfoWord, 2020. – Access mode: https://www.infoworld.com/article/3409071/java-challenger-7-debugging-java-inheritance.html
10. KEADA: Identifying Key Classes in Software Systems Using Dynamic Analysis and Entropy-Based Metrics / [L. Wang, X. Du, B. Jiang et al.]. – PubMed. – 2022. – Vol. 24, № 5. – P. 652. DOI: 10.3390/e24050652
11. Rashidi H. On Attributes of Objects in Object-Oriented Software Analysis / H. Rashidi, F. Azadi // International Journal of Industrial Engineering & Production Research. – 2019. – Vol. 30. – P. 341–352. DOI: 10.22068/ijiepr.30.3.341
12. Method for defining conceptual classes in the description of use cases / [O. Kungurtsev, N. Novikova, M. Reshetnyak et al.] // Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments, Vilga, 6 November 2019: proceedings. – SPIE P. 11176 doi: 10.1117/12.2537070
13. Kungurtsev O. B. Determination of inheritance relations and restructuring of software class models in the process of developing information systems / O. B. Kungurtsev, A. I. Vytnova // Radio Electronics, Computer Science, Control. – 2022. – № 4(63). – P. 98–107.
14. Slivnik B. On Parsing Programming Languages with Turing-Complete Parser / B. Slivnik, M. Mernik // Mathematics. – 2023. – Vol. 11, Issue 7. https://doi.org/10.3390/math11071594
15. Slivnik B. Context-sensitive parsing for programming languages / B. Slivnik // Journal of Computer Languages. – 2022. – Vol. 73. – P. 101172. https://doi.org/10.1016/j.cola.2022.101172
16. Sudheer N. Different Approach Analysis for Static Code in Software Development / N. Sudheer, S. Hrushikesava // International Journal of Computer Sciences and Engineering. – 2016. – Vol. 4 (9). – P. 111–118.
17. Wojszczyk R. Performance analysis of extracting object structure from source code / R. Wojszczyk, A. Hapka, T. Królikowski // 27th International Conference on Knowledge Based and Intelligent Information and Engineering Sytems (KES 2023), 2023 : proceedings. – Procedia Computer Science, 2023. – Vol. 225. – P. 4065–4073. https://doi.org/10.1016/j.procs.2023.10.402
18. Kaur S. How does object-oriented code refactoring influence software quality? Research landscape and challenges / S. Kaur, P. Singh // Journal of Systems and Software. – 2019. – Vol. 157. – P. 110394. https://doi.org/10.1016/j.jss.2019.110394
19. Wedyan F. Impact of design patterns on software quality: a systematic literature review / F. Wedyan, S. Abufakher // IET Software. – 2020. – Vol. 14, Issue 1. – P. 1–17. https://doi.org/10.1049/iet-sen.2018.5446

DOI 10.15588/1607-3274-2024-2-14

УДК 004.42

РОЗРОБКА ПЛАГІНА ДЛЯ ВІЗУАЛІЗАЦІЇ СТРУКТУРНИХ СХЕМ ОБЧИСЛЮВАЧІВ НА ОСНОВІ ТЕКСТОВОГО ОПИСУ АЛГОРИТМІВ ГАРМОНІЧНИХ ПЕРЕТВОРЕНЬ

Процько І. – д-р техн. наук, професор, кафедра автоматизованих систем управління, Національний університет «Львівська політехніка», Львів, Україна.

Теслюк В. – д-р техн. наук, професор, кафедра автоматизованих систем управління, Національний університет «Львівська політехніка», Львів, Україна.

АНОТАЦІЯ

Актуальність. У багатьох областях науки і техніки чисельне рішення задач недостатньо для подальшого розвитку реалізацій отриманих результатів. Серед існуючих підходів візуалізації інформації вибирають той, який дозволяє ефективно здійснити розкриття неструктурованих дієвих ідей, узагальнити або спростити аналіз отриманих даних. Результати візуалізації узагальнених структурних схем на основі текстового опису алгоритму наочно відображають взаємодію його частин, що важливо на системотехнічному етапі проєктування обчислювачів.

Мета дослідження – наліз та програмна реалізація візуалізації структури на прикладі обчислювачів дискретних гармонічних перетворень, отриманих в результаті синтезу алгоритму на основі циклічних згорток з можливістю розширення візуалізації структур на інші обчислювальні алгоритми.

Метод. Узагальнена схема синтезу алгоритмів швидких гармонічних перетворень у вигляді набору операцій циклічної згортки над комбінованими послідовностями вхідних даних і коефіцієнтами гармонічної функції перетворення з візуалізацією їх у вигляді узагальненої структурної схеми обчислювача.

Результати. Результатом роботи є програмна реалізація візуалізації узагальнених структурних схем для синтезованих алгоритмів косинусного та Хартлі перетворень, що наочно відображають взаємодію основних блоків обчислювача. Програмна реалізація візуалізації структури обчислювача виконана на мові TypeScript з використанням фреймворку Phaser 3.

Висновки. У роботі розглянуто та проаналізовано розроблену програмну реалізацію візуалізації загальної структури обчислювача для швидких алгоритмів дискретних гармонічних перетворень в області дійсних чисел, отриманих в результаті синтезу алгоритму на основі циклічних згорток. Результати візуалізації варіантів структурних схем обчислювачів наочно і зрозуміло відображають взаємодію його частини і дозволяють виконати оцінку того чи іншого варіанту обчислювального алгоритму в процесі проєктування.

КЛЮЧОВІ СЛОВА: комп’ютерна візуалізація, рендеринг, структурна схема, плагін візуалізації, гармонічні перетворення.

АБРЕВІАТУРИ

ISO/IEC – International Organization of Standardization/International Electrotechnical Commission;

ITU-T – International Telecommunication Union – Telecommunication sector;

UML – Unified Modeling Language;

ДКП – дискретне косинусне перетворення;

ДПХ – дискретне перетворення Хартлі;

ТМ – твірний масив;

ЦЗ – циклічні згортки.

НОМЕНКЛАТУРА

$a(n)$ – коефіцієнт зміщення;

h_{ij} – цілочисельний елемент твірного під масиву;

$H(L)$ – твірний масив;

$H_i(L_i)$ – твірний підмасив;

$Hr(n_1)$ – твірний масив рядків;

$Hc(n_2)$ – твірний масив стовпців;

k – кількість підмасивів у твірному масиві;

L – обсяг твірного масиву;

L_i – обсяг підмасиву;

N – обсяг перетворення;

T – період базисної функції;

$x(m)$ – вхідна послідовність перетворення;

$X^c(n)$ – вихідна послідовність косинусного перетворення;

$X^h(n)$ – вихідна послідовність перетворення Хартлі.

ВСТУП

Комп’ютерна візуалізація використовує конкретні процеси графічної побудови, які визначаються сферою застосувань [1]. У багатьох областях науки і техніки чисельне рішення задач недостатньо для подальшого розвитку реалізацій отриманих результатів. Для цього використовують різноманітні підходи візуалізації інформації, які дозволяють ефективно здійснити розкриття неструктурованих дієвих ідей, узагальнити або спростити аналіз отриманих даних. Сучасні можливості візуального представлення інформації дозволяють інтерпретувати дані, використовуючи процедури побудови графіків поверхонь, створення масивів даних для тривимірної графіки, вмикання і вимикання масштабної сітки, керування властивостями осей графіків та інші [2]. Ці засоби відображення результатів алгебраїчної аналітики можуть полегшити розуміння складних

DOI 10.15588/1607-3274-2024-2-15

концепцій і прийняття рішень для конкретних дослідників і розробників.

Об'єктом дослідження є процес візуалізації загальної структури обчислювача дискретних гармонічних перетворень в області дійсних чисел, отриманих в результаті синтезу алгоритму на основі ЦЗ. Описано процес отримання швидкого алгоритму у вигляді тексту, в який включає набір операцій ЦЗ над послідовностями вхідних даних та коефіцієнтами базисної функції перетворення.

Предметом дослідження є спосіб візуалізації загальних структурних схем обчислювачів ДКП та ДПХ на основі ЦЗ через рендеринг структури обчислювача з використанням конвольверів.

Метою дослідження є аналіз та програмна реалізація візуалізації узагальненої структури на прикладі обчислювачів дискретних гармонічних перетворень, отриманих в результаті синтезу алгоритмів на основі ЦЗ з можливістю розширення візуалізації структур на інші обчислювальні алгоритми.

Результати візуалізації варіантів структурних схем обчислювача для конкретного обсягу гармонічного перетворення наочно і зрозуміло відображають взаємодію його частини і дозволяють виконати їх аналіз на системотехнічному етапі проєктування.

1 ПОСТАНОВКА ПРОБЛЕМИ

За текстовим описом відповідної структури (Рис. 3) в результаті синтезу алгоритму ДКП та ДПХ на основі ЦЗ обсягу N з $x(m)$, m=0, 1,…N–1, що містить елементи SuperWrap з кількістю від 1 до N/2. SuperWrap складається з одного або більше Wrap елемента a), b),..., які представляють ЦЗ позначену (X). Кожен Wrap складається з складових: Sum Set, Summation Set. Складова Sum Set містить набір елементів Signed Number, що є знаковими цілочисельними аргументами базисної функції перетворення в межах від 0 до (N–1). Складова Summation Set складається з набору типу Summation Operand Set, який характеризується знаком («+» або «–»), а також набором $x(m)$, m=0, 1,…N–1, елементів Summation Operand.

Виконати візуалізацію загальної структурної схеми обчислювача за текстовим описом, що міститиме p-точкові конвольвери з відповідними входами, а виходи яких об'єднуються для визначення $X^c(n)$ або $X^h(n)$, n=0,1,…N–1. Користувацький інтерфейс реалізувати з вимогою мінімалізму та досягненням економії процесорного часу на промальовку, до позначень на структурній схемі включити текстове пояснення.

2 ОГЛЯД ЛІТЕРАТУРИ

Існує багато пакетів візуалізації та мов програмування і програмних середовищ для відображення у графічному вигляді розроблених обчислювальних алгоритмів [3]. Деякі з них зовсім прості: потрібно тільки завантажити дані та вибрати спосіб відображення. Інші програми більш складні і комплексні — вимагають налаштувань і відповідних знань для відображення особливостей отриманих рішень.

Наукові розробки з ефективних обчислень та сучасні системи генерації швидких алгоритмів описують їх у вигляді послідовності алгебраїчних операцій над вхідними даними до отримання кінцевих результуючих даних. Подавати інформацію так, щоб вона виділяла певні особливості взаємозв'язків серед інших в аналітичному методі є достатньо складним завданням. Для цього використовують у тому ж програмному середовищі, де реалізовано обчислювальний алгоритм, відповідні програмні засоби інфографіки з метою чіткого структурованого відображення комплексної інформації [4]. Тобто, результати синтезу ефективних обчислювальних алгоритмів потребують своєї інтерпретації не тільки у вигляді блок-схем, але і в узагальненому структурованому вигляді. Це потрібно для детальнішого аналізу та на системотехнічному етапі проєктування різноманітних інформаційних систем.

В багатьох роботах підкреслюється, що для розуміння особливостей отриманих результатів, необхідно виділити та вибрати необхідні об'єкти за допомогою візуалізації, яка наочно і зрозуміло відображає елементи та варіанти у вигляді структурних схем. Поряд з тим досліджуються питання простоти, гнучкості та масштабованості візуалізації великих структур даних [5]. В роботах [6, 7] відзначається, що краще розуміння концепції структури даних і алгоритмів полягає в посиленні основ об'єктно-орієнтованого програмування. У роботі [8] реалізовано візуалізацію різноманітних алгоритмів за допомогою модулів PyGame та Tkinter python, де крок за кроком відображається алгоритм і як різним алгоритмам потрібен різний час для виконання завдань. В багатьох роботах акцентується, що використання різноманітних інформаційних технологій забезпечує краще розуміння роботи обчислювального алгоритму та визначення його складності.

3 МАТЕРІАЛИ ТА МЕТОДИ

Для представлення інформаційних даних в спектральний образ поряд з перетворенням Фур'є використовуються алгоритми дійсного дискретного косинусного або синусного перетворення, дискретного перетворення Хартлі. Дані перетворення знаходять своє застосування в інформаційних технологіях різноманітного призначення в тому числі у згорткових нейронних мережах. Обчислення дискретних перетворень класу Фур'є являється однією з найбільш тривалих процедур в інформаційних технологіях [9]. Розроблено ряд підходів, що дозволять зменшити обчислювальну складність і, відповідно, пришвидшити роботу

DOI 10.15588/1607-3274-2024-2-15

програмного та апаратного забезпечення дискретних гармонічних перетворень. Існують ефективні алгоритми обчислення для одно, дво та багатовимірних дискретних перетворень класу Фур'є, які називають швидкими перетвореннями. Багатоваріантність ефективних обчислень розділяють на алгоритми з основою два, розщепленою основою, змішаною основою, непарного обсягу, складеного обсягу і алгоритм простих множників. Для візуального відображення цих швидких перетворень, що в більшості відповідають алгоритмам типу Кулі-Тюкі, широко використовують потоки графів з базовою операцією у вигляді так званого «метелика».

Одним з ефективних підходів є використання ЦЗ для обчислення дискретних перетворень класу Фур'є [10]. На відміну від структурних схем у вигляді потокових графів, алгоритми обчислення дискретних перетворень класу Фур'є на основі ЦЗ для свого візуального відображення потребують інші базові операції. Серед цих основних базових операцій є дії поелементного об'єднання послідовностей даних, обчислення *p*-точкових ЦЗ, результати виконання яких відповідним чином поелементно з'єднуються між собою. Для обсягів перетворень класу Фур'є, що розкладається на велику кількість простих множників, в швидких алгоритмах міститься велика кількість цих операцій. Це вимагає оптимізації їх розміщення з бажаними семантичними або візуальними кореляціями в остаточному макеті візуалізації структури обчислювача.

Організаціями ISO/IEC та ITU-T стандартизовано вісім видів дискретного косинусного перетворення, що знаходять широке застосування в процесі обробки інформаційних сигналів [11]. Широко застосовують ДПХ (1) для виконання перетворення в області дійсних значень даних в їх спектральний образ

$$X^{h}(n) = \frac{1}{N}\sum_{m=0}^{N-1} cas\left[\frac{2\pi\, n\, m}{N}\right]x(m) = \\ = \sum_{m=0}^{N-1} c(n,m)\,x(m), \quad n = 0,1,\ldots,\ N-1, \tag{1}$$

де $c(n,m)$=cas $(2\pi nm/N)$ = cos$(2\pi nm/N)$+sin$(2\pi nm/N)$.

На відміну від перетворення Фур'є, що відображає дійсні функції у комплексну область, перетворення Хартлі відображають дійсні вхідні дані у дійсний образ, використовуючи базисну функцію, що представляє собою суму косинуса і синуса одного аргументу [12].

Ці тригонометричні перетворення є подальшим розвитком дискретних перетворень Фур'є, що виконуються в дійсній області. Одним з підходів ефективного обчислення дискретних перетворень є приведення їх гармонічного базису до вигляду блочно-циклічних матричних структур з подальшим обчисленням перетворень за допомогою швидких ЦЗ [13]. В роботі [14] описується приведення гармонічного базису перетворення ДКП до набору циклічних зліва підматриць з використанням твірних масивів.

В результаті застосування підходу, структуру базисної матриці можна задати твірним масивом

$$H(L) = H_1(L_1)\,H_2(L_2)\ldots H_k(L_k) = \\ = (h_{11}, h_{12}, \ldots h_{1L1})(h_{21}, h_{22}, \ldots h_{2L2})\ldots(h_{kL1}, h_{kL2}, \ldots h_{kLk}). \tag{2}$$

Обсяг твірного масиву L дорівнює сумі обсягів підмасивів Li

$$L = (L_1 + L_2 + \ldots + L_k). \tag{3}$$

Цілочисельні елементи h_{ij} твірного підмасиву $H_i(L_i)$, які є аргументами базисної гармонічної функції перетворення (i=1,2,…,k; j=1,2,…,L_i) та $h_{ij} < T$ менші періоду повторення базисної гармонічної функції. Розглянемо синтез швидкого алгоритму для ДКП-II, яке просто називають дискретним косинусним перетворенням і вивели на основі дискретного перетворення Фур'є. Пряме ДКП-II представлено у вигляді:

$$X^{c}(n) = a(n)\sum_{m=0}^{N-1}\cos\left[\frac{n(2m+1)\pi}{2N}\right]x(m) = \\ = \sum_{m=0}^{N-1} c(n,m)\,x(m), \quad n = 0,1,\ldots,\ N-1, \tag{4}$$

де $c(n, m) = \cos(n(2m+1)\pi/2N)$.

Всі рядки $c(n,m)$ векторів $[c(k,0),\ldots,c(k,N-1)]$ є ортогональні і нормалізовані за виключенням першого. Коефіцієнт $a(n)$ зводить їх до ортонормальності

$$a(n) = \begin{cases}\sqrt{1/N}, & n = 0\\ \sqrt{2/N}, & n = 1,2,\ldots,N-1.\end{cases} \tag{5}$$

Твірний масив $H(L)$ визначається за підстановкою з рядків/стовпців аргументів гармонічної функції базисної матриці перетворення обсягу N. На основі твірного масиву здійснюється переіндексація рядків/стовпців базисної матриці, що в результаті приводить до формування блочно-циклічних матричних структур в базисній матриці ДКП. Внаслідок різних виразів індексів стовпців та рядків, що входять до аргументів базисної функції (4) для переіндексації використовуються твірний масив $Hr(n_1)$ рядків та твірний масив $Hc(n_2)$ стовпців.

Отже, одержуємо матрицю розмірності ($n_1 \times n_2$), що містить набір цілочисельних циклічних зліва підматриць різних обсягів, де розмірності $n_1 = 2N$, $n_2 = N$ визначаються обсягом твірних масивів для рядків та стовпців. Кожна з циклічних квадратних підматриць містить цілочисельні елементи, які

DOI 10.15588/1607-3274-2024-2-15

належать одному з твірних підмасивів $H_i(L_i)$. Обсяги L_i кожного з твірних підмасивів залежать від множників розкладу обсягу перетворення N, і в сумі відповідають умові (3).

Для дослідження структури базисних матриць ДКП розроблено універсальний програмний засіб аналізу цілочисельних матриць, який виконує сканування всього набору елементів блочно-циклічної матриці. Для пошуку заданого фрагменту за максимальною шириною і висотою L_i, в матриці виконується покрокове сканування з переміщенням згори вниз і зліва направо.

Для синтезу швидкого алгоритму ДКП необхідно виконати аналіз структури одержаної блочно-циклічної матриці з метою визначення ідентичних блоків, що розміщених горизонтально та вертикально один відносно іншого. Наявність ідентичних блоків приводить до зменшення обчислювальної складності та надає можливість розпаралелення виконання обчислення ДКП. В процесі автоматичного синтезу алгоритмів обчислення ДКП довільного обсягу N на основі ЦЗ на завершальному етапі аналізу блочно-циклічних матричних структур базисної матриці ДКП виконується визначення інформаційних даних про кількість ідентичних циклічних підмасивів та їх розташування в базисній матриці.

Ідентичні циклічні підматриці, що розміщені вертикально один відносно іншого приводять до одноразового обчислення циклічних згорток, результати яких в процесі об'єднання використовуються для визначення вихідних значень перетворення $X^c(m)$.

Ідентичні циклічні підматриці, що розміщені горизонтально один відносно іншого приводять до об'єднання груп вхідних значень $x(n)$ перетворення і одноразового обчислення ЦЗ. Результати ЦЗ в процесі об'єднання використовуються для однієї групи вихідних значень перетворення $X^c(m)$. Виконання поелементних додавань вхідних значень $x(n)$ будуть використовуватись для однотипових циклічних підматриць, розміщених по горизонталі.

Вихідні значення $X^c(m)$ перетворення ДКП отримують об'єднанням результатів згорток по горизонталі на основі відповідних координат.

Приклад виконання синтезу швидкого алгоритму ДКП для обсягу N=15 з твірним масивом $H_r(n_1)$ рядків та твірним масивом $H_c(n_2)$ стовпців мають вигляд

$H_r(30)$ = (0) (1 7 11 17) (29 23 19 13) (2 14 22 26) (28 16 8 4) (3 21 27 9) (5 25) (6 18) (24 12) (10) (20);
$H_c(15)$ = (0) (1 7 11 13) (14 8 4 2) (3 9) (12 6) (5) (10).

Результат (рис. 1) формування блочно-циклічних матричних структур аргументів базисної матриці ДКП для обсягу N=15 з твірними масивами $H_r(30)$ та $H_c(15)$.

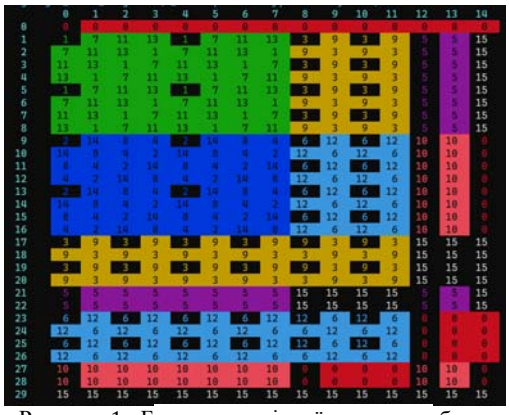


Рисунок 1– Блочно-циклічної структури базисної матриці ДКП для обсягу N=15

В результаті аналізу блочно-циклічної структури базисної матриці формується текстовий файл про кількість ідентичних циклічних підмасивів в базисній матриці (рис. 1). Ці дані дозволяють визначити кількість та обсяг ЦЗ, можливість їх паралельного обчислення та об'єднання обчислених значень ЦЗ для визначення вихідних значень ДКП. Текстовий файл ДКП-II для обсягу N=15 має вигляд

1. a) (+0) (X) { +x(0) +x(3) +x(5) +x(8) +x(14) +x(11) +x(9) +x(6) +x(1) +x(10) +x(13) +x(4) +x(2) +x(12) +x(7) }

2. a) (+1 +7 +11 –13) (X) { +(x(0), x(3), x(5), x(8)) –(x(14), x(11), x(9), x(6)) }
b) (+3 –9) (X) { +(x(1), x(10)) –(x(13), x(4)) }
c) (+5) (X) { +x(2) –x(12) }

3. a) (+2 +14 –8 –4) (X) { +(x(0), x(3), x(5), x(8)) +(x(14), x(11), x(9), x(6)) }
b) (+6 –12) (X) { +(x(1), x(10)) +(x(13), x(4)) }
c) (+10) (X) { +x(2) +x(12) }
d) (–0) (X) { –x(7) }

4. a) (+3 –9) (X) { +(x(0), x(3)) –(x(5), x(8)) –(x(14), x(11)) +(x(9), x(6)) }
b) (+9 +3) (X) { +(x(1), x(10)) –(x(13), x(4)) }

5. a) (+5) (X) { +x(0) –x(3) +x(5) –x(8) –x(14) +x(11) –x(9) +x(6) –x(2) +x(12) }

6. a) (+6 –12) (X) { +(x(0), x(3)) +(x(5), x(8)) +(x(14), x(11)) +(x(9), x(6)) }
b) (–12 +6) (X) { –(x(1), x(10)) –(x(13), x(4)) }
c) (–0) (X) { –x(2) –x(12) –x(7) }

7. a) (+10) (X) { +x(0) +x(3) +x(5) +x(8) +x(14) +x(11) +x(9) +x(6) +x(2) +x(12) }
b) (–0) (X) { –x(1) –x(10) –x(13) –x(4) –x(7) }

Обсяг та кількість ЦЗ для ДКП з N=15 дорівнює: 1– точкова згортка: 8; 2– точкова згортка: 6; 4– точкова згортка: 2.

В роботі [15] описується приведення гармонічного базису перетворення ДПХ до набору циклічних зліва

DOI 10.15588/1607-3274-2024-2-15

підматриць з використанням твірних масивів. Розглянемо приклад виконання синтезу швидкого алгоритму ДКП для обсягу N=15 з твірним масивом $H(N)$, що має вигляд

$$H(15) = (0)\ (1\ \ 2\ \ 4\ \ 8)\ (14\ \ 13\ \ 11\ \ 7)\ (3\ \ 6\ \ 12\ \ 9)\ (5\ \ 10).$$

Результат (рис. 2) формування блочно-циклічних матричних структур аргументів в базисній матриці ДПХ для обсягу N=15 з твірними масивами $H(15)$

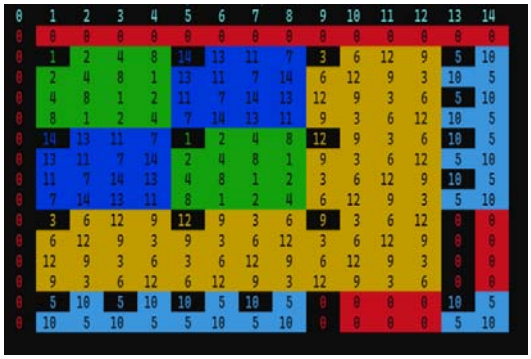


Рисунок 2 – Блочно-циклічної структури базисної матриці ДПХ для обсягу N=15

В результаті аналізу блочно-циклічної структури базисної матриці формується текстовий файл про кількість ідентичних циклічних підмасивів в базисній матриці (рис 2). Текстовий файл ДПХ для обсягу N=15 має вигляд

1. a) (+0) (X) { +x(0) +x(1) +x(2) +x(4) +x(8) +x(14) +x(13) +x(11) +x(7) +x(3) +x(6) +x(12) +x(9) +x(5) +x(10) }

2. a) (+0) (X) { +x(0) }
   b) (+1 +2 +4 –8) (X) { +(x(1), x(2), x(4), x(8)) }
   c) (+14 –13 –11 –7) (X) { +(x(14), x(13), x(11), x(7)) }
   d) (+3 –6 –12 –9) (X){+(x(3), x(6), x(12), x(9)) }
   e) (+5 –10) (X) { +(x(5), x(10)) }

3. a) (+0) (X) { +x(0) }
   b) (+14 –13 –11 –7) (X) { +(x(1), x(2), x(4), x(8)) }
   c) (+1 +2 +4 –8) (X) { +(x(14), x(13), x(11), x(7)) }
   d) (–12 –9 +3 –6) (X) {–(x(3), x(6), x(12), x(9)) }
   e) (–10 +5) (X) { –(x(5), x(10)) }

4. a) (+0) (X) { +x(0) +x(5) +x(10) }
   b) (+3 –6 –12 –9) (X) { +(x(1), x(2), x(4), x(8)) }
   c) (–12 –9 +3 –6) (X) { –(x(14), x(13), x(11), x(7)) }
   d) (–9 +3 –6 –12) (X) {–(x(3), x(6), x(12), x(9)) }

5. a) (+0) (X) { +x(0) +x(3) +x(6) +x(12) +x(9) }
   b) (+5 –10) (X) { +(x(1), x(2)) +(x(4), x(8)) }
   c) (–10 +5) (X) { –(x(14), x(13)) –(x(11), x(7)) –(x(5), x(10)) }

Обсяг та кількість ЦЗ для ДПХ з N=15 дорівнює: 1– точкова згортка: 4; 2– точкова згортка: 4; 4– точкова згортка: 9.

4 ЕКСПЕРИМЕНТИ

Для програмної реалізації візуалізації структури обчислювача вибрано мову програмування TypeScript. Мова TypeScript розширює свою попередницю JavaScript, забезпечуючи зворотну сумісність. Мова TypeScript є статично типізованою, що дасть можливість виявити помилки ще на етапі написання коду або компіляції. Для роботи з TypeScript (яка є розробкою Microsoft) найбільше підходять середовища розробки компанії Microsoft: Visual Studio та Visual Studio Code. Середовище Visual Studio є більш «важким» рішенням, адже працює повільніше, проте й має більше функціоналу. Середовище ж Visual Studio Code створено спеціально для роботи з веб-додатками і є надзвичайно простим, розширюваним з великою кількістю плагінів.

Для роботи з версіями програми використовується популярна і сильно розровсюджена система Git. Через свою розповсюдженість та простоту для керування пакетами використовуватиметься *npm*. Для пакування модулів використано Webpack, як стандарт в сучасній індустрії веб-додатків. Використання *npm* та Webpack передбачає використання Node.js. Для роботи з файлами (зчитування та запис) використовуватиметься бібліотека «file-saver». Для розроблення модульних тестів (unit-тестів) використовуватиметься набір бібліотек: chai, mocha, sinon.

Вибір інструментів для візуалізації. Найбільш популярними рішеннями для забезпечення візуалізації є: Pixi.js, Phaser 2, Phaser 3. Перший інструмент є дуже швидким, проте має лише базові функції, тому розроблення буде дещо повільнішим, ніж в інших варіантах. Оскільки швидкість виконання не є найважливішим атрибутом якості системи візуалізації, що розробляється, то розглянемо інші варіанти. Phaser 2 – це популярне рішення, що має велику кількість функцій, і з яким легко працювати. Проте даний фреймворк є дещо застарілим і не підтримується розробниками. Phaser 3 – фреймворк, що прийшов на заміну свого попередника Phaser 2, і є повним переосмисленням його. Дане рішення стало новим стандартом в індустрії і активно розвивається. Тому інструментом для візуалізації вибрано Phaser 3.

Початковими даними для роботи системи візуалізації є текстовий файл алгоритму у результаті аналізу блочно-циклічної структури базисної матриці. Текстовий файл формується за відповідною структурою представлення даних швидкого алгоритму обчислення. На рис. 3 зображено схему, яка на прикладі демонструє те, на які складові структуруються дані текстового опису алгоритму.

DOI 10.15588/1607-3274-2024-2-15

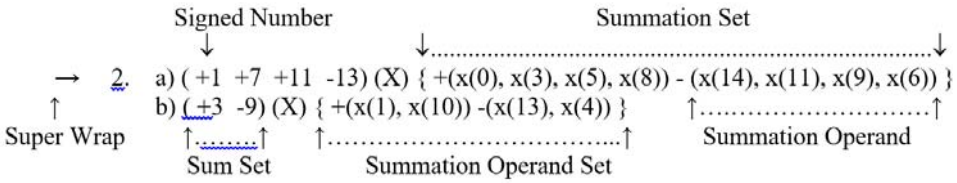


Рисунок 3 – Шаблон структур даних

Позначення (X) означає виконання операції, у даному випадку ЦЗ над послідовностями базисних функцій з аргументами поданими на початку у круглих дужках ( ) та послідовностями вхідних даних перетворення x(*i*), *i*=0,1,...*N*–1, які необхідно попередньо поелементно об'єднати.

Головним блоком (рис. 3) є елемент Super Wrap, який відповідає за одну згортку. Він складається з одного або більше елемента a), b), ... Wrap, які представляють ЦЗ. Кожен Super Wrap складається з складових: Sum Set, Summation Set. Складова Sum Set містить набір елементів Signed Number. Складова Summation Set складається з набору типу Summation Operand Set, який характеризується знаком («+» або «–»), а також набором елементів Summation Operand.

Програма візуалізації структури обчислювача використовує модулі: токенізер, парсер, оптимізатор, візуалізатор.

Токенізатор опрацьовує «сирі» символьні дані з текстового файлу опису і розбиває їх на послідовність токенів. В алгоритмі загальної логіки модуля-токенізатора виконуються повторювані дії доки не буде досягнуто кінця даних. Серед дій, що виконуються є зчитування символу і, в залежності від його типу, виконується відповідна підпроцедура для зчитування токена (токени можуть складатись з кількох символів). Крім того, виконується перевірка, чи підтримується зчитаний символ. Отриманий набір токенів передається на синтаксичний аналіз доки не буде досягнуто кінця даних.

Перед синтаксичним аналізом окремої згортки поділяється вхідний набір токенів на групи, які формують певну ієрархію. Потреба в цьому алгоритмі спричинена тим, що формат вхідних даних передбачає ієрархічність, де на першому рівні є чисельні індекси («1.», «2.» тощо), а на другому – символьні («а)», «б)» тощо). Даний алгоритм розділяє токени спершу на чисельно-індексовані групи, а потім кожну групу поділяє на символьно-індексовані підгрупи. Потім ці групи та підгрупи використовуютьсяв модулі синтаксичного аналізатора.

Синтаксичний аналізатор отримує токени і перетворює їх на синтаксичне дерево. Цей процес деколи називають парсингом. Тобто з текстового файлу опису згорток в алгоритмі синтаксичного аналізу зчитується кожен рядок і використовуються відповідні граматичні правила для аналізу лише однієї згортки. Далі виконується зчитування першої послідовності згортки, роздільника (X), другої послідовності. Далі перевіряється, чи залишились дані. Якщо так, то формується синтаксичне дерево, інакше – отримуємо помилку.

Наступним модулем є оптимізатор, завданням якого є, дослідивши структуру отриманого дерева, зробити певні перетворення, які б сформували дерево більш компактним та зручним для візуалізації. Оптимізація на даний момент є реалізована лише базово і буде допрацьовуватись в наступних версіях програмної системи.

Модуль візуалізатора здійснює інфографіку структурної схеми ДКП. Для цього виконується проста лінійна послідовність кроків:

– виконується підрахунок параметрів, що допоможуть виконати візуалізацію (він здійснюється на основі синтаксичного дерева);

– відображаються прямокутники та підписи, що їх стосуються;

– відображаються різноманітні стрілки та їх підписи.

Таким чином, оптимізоване синтаксичне дерево, а також налаштування, отримані з користувацького інтерфейсу, надходять у візуалізатор. Він створює візуальне зображення структури обчислювача.

На рис. 4 зображено UML-діаграму класів системи для візуального відображення структурної схеми обчислювача. UML-діаграму класів формує уявлення про те, якою є статична структура програмного забезпечення системи.

Варто також зазначити, що на даній схемі відображено лише основні ключові класи, методи та поля. Насправді система міститиме більше дрібних об'єктів, серед яких, зокрема, перелічення, інтерфейси, деякі не надто важливі допоміжні класи для синтаксичного аналізу, конфігурації, утилітарні класи тощо.

DOI 10.15588/1607-3274-2024-2-15

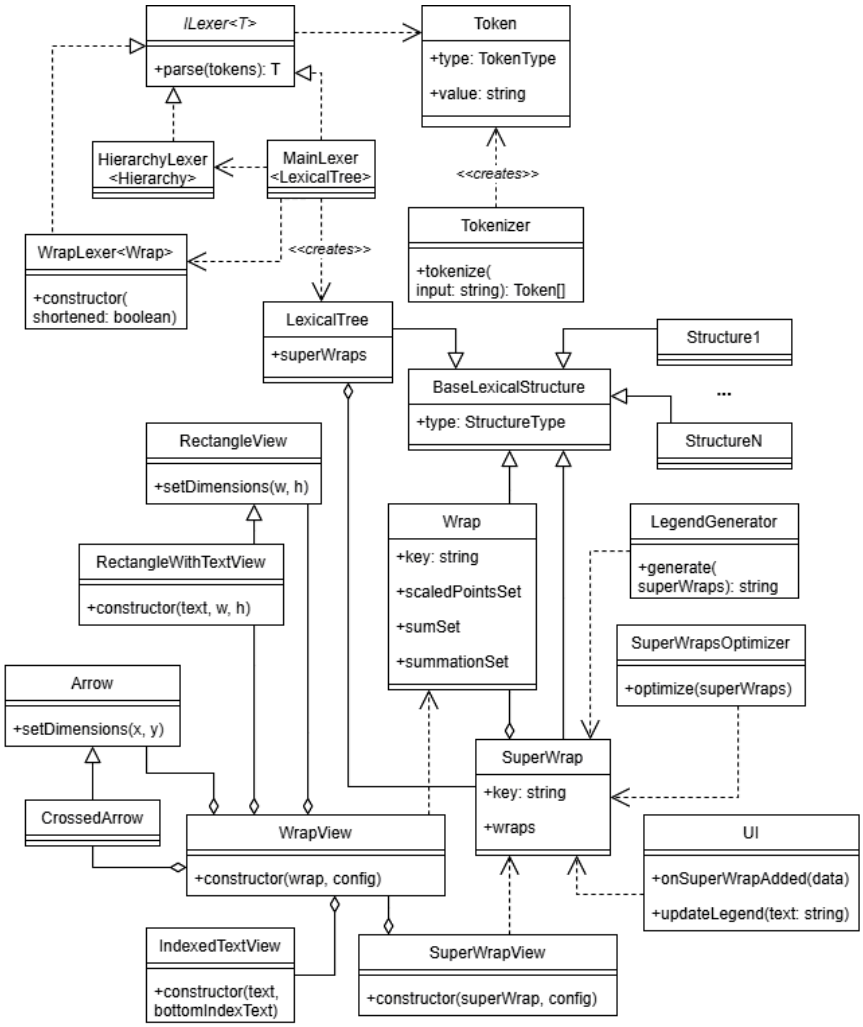


Рисунок 4 – Діаграма класів системи візуалізації

5 РЕЗУЛЬТАТИ

Взаємодія з системою здійснюється через користувацький інтерфейс, у якому проводяться налаштування для відображення структурної схеми (Рис. 5). В результаті отримуємо відображення системи на екрані, а також можливість завантажити зображення в файли, що міститимуть структурну схему обчислювача або частину.

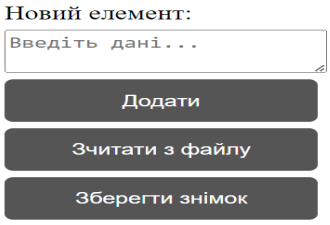


Рисунок 5 – Користувацький інтерфейс для візуалізації структури обчислювача

На рис. 6 бачимо приклад того, як виглядає відображення структури обчислювача ДКП на основі текстового файлу ДКП-II для обсягу N=15. Структура складається з набору стрілок (вхідних, проміжних, вихідних), підписів до них (як от x'_{2a}), та прямокутників (CC та U). CC є блоками p-точкових ЦЗ, U є блоками поелементного об'єднання вхідних даних $x(i)$, $X(i)$ вихідні дані перетворення, i=0,1,...,N–1.

Дане зображення передається на користувацький інтерфейс, який в свою чергу, відображає все на екран для користувача, а також дає можливість зберегти отримане зображення у файл. Крім цього, можна побачити текстове пояснення до позначень на схемі

$x'_{2a}=x'_{3a}=(x(0), x(3), x(5), x(8))$
$x''_{2a}=x''_{3a}=(x(14), x(11), x(9), x(6))$
$x'_{2b}=x'_{3b}=x'_{4b}=x'_{6b}=(x(1), x(10))$
$x''_{2b}=x''_{3b}=x''_{4b}=x''_{6b}=(x(13), x(4))$
$x'_{2c}=x(2)$
$x''_{2c}=x(12)$
$x'_{3c}=(x(2), x(12))$
$x'_{3d}=x(7)$
$x'_{4a}=x'_{6a}=(x(0), x(3), x(14), x(11))$
$x''_{4a}=x''_{6a}=(x(5), x(8), x(9), x(6))$
$x'_{5a}=(x(0), x(5), x(14), x(9), x(2))$
$x''_{5a}=(x(3), x(8), x(11), x(6), x(12))$
$x'_{6c}=(x(2), x(12), x(7))$
$x'_{7a}=(x(0), x(3), x(5), x(8), x(14), x(11), x(9), x(6), x(2), x(12))$
$x'_{7b}=(x(1), x(10), x(13), x(4), x(7))$

DOI 10.15588/1607-3274-2024-2-15

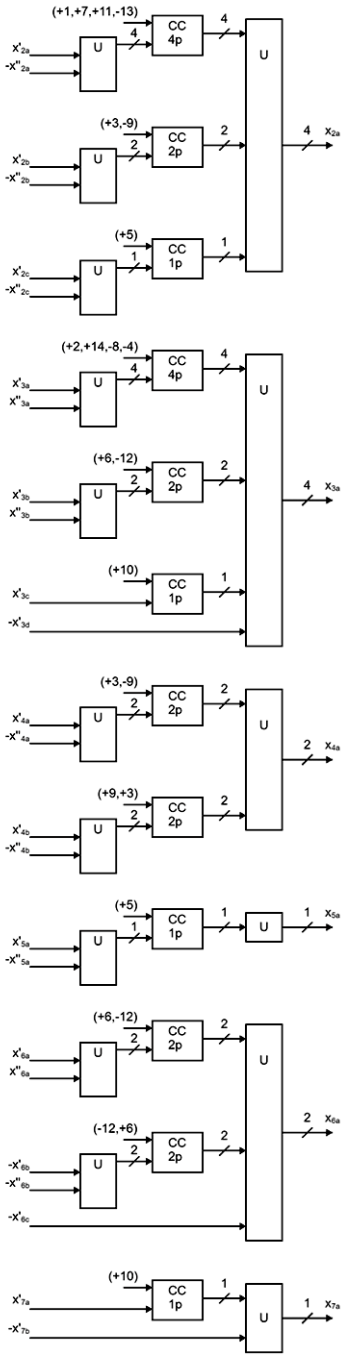


Рисунок 6 – Візуальне зображення структури обчислювача ДКП обсягу N=15

На рис. 7 бачимо результат того, як виглядає частина відображення структури обчислювача ДПХ обсягу N=15 на основі тексту

5. a) (+0) (X) { +x(0) +x(3) +x(6) +x(12) +x(9) }
b) (+5 –10) (X) { +(x(1), x(2)) +(x(4), x(8)) }
c) (–10 +5) (X) { –(x(14), x(13)) –(x(11), x(7)) –(x(5), x(10)) }

з текстового файлу ДПХ для обсягу N=15.

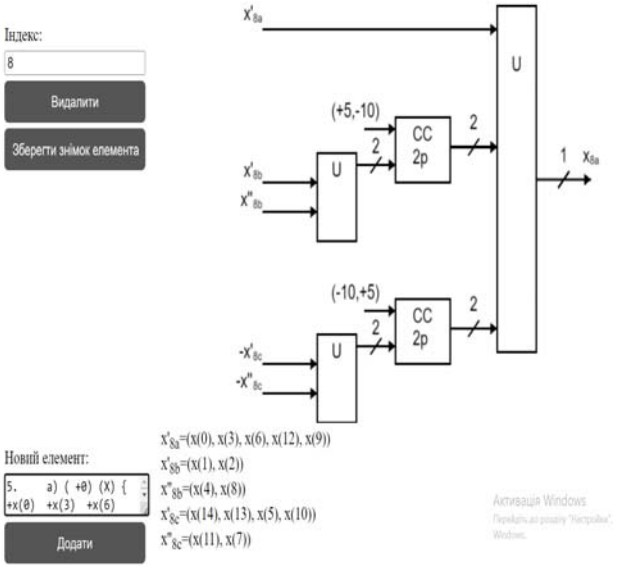


Рисунок 7 – Візуальне зображення частини структури обчислювача ДПХ обсягу N=15

Модуль генерації текстового пояснення (рис. 7) так само, як і візуалізатор, використовує оптимізоване синтаксичне дерево, та передає результати своєї роботи на вхід до користувацького інтерфейсу.

6 ОБГОВОРЕННЯ

Програма синтезу алгоритмів швидких гармонічних перетворень на основі ЦЗ завдяки отриманому текстовому опису алгоритму включає розроблений плагін для візуалізації структурних схем обчислювачів. Це дає можливість швидко оцінити кількість конвольверів їх взаємозв'язок, обсяги виконання ЦЗ, послідовності об'єднання вхідних даних на системотехнічному етапі проєктування обчислювача. Реалізація обчислювачів швидких гармонічних перетворень на основі ЦЗ у вигляді інтегральних схем має ряд переваг, що характеризуються високим показником параметрів площа затримка [16].

В плагіні застосовано технології Web з метою досягнення простоти розроблення та підтримки, кросплатформності. Використано провідну технологію Phaser 3 для реалізації візуалізації векторних примітивів. Для того, щоб система не мала таку шкідливу властивість, притаманну деяким програмним рішенням, як «сильну зв'язність», вона чітко розділена на дві частини. Ядро системи, частина *core*,, відповідає за логіку програмного застосунку і виконує:

– токенізацію;
– синтаксичний аналіз;
– оптимізацію.

Частина *view*, що має відношення до відображення будь-чого на екрані, включає:

– графічний користувацький інтерфейс;
– візуалізаційний модуль для згорток.

Оскільки система є насиченою алгоритмами, то легко можна допустити помилки, які буде важко

DOI 10.15588/1607-3274-2024-2-15

виправити, коли розробка досягне великого масштабу. Для того щоб «відловити» та виправити помилки, було прийнято важливе рішення – під час розроблення системи розробляти також й модульні тести (unit tests). Загалом написано 12 тестів, які перевіряють коректність 5 класів, які стосуються найважливішої частини логіки системи – токенізація та синтаксичний аналіз.

В ході розроблення, неодноразово було помічено, що результати виконання тестів свідчать про наявні помилки у виконанні програми, і, таким чином, було легко вирішено всі проблеми, які виникали через випадково допущені помилки в коді.

Система має мінімально необхідний набір функцій, тому може розширюватись, зокрема шляхом покращення оптимізації лексичного дерева. Також, розроблену програмну візуалізацію можна покращити для більш інтерактивного та більш гнучкого налаштування на візуалізацію структур різноманітних обчислювачів.

ВИСНОВКИ

У роботі розглянуто візуалізацію загальної структури обчислювача, отриманого в результаті синтезу алгоритмів швидких гармонічних перетворень на основі ЦЗ.

Наукова новизна роботи полягає у визначенні особливостей структур обчислювачів на основі загального відображення через конвольвери, які візуалізують опис синтезованих блочно-циклічних базисних матриць дискретних гармонічних перетворень. На основі створеного опису алгоритму, що включає багатоярусний набір ЦЗ, розроблена програма візуалізації структури обчислювача дискретних гармонічних перетворень. Програма візуалізації реалізована на мові програмування TypeScript з використанням фреймворка Phaser 3 для візуалізації векторних примітивів. Результати візуалізації структурних схем обчислювача ДКП і ДПХ для конкретного обсягу перетворення наочно і зрозуміло відображають взаємодію його частини, що важливо на системотехнічному етапі проєктування обчислювача.

Для розроблення застосовано технології Web, досягнувши таким чином простоту розроблення, підтримки, кросплатформність. Користувацький інтерфейс реалізовано з вимогою мінімалізму з досягненням економії процесорного час на промальовку. Дизайн є контрастним та з приглушеними спокійними кольорами: білий, чорний, відтінки сірого. Кросплатформність надає можливість працювати з системою без встановлення додаткового програмного забезпечення.

Практичне значення роботи полягає у тому, що розроблений плагін для візуалізації загальних структурних схем обчислювачів ДКП і ДПХ може використовуватись для відображення загальних структур і інших обчислювальних алгоритмів, який дозволить наочно і зрозуміло відобразити взаємодію його частини та виконати їх аналіз на системотехнічному етапі проєктування.

Напрям подальших досліджень полягатиме в розробці алгоритмічного та програмного забезпечення що забезпечить аналіз та вибір обчислювальних структур за відповідними критеріями на основі різноманітних варіантів синтезованих швидких алгоритмів обчислення дискретних гармонічних перетворень певного типу та обсягу.

ПОДЯКИ

Робота виконана в рамках держбюджетної науково-дослідної роботи «Методи та засоби інтелектуального вимірювання параметрів руху та визначення просторової орієнтації наземних мобільних робототехнічних платформ» національного університету «Львівська політехніка».

ЛІТЕРАТУРА

1. Дрєєв О. М. Аналіз комп'ютерних систем візуалізації з метою алгоритмізації обґрунтування щодо їх використання / О. М. Дрєєв, Б. Ю. Железняк // Конструювання, виробництво та експлуатація сільськогосподарських машин. – 2021. – Вип. 51. – С. 227–238. DOI: 10.32515/2414-3820.2021.51.227-238
2. Xue M. Research on Information Visualization Graphic Design Teaching Based on DBN Algorithm / M. Xue // Computational Intelligence and Neuroscience. – 2021. – Vol. 6. – P. 1–10. DOI: 10.1155/2021/3355030
3. 36 кращих інструментів для візуалізації даних. – Access mode: https://toplead.com.ua/ua/blog/id/38-luchshih-instrumentov-dlja-vizualizacii-dannyh-160/
4. Performance evaluation and analysis of sparse matrix and graph kernels on heterogeneous processors / [F. Zhang, W. Liu, N. Feng et al.] // CCF Transactions on High Performance Computing. – 2019. – Vol. 1. – P. 131–143
5. Garriga J. Towards a comprehensive visualization of structure in data / J. Garriga, F. Bartumeus // arXiv. – 2021. – Access mode: https://arxiv.org/abs/2111.15506?context=cs
6. Simonak S. Increasing the Engagement Level in Algorithms and Data Structures Course by Driving Algorithm Visualizations / S. Simonak // Informatica. – September 2020. – Vol. 44, Issue 3, DOI: 10.31449/inf.v44i3.2864
7. Ghadge S. A Survey paper on data structure and algorithm visualization / S. Ghadge, V. Mane // International Research Journal of Modernization in Engineering Technology and Science. – April-2022. – Vol. 04, Issue 04. – P. 232–236
8. Gupta A. S. Algorithm Visualization / A. S. Gupta, M. Vyawahare, A. Viz // 5th Biennial International Conference on Nascent Technologies in Engineering (ICNTE). – Navi Mumbai, India, 20–21 January 2023. – P. 1–5. DOI: 10.1109/ICNTE56631.2023.10146719
9. Oppenheimer A. V. Discrete-Time Signal Processing. Third edition. Englewood Cliffs / A. V. Oppenheimer, R. W. Schafer. – NJ : Prentice Hall. Pearson Education Limited, 2014. –1042 p.
10. Blahut R. E. Fast algorithms for signal processing / R. E. Blahut. – Cambridge : University Press, 2010. – 469 p. DOI: 10.1017/CBO9780511760921
11. MPEG-4, MPEG-4: ISO/IEC 14496-2:2004. Information technology – Coding of audio-visual objects – Part 2: Visual, ISO, 2004.
12. Dziech A. New Orthogonal Transforms for Signal and Image Processing / A. Dziech // Applied Sciences. – 2021. – Vol. 11, Issue 16. – P. 7433. DOI: 10.3390/app11167433

DOI 10.15588/1607-3274-2024-2-15

13. Prots'ko I. Block-Cyclic Structuring of the Basis of Fourier Transforms Based on Cyclic Substitution / I. Prots'ko, M. Mishchuk // Cybernetics and Systems Analysis. – 2021. – Vol. 57, Issue 6. – P. 1008–1016.
14. Prots'ko I. Algorithm of Efficient Computation of DCT I–IV Using Cyclic Convolutions / I. Prots'ko // International Journal of Circuits, Systems and Signal Processing. – 2013. – Vol. 7, Issue 1. – P. 1–9.
15. Prots'ko I. Algorithm of efficient computation of generalized discrete Hartley transform based on cyclic convolutions / I. Prots'ko // IET Signal Processing. – 2014. – Vol. 8, Issue 4. – P. 301–308.
16. Chiper D. F. An Efficient Algorithm and Architecture for the VLSI Implementation of Integer DCT That Allows an Efficient Incorporation of the Hardware Security with a Low Overhead / D. F. Chiper, A. Cracan // Applied Sciences. – 2023, Vol. 13, Issue 12, 6927. DOI: 10.3390/app13126927

Стаття надійшла до редакції 09.02.2024.
Після доробки 25.04.2024.

UDC 004.42

DEVELOPMENT OF A PLUG-IN FOR VIZUALIZATION OG STRUCTURAL SCHEMES OF COMPUTERS BASED ON THE TEXTUAL DESCRIPTION OF ALGORITHMS OF HARMONIC TRANSFORMS

Prots'ko I. – Dr. Sc., Professor, Department of Automated Control Systems, Lviv National Polytechnic University, Lviv, Ukraine.
Teslyuk V. – Dr. Sc., Professor, Department of Automated Control Systems, Lviv Polytechnic National University, Lviv, Ukraine.

ABSTRACT

Context. In many areas of science and technology, the numerical solution of problems is not enough for the further development of the implementation of the obtained results. Among the existing information visualization approaches, the one that allows you to effectively reveal unstructured actionable ideas, generalize or simplify the analysis of the received data is chosen. The results of visualization of generalized structural diagrams based on the textual description of the algorithm clearly reflect the interaction of its parts, which is important at the system engineering stage of computer design.

Objective of the study is the analysis and software implementation of structure visualization using the example of discrete harmonic transformation calculators obtained as a result of the synthesis of an algorithm based on cyclic convolutions with the possibility of extending the structure visualization to other computational algorithms.

Method. The generalized scheme of the synthesis of algorithms of fast harmonic transformations in the form of a set of cyclic convolution operations on the combined sequences of input data and the coefficients of the harmonic transformation function with their visualization in the form of a generalized structural diagram of the calculator.

The results. The result of the work is a software implementation of the visualization of generalized structural diagrams for the synthesized algorithms of cosine and Hartley transformations, which visually reflect the interaction of the main blocks of the computer. The software implementation of computer structure visualization is made in TypeScript using the Phaser 3 framework.

Conclusions. The work considers and analyzes the developed software implementation of visualization of the general structure of the calculator for fast algorithms of discrete harmonic transformations in the domain of real numbers, obtained as a result of the synthesis of the algorithm based on cyclic convolutions. The results of visualization of variants of structural schemes of computers clearly and clearly reflect the interaction of its parts and allow to evaluate one or another variant of the computing algorithm in the design process.

KEYWORDS: computer visualization, rendering, structural scheme, visualization plugin, harmonic transforms.

REFERENCES

1. Drieiev O., Zhelesnya B. Analysis of Computer Visualization Systems in Order to Algorithmize the Rationale for their Use, *Design, production and Exploitation of Agricultural Machines*, 2021, Vol. 51, pp. 227–238. DOI: 10.32515/2414-3820.2021.51.227-238
2. Xue M. Research on Information Visualization Graphic Design Teaching Based on DBN Algorithm, *Computational Intelligence and Neuroscience*, 2021, Vol. 6, pp. 1–10. DOI: 10.1155/2021/3355030
3. 36 best data visualization tools. – Access mode: https://toplead.com.ua/ua/blog/id/38-luchshih-instrumentov-dlja-vizualizacii-dannyh-160/
4. Zhang F., Liu W., Feng N., Zhai J., Du X. Performance evaluation and analysis of sparse matrix and graph kernels on heterogeneous processors, *CCF Transactions on High Performance Computing*, 2019, Vol. 1, pp. 131–143
5. Garriga J., Bartumeus F. Towards a comprehensive visualization of structure in data, *arXiv*, 2021, Access mode: https://arxiv.org/abs/2111.15506?context=cs
6. Simonak S. Increasing the Engagement Level in Algorithms and Data Structures Course by Driving Algorithm Visualizations, *Informatica*, September 2020, Vol. 44, Issue 3, DOI: 10.31449/inf.v44i3.2864
7. Ghadge S., Mane V. A Survey paper on data structure and algorithm visualization, *International Research Journal of Modernization in Engineering Technology and Science*, April-2022, Vol. 04, Issue 04, pp. 232–236
8. Gupta A. S., Vyawahare M., Viz A. Algorithm Visualization, *5th Biennial International Conference on Nascent Technologies in Engineering (ICNTE)*. Navi Mumbai, India, 20–21 January 2023, pp. 1–5. DOI: 10.1109/ICNTE56631.2023.10146719
9. Oppenheimer A. V. Schafer R. W. Discrete-Time Signal Processing. Third edition. Englewood Cliffs. NJ, Prentice Hall, Pearson Education Limited, 2014, 1042 p.
10. Blahut R. E. Fast algorithms for signal processing. Cambridge, University Press, 2010, 469 p. DOI: 10.1017/CBO9780511760921
11. MPEG-4, MPEG-4: ISO/IEC 14496-2:2004. Information technology, Coding of audio-visual objects, Part 2, Visual, ISO, 2004.
12. Dziech A. New Orthogonal Transforms for Signal and Image Processing, *Applied Sciences*, 2021, Vol. 11, Issue 16, P. 7433. DOI: 10.3390/app11167433
13. Prots'ko I., Mishchuk M. Block-Cyclic Structuring of the Basis of Fourier Transforms Based on Cyclic Substitution, *Cybernetics and Systems Analysis*, 2021, Vol. 57, Issue 6, pp. 1008–1016.
14. Prots'ko I. Algorithm of Efficient Computation of DCT I–IV Using Cyclic Convolutions, *International Journal of Circuits, Systems and Signal Processing*, 2013, Vol. 7, Issue 1, pp. 1–9.
15. Prots'ko I. Algorithm of efficient computation of generalized discrete Hartley transform based on cyclic convolutions, *IET Signal Processing*, 2014, Vol. 8, Issue 4, pp. 301–308.
16. Chiper D. F., Cracan A. An Efficient Algorithm and Architecture for the VLSI Implementation of Integer DCT That Allows an Efficient Incorporation of the Hardware Security with a Low Overhead, *Applied Sciences*, 2023, Vol. 13, Issue 12, P. 6927. DOI: 10.3390/app13126927

DOI 10.15588/1607-3274-2024-2-15

p-ISSN 1607-3274 Радіоелектроніка, інформатика, управління. 2024. № 2
e-ISSN 2313-688X Radio Electronics, Computer Science, Control. 2024. № 2

UDC 004.412:519.237.5

DETERMINING OBJECT-ORIENTED DESIGN COMPLEXITY DUE TO THE IDENTIFICATION OF CLASSES OF OPEN-SOURCE WEB APPLICATIONS CREATED USING PHP FRAMEWORKS

Prykhodko A. S. – Post-graduate student of the Department of Mathematical Support of Computer Systems, Odesa I. I. Mechnikov National University, Odesa, Ukraine.

Malakhov E. V. – Dr. Sc., Professor, Head of the Department of Mathematical Support of Computer Systems, Odesa I. I. Mechnikov National University, Odesa, Ukraine.

ABSTRACT

Context. The problem of determining the object-oriented design (OOD) complexity of the open-source software, including Web apps created using the PHP frameworks, is important because nowadays open-source software is growing in popularity and using the PHP frameworks making app development faster. The object of the study is the process of determining the OOD complexity of the open-source Web apps created using the PHP frameworks. The subject of the study is the mathematical models to determine the OOD complexity due to the identification of classes of the open-source Web apps created using the PHP frameworks.

Objective. The goal of the work is the build a mathematical model for determining the OOD complexity due to the identification of classes of the open-source Web apps created using the PHP frameworks based on the three-variate Box-Cox normalizing transformation to increase confidence in determining the OOD complexity of these apps.

Method. The mathematical model for determining the OOD complexity due to the identification of classes of the open-source Web apps created using the PHP frameworks is constructed in the form of the prediction ellipsoid equation for normalized metrics WMC, DIT, and NOC at the app level. We apply the three-variate Box-Cox transformation for normalizing the above metrics. The maximum likelihood method is used to compute the parameter estimates of the three-variate Box-Cox transformation.

Results. A comparison of the constructed model based on the *F* distribution quantile with the prediction ellipsoid equation based on the Chi-Square distribution quantile has been performed.

Conclusions. The mathematical model in the form of the prediction ellipsoid equation for the normalized WMC, DIT, and NOC metrics at the app level to determine the OOD complexity due to the identification of classes of the open-source Web apps created using the PHP frameworks is firstly built based on the three-variate Box-Cox transformation. This model takes into account the correlation between the WMC, DIT, and NOC metrics at the app level. The prospects for further research may include the use of other data sets to confirm or change the prediction ellipsoid equation for determining the OOD complexity due to the identification of classes of the open-source Web apps created using the PHP frameworks.

KEYWORDS: object-oriented design complexity, identification of classes, open-source software, Web app, prediction ellipsoid, Box-Cox transformation, depth of inheritance tree, number of children, weighted methods per class.

ABBREVIATIONS

BCT is the Box-Cox transformation;
CBO is coupling between object classes;
DIT is a depth of inheritance tree;
KLOC is a thousand lines of code;
LCOM is a lack of cohesion in methods;
NOC is a number of children;
OOD is the object-oriented design;
PHP is a hypertext preprocessor;
RFC is a response for a class;
RMSD is a root mean square deviation;
SMD is a squared Mahalanobis distance;
WMC is weighted methods per class.

NOMENCLATURE

k is a number of variables (metrics);
N is a number of data points;
$\mathbf{S_Z}$ is a sample covariance matrix for normalized data;
$\mathbf{X}$ is a non-Gaussian random vector;
$\overline{\mathbf{X}}$ is a vector of sample means of the X_j variables;
X_1 is a WMC metric at the app level;
X_2 is a DIT metric at the app level;
X_3 is a NOC metric at the app level;
X_j is a j-th non-Gaussian variable;
$\overline{X}_j$ is a sample mean of the X_j values;
$\mathbf{Z}$ is a Gaussian random vector;
$\overline{\mathbf{Z}}$ is a vector of sample means of the Z_j variables;
Z_j is a j-th Gaussian variable that is obtained by transforming variable X_j;
$\overline{Z}_j$ is a sample mean of the Z_j values;
α is a significance level;
β_1 is a multivariate skewness;
β_2 is a multivariate kurtosis;
ν is a number of degrees of freedom;
$\chi^2_{m,\alpha}$ is the Chi-Square distribution quantile with m degrees of freedom and significance level α;
$\boldsymbol{\psi}$ is a vector of multivariate normalizing transformation.

INTRODUCTION

It's known [1], that "Complexity is one of the basic problems that associated with software development tools and methods." And complexity is one of main components of quality. Today the creation of high-quality soft-

DOI 10.15588/1607-3274-2024-2-16

ware is one of the most main tasks in the software development industry.

The design in general, and the object-oriented design (OOD) in particular, is one of the important stages of software development [1]. Booch outlined four major steps involved in the OOD process [2]. The first step in OOD is the identification of classes. In this step, key abstractions in the problem space are identified and labeled as potential classes and objects.

Now many Web apps are created using the PHP frameworks making app development faster. The above also applies to open-source Web apps since nowadays, open-source software is growing in popularity [3]. This demands the construction of mathematical models for determining the OOD complexity due to the identification of classes of the open-source Web apps created using the PHP frameworks.

The object of study is the process of determining the OOD complexity of the open-source Web apps created using the PHP frameworks.

The subject of study is the mathematical models to determine the OOD complexity due to the identification of classes of the open-source Web apps created using the PHP frameworks.

The purpose of the work is the build a mathematical model for determining the OOD complexity due to the identification of classes of the open-source Web apps created using the PHP frameworks.

1 PROBLEM STATEMENT

Suppose given the original sample as the three-dimensional non-Gaussian data set following metrics at the app level: WMC X_1, DIT X_2, and NOC X_3 from N open-source Web apps created using PHP frameworks. Suppose that there are invertible three-variate normalizing transformation of non-Gaussian random vector $\mathbf{X} = \{X_1, X_2, X_3\}^T$ to Gaussian random vector $\mathbf{Z} = \{Z_1, Z_2, Z_3\}^T$ is given by

$$\mathbf{Z} = \boldsymbol{\psi}(\mathbf{X}) \tag{1}$$

and the inverse transformation for (1)

$$\mathbf{X} = \boldsymbol{\psi}^{-1}(\mathbf{Z}). \tag{2}$$

It is required to build the mathematical model for determining OOD complexity due to the identification of classes of open-source Web apps created using PHP frameworks based on the transformations (1) and (2).

2 REVIEW OF THE LITERATURE

In [4] Chidamber and Kemerer proposed a set of software metrics for OOD. According to [4], the six metrics are designed to measure the three non-implementation steps in Booch's definition of OOD. These are the metrics WMC, DIT, NOC, RFC, CBO, and LCOM, which define the OOD complexity in the above steps. In particular, the metrics WMC, DIT, and NOC define the OOD complexity due to the identification of classes in the first step. According to [5], the metrics WMC, DIT, and NOC are related to object definition which is one of the fundamental elements of OOD as outlined by Booch [6].

The metric set of Chidamber and Kemerer (CK) serves as a generalized solution for other researchers to rely on for particular purposes [7–14], including the OOD complexity [1, 15, 16].

In paper [1] the authors fitted the multiple linear regression equation to determine the OOD complexity based on the minimal set of complexity metrics which are defined using the CK metrics. Although the proposed linear regression equation is fruitful for quantifying the complexity of the OOD hierarchy, it does not directly allow taking into account the correlation between the factors – software metrics. That can affect the result of such quantifying the complexity.

The research in [16] "focuses on two primary topics: (1) how indirect coupling measurements can aid developers with maintenance tasks and (2) how indirect coupling metrics can quantify software complexity and size, leveraging weighted differences across techniques. The study presents a comprehensive set of measures designed to assist developers and project managers with project management and maintenance activities." Also, the research in [16] does not directly take into account the correlation between the software metrics. Although other researchers indicate a significant correlation between certain software metrics, including the CK metrics [7–9, 15].

In the paper [15] the authors constructed a prediction ellipse equation for the normalized RFC and CBO metrics based on the bivariate Box-Cox transformation (BCT). They apply the above equation to evaluate complexity at the third step in the OOD of apps developed in Java. The authors proposed to use the squared Mahalanobis distance (SMD) from the prediction ellipse equation for the normalized RFC and CBO metrics as the complexity indicator of OOD of open-source apps in Java from the point of view of the relationships between classes.

But the ellipse allows you to take into account the correlation only for two variables. In the case of three non-Gaussian variables, we need to apply the prediction ellipsoid for the normalized data [17].

That is why to determine the OOD complexity due to the identification of classes we apply the approach proposed in [15] with the only difference that we are going to use the prediction ellipsoid for the normalized metrics WMC, DIT, and NOC.

3 MATERIALS AND METHODS

The equation for the prediction ellipsoid for the normalized WMC, DIT, and NOC metrics is defined as

$$\left(\mathbf{Z}-\overline{\mathbf{Z}}\right)^T \mathbf{S}_{\mathbf{Z}}^{-1}\left(\mathbf{Z}-\overline{\mathbf{Z}}\right) = \frac{3\left(N^2-1\right)}{N\left(N-3\right)} F_{3,N-3,\alpha}, \tag{3}$$

DOI 10.15588/1607-3274-2024-2-16

where $\mathbf{Z}$ is Gaussian random vector, $\mathbf{Z}=\{Z_1,Z_2,Z_3\}^T$; $\overline{\mathbf{Z}}$ is the sample mean vector, $\overline{\mathbf{Z}}=\{\overline{Z}_1,\overline{Z}_2,\overline{Z}_3\}^T$; N is the data point number; $F_{2,N-2,\alpha}$ is a quantile of the F distribution with 3 and $N-3$ degrees of freedom; α is a significance level; $\mathbf{S_Z}$ is the sample covariance matrix

$$\mathbf{S_Z}=\frac{1}{N}\sum_{i=1}^{N}(\mathbf{Z}_i-\overline{\mathbf{Z}})(\mathbf{Z}_i-\overline{\mathbf{Z}})^T. \quad (4)$$

We can write matrix (4) in the form

$$\mathbf{S}_Z=\begin{pmatrix} S_{Z_1Z_1} & S_{Z_1Z_2} & S_{Z_1Z_3} \\ S_{Z_2Z_1} & S_{Z_2Z_2} & S_{Z_2Z_3} \\ S_{Z_3Z_1} & S_{Z_3Z_2} & S_{Z_3Z_3} \end{pmatrix}, \quad (5)$$

where $S_{Z_qZ_r}=\sum_{i=1}^{N}\left[Z_{q_i}-\overline{Z}_q\right]\left[Z_{r_i}-\overline{Z}_r\right]$, $q,r=1,2,3$.

To construct equation (3) for determining OOD complexity due to the identification of classes of open-source Web apps created using PHP frameworks, we used 121 apps hosted on GitHub for five well-known frameworks (CakePHP, CodeIgniter, Laravel, Symfony, and Yii). Moreover, we also used the apps from [18] for the CakePHP framework. We obtained the data set using the PhpMetrics tool [19] around the following software metrics at the app level: WMC, DIT, and NOC. Table 1 contains the descriptive statistics of that data set.

Table 1 – App Descriptive Statistics (N=121)

Metric Name	Min	Max	Mean	RMSD
App size in KLOC	1.008	339.674	30.982	47.667
Number of classes	25	7512	525.554	920.147
WMC	1.55	67.84	14.797	13.755
DIT	1.06	2.33	3.593	4.602
NOC	0.32	0.95	0.685	0.141

We checked the three-dimensional data for three-variate outliers. But before that, we tested the normality of three-variate data because well-known statistical methods are used to detect outliers in multivariate data under the assumption that the data is described by a multivariate Gaussian distribution [20]. We applied a multivariate normality test proposed by Mardia and based on measures of the multivariate skewness β_1 and kurtosis β_2 [21]. According to this test, the distribution of three-dimensional data of the above set is not Gaussian since the test statistic for multivariate skewness $N\beta_1/6$ of this data is greater than the quantile of the Chi-Square distribution, which is 25.19 for 10 degrees of freedom and 0.005 significance level.

That is why, as in [18], to detect multivariate outliers in the three-dimensional data (121 points), we use the statistical technique based on the multivariate normalizing transformations and the squared Mahalanobis distance (SMD) for normalized data. The SMD for normalized data is the left part of equation (3).

To normalize the data, we apply the three-variate BCT with components

$$Z_j=x(\lambda_j)=\begin{cases}\left(X_j^{\lambda_j}-1\right)/\lambda_j, & \text{if} \quad \lambda_j\neq 0;\\ \ln(X_j), & \text{if} \quad \lambda_j=0.\end{cases} \quad (6)$$

Here Z_j is a Gaussian variable; λ_j is a parameter of the Box-Cox transformation, $j=1,2,3$. We denoted metrics WMC, DIT, and NOC as X_1, X_2, and X_3, respectively.

The parameter estimates of the three-variate BCT for 121 data points are calculated by the maximum likelihood method according to [20] and are $\hat{\lambda}_1=-0.177882$, $\hat{\lambda}_2=-1.272192$, $\hat{\lambda}_3=1.811640$.

In our case, there are no multivariate outliers in three-dimensional non-Gaussian data since the SMD values for all 121 data points normalized using (6) are less than the statistic

$$\frac{3\left(N^2-1\right)}{N\left(N-3\right)}F_{3,N-3,\alpha}, \quad (7)$$

which equals 13.85 for the 0.005 significance level.

To calculate the SMD value for normalized data in equation (3) we have the sample mean vector $\overline{\mathbf{Z}}$ with components 1.869, 0.284, and –0.265, respectively, and matrix (5)

$$\mathbf{S}_Z=\begin{pmatrix} 0.30283 & -0.01204 & 0.00775 \\ -0.01204 & 0.01370 & 0.00726 \\ 0.00775 & 0.00726 & 0.00967 \end{pmatrix}. \quad (8)$$

In our case, the inverse matrix of matrix (8) is

$$\mathbf{S}_Z^{-1}=\begin{pmatrix} 3.877 & 8.398 & -9.411 \\ 8.398 & 139.516 & -111.477 \\ -9.411 & -111.477 & 194.605 \end{pmatrix}. \quad (9)$$

Equation (3) for the 0.005 significance level defines the prediction ellipsoid boundary beyond which three-variate outliers can appear.

To determine OOD complexity due to the identification of classes of open-source Web apps created using PHP frameworks by (3), we need to use in (3) statistic (7) for the 0.05 significance level. Statistic (7) equals 8.25 for the 0.05 significance level.

Like [15], we can use equation (3) for determining the OOD complexity due to the identification of classes of open-source Web apps created using PHP frameworks. To do this we need to calculate the SMD value for normal-

DOI 10.15588/1607-3274-2024-2-16

ized data. If the SMD value for normalized data is greater than 13.85, then this data point is the three-variate outlier. In this case, we cannot determine OOD complexity by (3). If the SMD value for normalized data is greater than 8.25 and less than 13.85, it means that the app has high complexity due to the identification of classes. Otherwise, there is no high complexity due to the identification of classes for the Web app created using the PHP framework.

4 EXPERIMENTS

For comparison of equation (3) with another equation for the prediction ellipsoid for the normalized WMC, DIT, and NOC metrics, we built the corresponding equation in the form

$$\left(\mathbf{Z}-\overline{\mathbf{Z}}\right)^T \mathbf{S}_{\mathbf{Z}}^{-1}\left(\mathbf{Z}-\overline{\mathbf{Z}}\right)=\chi^2_{3,\alpha}, \qquad (10)$$

where $\chi^2_{3,\alpha}$ is the quantile of the Chi-Square distribution with 3 degrees of freedom and a significance level α. Other notations are the same as in (3).

In our case, there are no multivariate outliers in three-dimensional non-Gaussian data since the SMD values for all 121 data points normalized using (6) are less than the quantile of the Chi-Square distribution, which equals 12.84 for the 0.005 significance level.

To determine OOD complexity due to the identification of classes of open-source Web apps created using PHP frameworks by (10), we need to use in (10) the quantile $\chi^2_{3,\alpha}$ for the 0.05 significance level. The quantile $\chi^2_{3,\alpha}$ equals to 7.81 for the 0.05 significance level.

Like applying (3), we can use equation (10) to determine the OOD complexity due to the identification of classes of open-source Web apps created using PHP frameworks. To do this we also need to calculate the SMD value for normalized data. If the SMD value for normalized data is greater than 12.84, then this data point is the three-variate outlier. In this case, we cannot determine OOD complexity by (10). If the SMD value for normalized data is greater than 7.81 and less than 12.84, it means that the app has high complexity due to the identification of classes. Otherwise, there is no high complexity due to the identification of classes for the Web app created using the PHP framework.

We developed the computer program implementing built equations (3) and (10) to conduct experiments. The program was written in Python.

5 RESULTS

The results of determining OOD complexity level (CL) due to the identification of classes for ten open-source Web apps created using various PHP frameworks are shown in Table 2. We selected two such apps for each framework, for which normalized metrics have SMD values of both maximum and minimum. We denoted the SMD value for normalized metrics as SMD_Z in Table 2. Also, we denoted metrics WMC, DIT, and NOC at the app level as X_1, X_2, and X_3, respectively.

The SMD_Z values from Table 2 indicate there is no high complexity due to the identification of classes for eight apps (rows 3–10) because its SMD_Z values are less than 7.81. These are the following apps: Wildflower, Croogo, AdaptCMS, Wallabag, Yii2-podium, Yupe, Classroombookings, and Electronic invoicing and warehouse management system.

Wildflower (https://github.com/klevo/wildflower) is a CakePHP Content Management System. Croogo (https://github.com/croogo/croogo) is a CakePHP-powered Content Management System. AdaptCMS (https://github.com/adaptcms/AdaptCMS) is an open-source CMS that is made using the Laravel framework for complete control of your website. Wallabag (https://github.com/wallabag/wallabag) is a web application that allows you to save web pages for later reading and that is created using the Symfony framework. Yii2-podium (https://github.com/bizley/yii2-podium) is a Yii2 forum module project. Yupe (https://github.com/yupe/yupe) is an open-source Yii-framework-based online e-commerce solution. Classroombookings (https://github.com/classroombookings/classroombookings) is an open source hassle-free room booking system for schools that is made using the CodeIgniter framework. The electronic invoicing and warehouse management system (https://github.com/kirilkirkov/Electronic-Invoicing-And-Warehouse-Management-System) is a CodeIgniter and Bootstrap self-hosted open-source app.

Table 2 – OOD complexity levels of ten apps

N	App Name	X_1	X_2	X_3	SMD_Z	CL
1	Apiato	1.55	2.33	0.83	9.63	high
2	Ilios	8.82	1.22	0.86	8.59	high
3	Wildflower	60.34	1.56	0.95	7.77	no high
4	Croogo	9.8	1.64	0.75	0.53	no high
5	AdaptCMS	6.93	1.46	0.71	0.21	no high
6	Wallabag	7.75	1.41	0.70	0.12	no high
7	Yii2-podium	16.41	2.17	0.79	4.86	no high
8	Yupe	10.07	1.59	0.83	1.23	no high
9	Classroombooking	16.34	1.35	0.79	1.55	no high
10	Electronic invoicing and warehouse management system	49.43	1.27	0.57	4.59	no high

In contrast, two apps (Apiato and Ilios) have high complexity due to the identification of classes because their SMD_Z values are greater than 7.81 and less than 12.84. Apiato (https://github.com/apiato/apiato) is a framework for building scalable and testable APi-centric apps with PHP, built on top of Laravel. Ilios (https://github.com/ilios/ilios) is the Curriculum Management System for Health Professions that is made using the Symfony framework.

6 DISCUSSION

We apply the three-variate BCT to build the prediction ellipsoid equation for the normalized WMC, DIT, and NOC metrics for determining OOD complexity due to the

DOI 10.15588/1607-3274-2024-2-16

identification of classes of open-source Web apps created using PHP frameworks since the distribution of the three-dimensional data is not Gaussian on that the Mardia multivariate normality test based on measures of the multivariate skewness and kurtosis indicates. This is also the reason we use the statistical technique based on the multivariate normalizing transformations and the SMD for normalized data to detect three-variate outliers in the three-dimensional non-Gaussian data. According to [17, 20], we apply the 0.005 significance level for three-variate outlier detection. Also, we use the 0.05 significance level for both equations as (3) and (10). The use of both equations has led to the following results. Only two apps have high complexity. These are apps Apiato and Ilios (respectively, rows 1 and 2 in Table 2). All other 119 apps have no high complexity because the SMD values for their normalized metrics are less than 7.82. Note, that we used the 0.05 significance level that is usually assigned, although this value may be discussed.

The advantages of the proposed prediction ellipsoid equations (3) and (10) include the possibility of determining OOD complexity due to the identification of classes of open-source Web apps created using PHP frameworks. Also, the above equations take into account, firstly, the correlation between the normalized WMC, DIT, and NOC metrics, and secondly, that their three-variate distribution is not Gaussian.

Concerning the considered prediction ellipsoid equations for the normalized WMC, DIT, and NOC metrics for determining OOD complexity due to the identification of classes of open-source Web apps created using PHP frameworks, two limitations should be acknowledged and addressed concerning the data sample from 121 open-source apps in PHP. The first limitation concerns the processing of the data sample for open-source apps developed using PHP frameworks only. The processing of other data samples, for example, for commercial apps may affect the volume of the prediction ellipsoids. In such cases, equations (3) and (10) remain to be confirmed or changed. The second limitation concerns the following restrictions on software metrics at the app level: the interval for WMC is from 1.55 to 67.84, the interval for DIT is from 1.06 to 2.33, and the interval for NOC is from 0.32 to 0.95. In addition to the above restrictions, the SMD value for normalized WMC, DIT, and NOC metrics at the app level cannot be greater than 13.85 for equation (3) and cannot be greater than 12.84 for equation (10).

CONCLUSIONS

The important problem of determining the OOD complexity due to the identification of classes of the open-source Web apps created using the PHP frameworks is solved.

The scientific novelty of obtained results is that the mathematical model in the form of the prediction ellipsoid equation for the normalized WMC, DIT, and NOC metrics at the app level to determine the OOD complexity due to the identification of classes of the open-source Web apps created using the PHP frameworks is firstly built based on the three-variate Box-Cox transformation. This model takes into account the correlation between the WMC, DIT, and NOC metrics at the app level.

The practical significance of the obtained results is that the software realizing the constructed model is developed in Python. The empirical study allows us to recommend the built model for use in practice.

Prospects for further research may include the use of other data sets to confirm or change the prediction ellipsoid equation for determining the OOD complexity due to the identification of classes and implement the algorithms developed in [22] for the classification of mass problems of production subject domains to design the Web apps which are created using the PHP frameworks.

ACKNOWLEDGEMENTS

This work is proactive. The research was carried out within the framework of the scientific activity of the working hours of the authors according to their main positions.

REFERENCES

1. Khan S. A., Khan R. A. Object oriented design complexity quantification model, *Procedia Technology*, 2012, Vol. 4, pp. 548–554. DOI: 10.1016/j.protcy.2012.05.087.
2. Booch G. Object oriented design with applications. Redwood City, CA, Benjamin/Cummings, 1991, 580 p.
3. Madaehoh A., Senivongse T. OSS-AQM: An open-source software quality model for automated quality measurement, *Data and Software Engineering (ICoDSE) : the 2022 International Conference*. Denpasar, Indonesia, proceedings, IEEE, 2022, pp. 126–131. DOI: 10.1109/ICoDSE56892.2022.9972135
4. Chidamber S. R., Kemerer C.F. Towards a metrics suite for object oriented design, *ACM SIGPLAN Notices*, 1991, Vol. 26, Issue 11, pp. 197–211. DOI: 10.1145/118014.117970
5. Chidamber S. R., Kemerer C. F. A metrics suite for object-oriented design, *IEEE Transactions on Software Engineering*, 1994, Vol. 20, No. 6, pp. 476–493. DOI: 10.1109/32.295895
6. Booch G. Object-oriented development, *IEEE Transactions on Software Engineering*, 1986, Vol. 12, No. 1, pp. 211–221. DOI: 10.1109/TSE.1986.6312937
7. Barkmann H., Lincke R., Löwe W. Quantitative evaluation of software quality metrics in open-source projects, *Advanced Information Networking and Applications Workshops, 2009 International Conference.* Bradford, UK, 2009, proceedings, pp. 1067–1072. DOI: 10.1109/WAINA.2009.190
8. Sabahat N., Afzal Malik A., Azam F. Utility of CK metrics in predicting size of board-based software games, *Mehran University Research Journal of Engineering and Technology*, 2017, Vol. 36, No. 4, pp. 975–986.
9. Molnar AJ., Neamțu A., Motogna S. Evaluation of software product quality metrics, in E. Damiani, G. Spanoudakis, L. Maciaszek. Eds. *Evaluation of Novel Approaches to Software Engineering*. ENASE 2019. Communications in Computer and Information Science, Vol. 1172, Springer, Cham, 2020, pp. 163–187. DOI: 10.1007/978-3-030-40223-5_8

DOI 10.15588/1607-3274-2024-2-16

10. Rizwan M., Nadeem A., Sindhu M. A. Empirical evaluation of coupling metrics in software fault prediction, *Applied Sciences and Technology (IBCAST), 2020 17th International Bhurban Conference*. Islamabad, Pakistan, 2020, proceedings, IEEE, 2020, pp. 434–440. DOI: 10.1109/IBCAST47879.2020.9044489
11. Tapia V., Gaona C. Research opportunities in microservices quality assessment: A systematic literature review, *Journal of Advances in Information Technology*, 2023, Vol. 14, No. 5, pp. 991–1002. DOI: 10.12720/jait.14.5.991-1002
12. Wikantyasa I. M. A., Kurniawan A. P., Rochimah S. CK metric and architecture smells relations: Towards software quality assurance, *Information & Communication Technology and System (ICTS), 2023 14th International Conference*. Surabaya, Indonesia, 2023, proceedings, IEEE, 2023, pp. 13–17. DOI: 10.1109/ICTS58770.2023.10330874
13. Jin W. Zhang Y., Shang J. et al. Identifying code changes for architecture decay via a metric forest structure, *Technical Debt (TechDebt) : 2023 ACM/IEEE International Conference*. Melbourne, Australia, 2023, proceedings, pp. 62–71. DOI: 10.1109/TechDebt59074.2023.00014
14. Levasseur M., Badri M. Prioritizing unit tests using object-oriented metrics, centrality measures, and machine learning algorithms [Electronic resource], *Innovations in Systems and Software Engineering*, 2024. DOI: 10.1007/s11334-024-00550-9. Access mode: https://doi.org/10.1007/s11334-024-00550-9
15. Prykhodko S., Prykhodko N., Smykodub T. A joint statistical estimation of the RFC and CBO metrics for open-source applications developed in Java, *Computer Sciences and Information Technologies : the 2022 IEEE 17th International Conference (CSIT)*. Lviv, Ukraine, 10–12 November, 2022, proceedings. pp. 442–445. DOI: 10.1109/CSIT56902.2022.10000457
16. Navas-Su J., Gonzalez-Torres A., Hernandez-Vasquez M. et al. A metrics suite for measuring indirect coupling complexity, *Programming and Computer Software*, 2023, Vol. 49, Issue 8, pp. 735–761. DOI: 10.1134/S0361768823080157
17. Prykhodko S. Makarova L., Prykhodko K. et al. Application of transformed prediction ellipsoids for outliers detection in multivariate non-gaussian data, *Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET) : the 15th International Conference, IEEE*, Lviv-Slavske, 2020, proceedings, pp. 359–362. DOI: 10.1109/TCSET49122.2020.235454
18. Prykhodko S. B., Shutko I. S., Prykhodko A. S. A nonlinear regression model to estimate the size of web apps created using the CakePHP framework, *Radio Electronics, Computer Science, Control*, 2021, Vol. 59, No. 4, pp. 129–139. DOI: 10.15588/1607-3274-2021-4-12
19. PhpMetrics is a static analysis tool for PHP [Electronic resource]. Access mode: https://phpmetrics.org/
20. Johnson R. A., Wichern D. W. Applied multivariate statistical analysis. Pearson Prentice Hall, 2007, 800 p.
21. Mardia K.V. Measures of multivariate skewness and kurtosis with applications, *Biometrika*, 1970, Vol. 57, pp. 519–530. DOI: 10.1093/biomet/57.3.519
22. Malakhov E., Shchelkonogov D., Mezhuyev V.Algorithms of classification of mass problems of production subject domains, *Software and Computer Applications (ICSCA 2019) : 2019 8th International Conference, Feb. 19–21, 2019*. Penang, Malaysia, proceedings, pp. 149–153. DOI: 10.1145/3316615.3316676

Received 20.02.2024.
Accepted 01.04.2024.

УДК 004.412:519.237.5

ВИЗНАЧЕННЯ СКЛАДНОСТІ ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОЕКТУВАННЯ ЗАВДЯКИ ІДЕНТИФІКАЦІЇ КЛАСІВ ВЕБ-ЗАСТОСУНКІВ З ВІДКРИТИМ КОДОМ, СТВОРЕНИХ ЗА ДОПОМОГОЮ PHP-ФРЕЙМВОРКІВ

Приходько А. С. – аспірант кафедри математичного забезпечення комп'ютерних систем Одеського національного університету імені І. І. Мечникова, Одеса, Україна.

Малахов Є. В. – д-р техн. наук, професор, завідувач кафедри математичного забезпечення комп'ютерних систем Одеського національного університету імені І. І. Мечникова, Одеса, Україна.

АНОТАЦІЯ

Актуальність. Проблема визначення складності об'єктно-орієнтованого проектування (ООП) програмного забезпечення з відкритим вихідним кодом, включаючи веб-програми, створені за допомогою фреймворків PHP, є важливою, оскільки сьогодні програмне забезпечення з відкритим кодом стає все популярнішим і використання фреймворків PHP робить розробку застосунків швидшою. Об'єктом дослідження є процес визначення складності ООП веб-застосунків з відкритим кодом, створених за допомогою фреймворків PHP. Предметом дослідження є математичні моделі для визначення складності ООП завдяки ідентифікації класів веб-застосунків з відкритим кодом, створених за допомогою фреймворків PHP.

Мета. Метою роботи є побудова математичної моделі для визначення складності ООП завдяки ідентифікації класів веб-застосунків з відкритим кодом, створених з використанням фреймворків PHP, на основі тривимірного нормалізуючого перетворення Бокса-Кокса для підвищення достовірності визначення складність ООП цих застосунків.

Метод. Математична модель для визначення складності ООП завдяки ідентифікації класів веб-застосунків з відкритим кодом, створених за допомогою фреймворків PHP, побудована у формі рівняння еліпсоїда прогнозування для нормалізованих метрик WMC, DIT і NOC на рівні застосунку. Ми застосовуємо тривимірне перетворення Бокса-Кокса для нормалізації наведених вище метрик. Метод максимальної правдоподібності використовується для обчислення оцінок параметрів тривимірного перетворення Бокса-Кокса.

Результати. Проведено порівняння побудованої моделі на основі квантиля F-розподілу з рівнянням еліпсоїда прогнозування на основі квантиля розподілу хі-квадрат.

Висновки. Математична модель у формі рівняння еліпсоїда прогнозування для нормалізованих метрик WMC, DIT та NOC на рівні програми для визначення складності ООП через ідентифікацію класів веб-застосунків з відкритим кодом, створених за допомогою фреймворків PHP, у перше побудована на основі тривимірного перетворення Бокса-Кокса. Ця мо-

DOI 10.15588/1607-3274-2024-2-16

дель враховує кореляцію між метриками WMC, DIT та NOC на рівні програми. Перспективи подальших досліджень можуть включати використання інших наборів даних для підтвердження або зміни рівняння еліпсоїда прогнозування для визначення складності ООП завдяки ідентифікації класів веб-застосунків з відкритим кодом, створених за допомогою фреймворків PHP.

КЛЮЧОВІ СЛОВА: складність об'єктно-орієнтованого проектування, ідентифікація класів, програмне забезпечення з відкритим кодом, веб-застосунок, еліпсоїд прогнозування, перетворення Бокса-Кокса, глибина дерева успадкування, кількість дітей, зважені методи на клас.

ЛІТЕРАТУРА

1. Khan S. A. Object oriented design complexity quantification model / Suhel Ahmad Khan, Raees Ahmad Khan // Procedia Technology. – 2012. – Vol. 4 – P. 548–554. DOI: 10.1016/j.protcy.2012.05.087.
2. Booch G. Object oriented design with applications / G. Booch. – Redwood City, CA: Benjamin/Cummings, 1991. 580 p.
3. Madaehoh A. OSS-AQM: An open-source software quality model for automated quality measurement / A. Madaehoh, T. Senivongse // Data and Software Engineering (ICoDSE) : the 2022 International Conference, Denpasar, Indonesia : proceedings. – IEEE, 2022. – P. 126–131. DOI: 10.1109/ICoDSE56892.2022.9972135
4. Chidamber S. R. Towards a metrics suite for object oriented design / S. R. Chidamber, C. F. Kemerer // ACM SIGPLAN Notices. – 1991. – Vol. 26, Issue 11. – P. 197–211. DOI: 10.1145/118014.117970
5. Chidamber S. R. A metrics suite for object-oriented design / S. R. Chidamber, C. F. Kemerer // IEEE Transactions on Software Engineering. – 1994. – Vol. 20, No. 6. – P. 476–493. DOI: 10.1109/32.295895
6. Booch G. Object-oriented development / G. Booch // IEEE Transactions on Software Engineering. – 1986. – Vol. 12, No. 1. – P. 211–221. DOI: 10.1109/TSE.1986.6312937
7. Barkmann H. Quantitative evaluation of software quality metrics in open-source projects / H. Barkmann, R. Lincke, W. Löwe // Advanced Information Networking and Applications Workshops : 2009 International Conference, Bradford, UK, 2009 : proceedings. – P. 1067–1072. DOI: 10.1109/WAINA.2009.190
8. Sabahat N. Utility of CK metrics in predicting size of board-based software games / N. Sabahat, A. Afzal Malik, F. Azam // Mehran University Research Journal of Engineering and Technology. – 2017. – Vol. 36, No. 4. – P. 975–986.
9. Molnar AJ. Evaluation of software product quality metrics / AJ. Molnar, A. Neamţu, S. Motogna // E. Damiani, G. Spanoudakis, L. Maciaszek. Eds. Evaluation of Novel Approaches to Software Engineering. ENASE 2019. Communications in Computer and Information Science, Vol. 1172. – Springer, Cham, 2020. – P. 163–187. DOI: 10.1007/978-3-030-40223-5_8
10. Rizwan M. Empirical evaluation of coupling metrics in software fault prediction / M. Rizwan, A. Nadeem, M. A. Sindhu // Applied Sciences and Technology (IBCAST) : 2020 17th International Bhurban Conference, Islamabad, Pakistan, 2020 : proceedings. – IEEE, 2020. – P. 434–440. DOI: 10.1109/IBCAST47879.2020.9044489
11. Tapia V. Research opportunities in microservices quality assessment: A systematic literature review / V. Tapia, C. Gaona // Journal of Advances in Information Technology. – 2023. – Vol. 14, No. 5. – P. 991–1002. DOI: 10.12720/jait.14.5.991-1002
12. Wikantyasa I. M. A. CK metric and architecture smells relations: Towards software quality assurance / I. M. A. Wikantyasa, A. P. Kurniawan, S. Rochimah // Information & Communication Technology and System (ICTS) : 2023 14th International Conference, Surabaya, Indonesia, 2023 : proceedings. – IEEE, 2023. – P. 13–17. DOI: 10.1109/ICTS58770.2023.10330874
13. Identifying code changes for architecture decay via a metric forest structure / [W. Jin, Y. Zhang, J. Shang et al.] // Technical Debt (TechDebt) : 2023 ACM/IEEE International Conference, Melbourne, Australia, 2023: proceedings. – P. 62–71. DOI: 10.1109/TechDebt59074.2023.00014
14. Levasseur M. Prioritizing unit tests using object-oriented metrics, centrality measures, and machine learning algorithms [Electronic resource] / M. Levasseur, M. Badri // Innovations in Systems and Software Engineering. – 2024. DOI: 10.1007/s11334-024-00550-9. – Access mode: https://doi.org/10.1007/s11334-024-00550-9
15. Prykhodko S. A joint statistical estimation of the RFC and CBO metrics for open-source applications developed in Java / S. Prykhodko, N. Prykhodko, T. Smykodub // Computer Sciences and Information Technologies : the 2022 IEEE 17th International Conference (CSIT), Lviv, Ukraine, 10–12 November, 2022 : proceedings. – P. 442–445. DOI: 10.1109/CSIT56902.2022.10000457
16. Navas-Su J. A metrics suite for measuring indirect coupling complexity / [J. Navas-Su, A. Gonzalez-Torres, M. Hernandez-Vasquez et al.] // Programming and Computer Software. – 2023. – Vol. 49, Issue 8. – P. 735–761. DOI: 10.1134/S0361768823080157
17. Application of transformed prediction ellipsoids for outliers detection in multivariate non-gaussian data / [S. Prykhodko, L. Makarova, K. Prykhodko et al.] // Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET) : the 15th International Conference, IEEE, Lviv-Slavske, 2020 : proceedings. – P. 359–362. DOI: 10.1109/TCSET49122.2020.235454
18. Prykhodko S.B. A nonlinear regression model to estimate the size of web apps created using the CakePHP framework / S. B. Prykhodko, I. S. Shutko, A. S. Prykhodko // Radio Electronics, Computer Science, Control. – 2021. – Vol. 59. No. 4. – P. 129–139. DOI: 10.15588/1607-3274-2021-4-12
19. PhpMetrics is a static analysis tool for PHP [Electronic resource]. – Access mode: https://phpmetrics.org/
20. Johnson R. A. Applied multivariate statistical analysis / R. A. Johnson, D. W. Wichern. – Pearson Prentice Hall, 2007. – 800 p.
21. Mardia K. V. Measures of multivariate skewness and kurtosis with applications / K. V. Mardia // Biometrika. – 1970. – Vol. 57. – P. 519–530. DOI: 10.1093/biomet/57.3.519
22. Malakhov E. Algorithms of classification of mass problems of production subject domains / E. Malakhov, D. Shchelkonogov, V. Mezhuyev // Software and Computer Applications (ICSCA 2019) : 2019 8th International Conference, Feb. 19–21, 2019, Penang, Malaysia : proceedings. – P. 149–153. DOI: 10.1145/3316615.3316676

DOI 10.15588/1607-3274-2024-2-16

УПРАВЛІННЯ У ТЕХНІЧНИХ СИСТЕМАХ

CONTROL IN TECHNICAL SYSTEMS

УДК 658.5

ФОРМАЛІЗАЦІЯ ЗАДАЧІ ФОРМУВАННЯ ГОЛОВНОГО КАЛЕНДАРНОГО ПЛАНУ В СИСТЕМІ ПЛАНУВАННЯ MRP II

Новінський В. П. – канд. техн. наук, доцент кафедри Інформатики і програмної інженерії Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна.

Попенко В. Д. – канд. техн. наук, доцент кафедри Інформаційних систем і технологій Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна.

АНОТАЦІЯ

Актуальність. Розглянуто задачу формування Головного календарного плану в процесі управління виробництвом на базі стандарту MRP II. Об'єктом дослідження є алгоритм формування цього плану для подальшого планування постачання матеріалів у виробництво і організації самого виробництва.

Мета роботи – вдосконалення алгоритму формування Головного календарного плану для уникнення зайвих стадій алгоритму.

Метод. Запропоновано вдосконалення алгоритму формування Головного календарного плану. Воно полягає в одночасному врахуванні вимог щодо своєчасної реалізації замовникам продукції, обмежень щодо потужностей робочих центрів підприємства і обмежень щодо тривалості закупівельних циклів у процесі постачання матеріалів. У стандарті MRP II передбачені спочатку планування термінів і кількості випуску продуктів, а лише на наступному кроці перевірка сформованого плану на припустимість щодо потрібного часу роботи обладнання і доступності потрібних кількостей матеріалів. У разі порушення обмежень розрахованим планом треба або планувати і здійснювати заходи для подолання вказаних обмежень, тобто організовувати додаткові зміни для робочих центрів, задіяти додаткові потужності, пришвидшити доставку деяких матеріалів, або зменшувати план продажу. Всі ці заходи пов'язані з додатковими витратами. В запропонованому варіанті процесу планування це треба робити лише якщо алгоритм не знайде припустимого рішення. Задача формування Головного календарного плану, яка є центральною в стандарті MRP, сформульована авторами як задача лінійного програмування завдяки лінійному характеру вказаних обмежень по виробничих потужностях і матеріалах. Зокрема, в разі достатньо жорстких обмежень на потужності робочих центрів план поповнення залишків продуктів з виробництва зсувається на більш ранні інтервали планування і лише після цього упирається в обмеження. Запропоновано кілька стратегій планування поповнень з виробництва залишків продуктів.

Результати. Розроблені алгоритми реалізовані в формі шаблонів Microsoft Excel і доступні для використання з метою поглиблення розуміння стандарту MRP II. Також вони використовуються в навчальному процесі.

Висновки. Апробація рішення авторами підтвердила його працездатність, а також доцільність впровадження розробленої модифікації процесу планування MRP II у програмне забезпечення провідних постачальників систем класу ERP. Перспективи подальших досліджень можуть полягати в порівняльному аналізі запропонованих варіантів розміщення поповнень залишків продуктів з виробництва, шляхом економічного оцінювання цих варіантів, а також шляхом імітаційного моделювання.

КЛЮЧОВІ СЛОВА: планування, виробництво, матеріали, продукти, напівпродукти, MRP II.

АБРЕВІАТУРИ

APICS – American Production and Inventory Control Society;

CPS – Cyber-physical system;

CRP – Capacity Requirements Plan, укр. ПЗП – План завантаження потужностей;

DDMRP – Demand Driven MRP;

IoT – Internet of Things;

JIT – Just in time;

MPS – Master Production Schedule, укр. ГКП – Головний календарний план;

MRP II – Manufacturing Resource Planning, стандарт де-факто;

S&OP – Sales & Operations Plan;

ToC – Theory of Constraints.

НОМЕНКЛАТУРА

p – номер продукту;

t – номер інтервалу зони планування;

plp_{pt} – планова кількість для пари продукт-інтервал, це кількість продукту p, яку треба виробити не пізніше інтервалу t;

DOI 10.15588/1607-3274-2024-2-17

pst_{pt} – прогнозний запас продукту p на початок інтервалі зони планування t;

sal_{pt} – потреба в продукті p і в інтервалі t за планом продажів;

sop_{pt} – потреба в продукті p і в інтервалі t за планом продажів і операцій;

w – номер робочого центру;

rqw_{wt} – потреба в часі робочого центру;

nw_{pw} – норма продуктивності робочого центру w при виробництві продукту p (у штуках на годину);

fnt_{wt} – фонд часу робочого центру w в інтервалі t;

fnw_{wt} – сумарна, починаючи з інтервалу 1 до інтервалу t, потреба в робочому часі робочого центру w;

nm_{pm} – норма витрат матеріалу m при виробництві продукту p;

rqm_{mt} – планова потреба витрат матеріалу m у виробництві в інтервалі t;

stm_{mt} – прогнозний рівень запасу матеріалу $m = 1, \ldots, M$ в інтервалі зони планування t;

dc_m – цикл постачання (delivery cycle) для матеріалу m;

plm_m – планова партія поставки для матеріалу m.

$plp^{\max}_{pt}$ – верхня межа плану поповнення запасу продукту p в інтервалі t.

ВСТУП

В наш час найбільш відомою методикою управління виробництвом є методика MRP II (Manufacturing Resource Planning), вона була розроблена в кінці 20-го сторіччя американською асоціацією APICS (American Production and Inventory Control Society), та у 21-му сторіччі набула додаткового розвитку та просування в практику управління у виробничій бізнес сфері. В наші дні вона є основною методикою, на яку орієнтуються інформаційні системи в цій області. Це визначається наступними факторами.

1. MRP II є найбільш комплексною теорією, яка не залежить від типу виробництва (дискретне чи неперервне; одиничне, серійне чи масове; виробництво на склад чи на замовлення).

2. MRP II визначає побудову системи управління, як сукупність та послідовність виконання фаз управління: планування, організацію діяльності, обліку та нормування. Для цих фаз представлено сукупність процедур та методів, які повинні бути реалізовані.

3. MRP II визначає архітектуру інформаційної системи та їх взаємозв'язків. Разом з обов'язковим переліком бізнес-процедур та методів їх реалізація в рамках інформаційної системи є обов'язковим для практичного застосування MRP II.

4. MRP II є основою системи класу ERP (Enterprise Resource Planning). На світовому ринку ERP систем кожний вендор – розробник системи цього класу має методику MRP II, реалізовану в системі.

APICS в APICS Dictionary [1] дає наступне визначення систем класу MRP II.

Планування ресурсів виробництва (Manufacturing Resource Planning – MRP II) – метод ефективного планування всіх ресурсів виробничої компанії. Він використовує операційне планування в натуральних одиницях виміру, фінансове планування у вартісних одиницях виміру, і містить у собі можливості відповіді на питання «що буде, якщо…?» шляхом моделювання. Метод складається з безлічі процесів, кожен з яких пов'язаний з іншими: бізнес-планування, планування виробництва (планування продажів та операцій), розробка головного календарного плану виробництва, планування потреби в матеріалах, планування потреби в потужностях та системи підтримки контролю виконання за потужностями та матеріалами. Результат таких систем використовується фінансовими звітами, такими як бізнес-план, звіт про угоди щодо закупівель, бюджет відвантаження та прогноз запасів у вартісному вираженні.

Ресурсами виробництва, які підлягають управлінню в рамках MRP II, є:

– готова продукція та напівпродукти на стадіях виробництва та реалізації;

– виробничі потужності;

– покупні матеріали на стадіях постачання та використання.

Логіка роботи MRP II представлена на рис. 1.

MRP II охоплює три напрямку управління виробничою системою: бізнес-(тактичне) планування; операційне (оперативне) планування; організацію та облік виконання планів.

В рамках бізнес-планування передбачене формування плану продажів та операцій (Sales & Operations Plan – S&OP). План розробляється не менше ніж на рік з розбивкою по кварталах по номенклатурі продукції (якщо її кількість велика – тисяча позицій та більше – тоді по групах номенклатури) в кількісних та вартісних показниках. Для розробки S&OP потрібен загальний (укрупнений) план продажів. В фінансовій сфері на рівні бізнес-планування отримують фінансові бюджети витрат та обігу грошових коштів.

Операційне планування – це формування трьох основних планів: головного календарного плану (ГКП, Master Production Schedule – MPS), плану завантаження потужностей (ПЗП, Capacity Requirements Plan – CRP), плану потреб у матеріалах (ППМ, Material Reqirements Plan – MRP).

MPS є об'ємним календарним планом. Його горизонт планування визначається технологічними циклами виробництва продуктів та закупівельними циклами постачання матеріалів. План формується як правило з розбивкою по тижнях по номенклатурі продукції в кількісних показниках.

Два інших плани CRP та MRP є виконавчими планами. Перший вказує порядок виконання операцій у виробництві. Розробляється з горизонтом в місяць з розбивкою по днях чи виробничих змінах. Регламентує послідовність виконання технологічних операцій та технологічних процесів з прив'язкою до конкретного технологічного обладнання (робочих центрів) та часу виконання. Другий план (Material Requirements Plan – MRP) вказує порядок виконання операцій в сфері постачання у виробництво необхідних покуп-

DOI 10.15588/1607-3274-2024-2-17

них комплектуючих та матеріалів. Розробляється з горизонтом від місяця до декількох місяців (відповідно до максимальних циклів постачання окремих матеріалів) з розбивкою по днях. Постачання конкретних партій конкретних матеріалів вказується з призначенням постачальників для кожної поставки.

Для формування CRP та MRP потрібні відповідні виробничі та постачальні нормативи: робочі центри, графіки їх роботи і відповідно фонди робочого часу, нормативи продуктивності, реєстри постачальників, цикли постачання тощо.

Напрямок організації та обліку виконання представлений в MRP II відповідними процесами. Виконанню підлягають CRP та MRP.

На рисунку 1 позначені зворотні зв'язки, які є обов'язковими для системи управління, якою є система MRP II. Якщо нема проблем з виконанням планів на рівні організації та обліку, управління передається на ніжній рівень планування – формування CRP та MRP. Якщо проблеми є на рівнях організації та обліку або на нижніх рівнях, може бути необхідним перепланувати MPS.

Формування MPS займає центральне місце в оперативному плануванні MRP II, це гарно видно на рис. 1. Тому змістом нашого дослідження є вдосконалення та розвиток MRP II в цій ланці оперативного планування, але для цього треба розглянути більш докладно саму процедуру та перелік дій формування MPS.

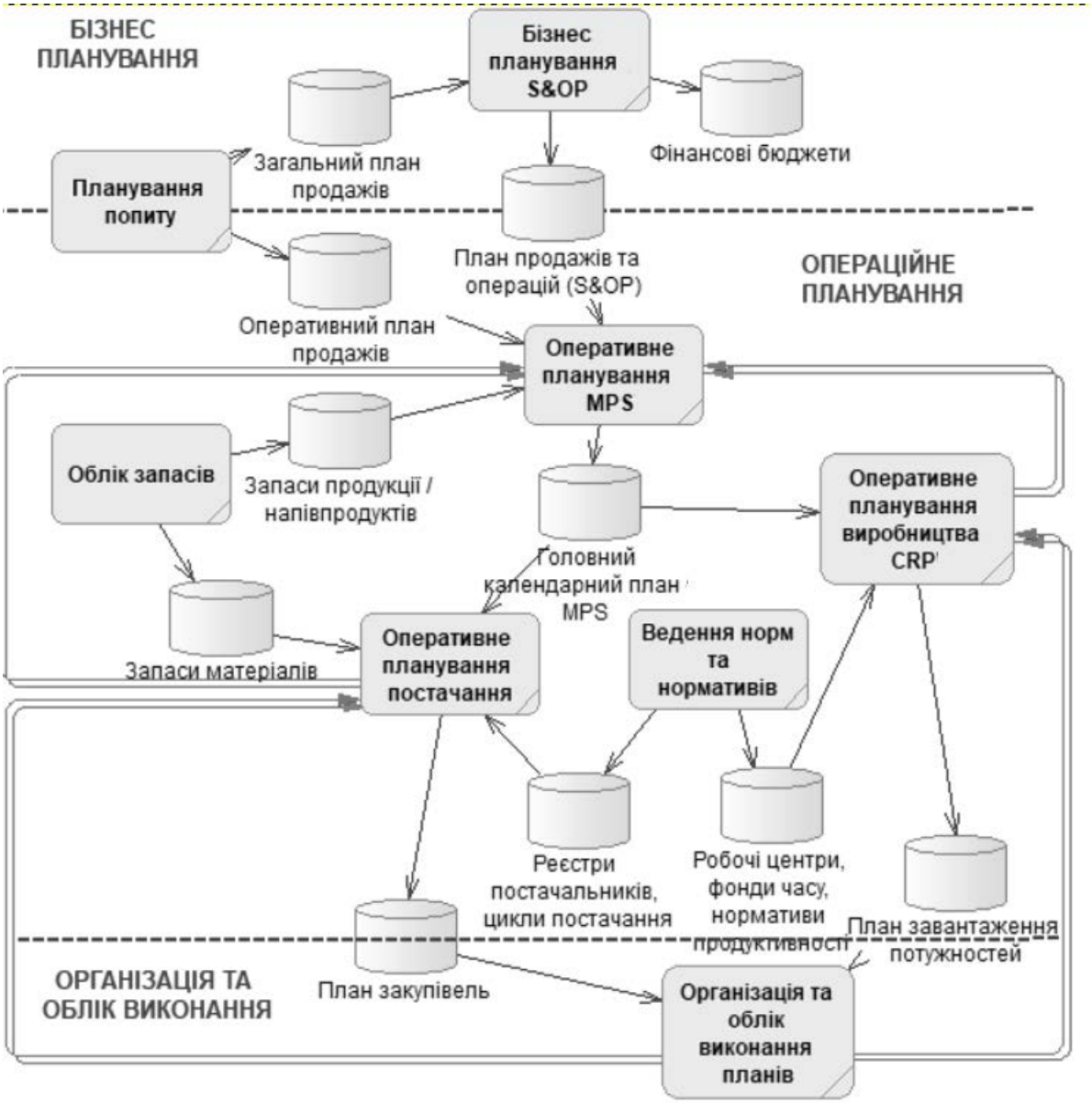


Рисунок 1 – Логіка роботи MRP II

DOI 10.15588/1607-3274-2024-2-17

У відповідності до [2, 3] схематично процедура формування MPS з переліком факторів та отриманих результатів представлена на рис. 2.

MPS складається з окремих позицій – планових замовлень (planned orders).

Генератор MPS – це процедура формування сукупності планових замовлень для різних продуктів (напівпродуктів) з урахуванням означених на рисунку факторів:

– бізнес плану MRP II – S&OP;

– інформації про замовлення покупців (клієнтів) та прогнози продажів продуктів;

– інформації про наявні запаси готової продукції та матеріалів;

– значення параметрів (модифікаторів) планування.

Формування MPS робиться в загальному випадку в два етапи.

На першому етапі формуються планові замовлення для продуктів. Вхідною інформацією для розрахунку є план продажів (закази покупців та прогнози продажів), план продажів та операцій, інформація про прогноз запасів продуктів на початок інтервалу планування. Робиться прогнозування падіння запасів готових продуктів та знаходяться для кожного продукту точки поповнення запасу – інтервали плану, які передують інтервалам з від'ємним значенням прогнозу запасу. Кількість у плановому замовленні приймається рівною дефіциту в наступному чи в декількох наступних інтервалах. Цей прийом розрахунку кількості має назву розрахунку по «переслідуванню дефіциту».

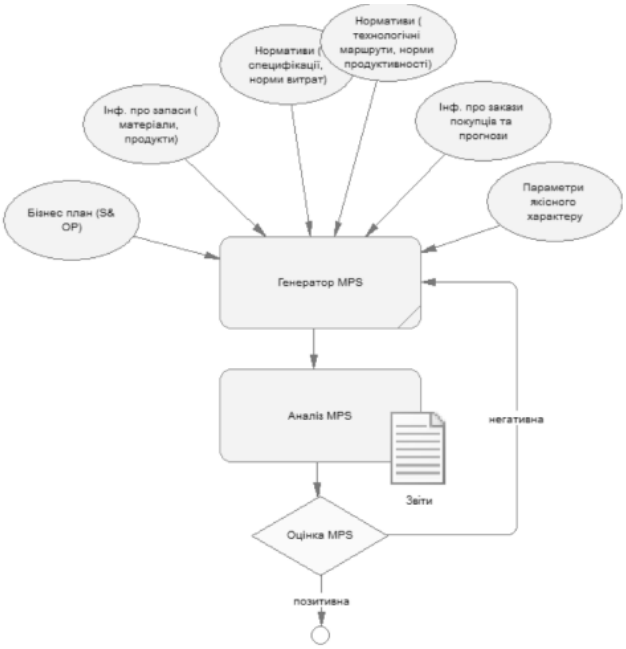


Рисунок 2 – Процедура формування MPS

На другому етапі процедури генерації MPS робиться «прогін» формування MRP. З урахуванням специфікацій продуктів розраховується кількості виробництва напівпродуктів для виробництва вже спланованих позицій продуктів. Планові кількості напівпродуктів обраховуються на основі планових кількостей продуктів та нормативів витрат матеріалів або напівпродуктів з відповідної специфікації.

Об'єктом досліджень є процес розробки MPS на базі стандарту MRP II. Відповідно до стандарту, план розраховується, а потім перевіряється на припустимість, тобто на дотримання обмежень по виробничих потужностях і доступності матеріалів.

Суб'єктом досліджень є алгоритм розробки MPS. В результаті запропонованого вдосконалення він має відразу задовольняти вказані обмеження, якщо це можливо.

Метою дослідження є збільшення ефективності і спрощення процедури розробки MPS, що є важливим, зважаючи на високі вимоги стандарту MRP II до рівня організованості підприємства.

В цілому представлена вище процедура формування MPS є багатоетапною з необхідністю виконання цих етапів в рамках багатоітеративного процесу. На погляд авторів, цього можливо уникнути, якщо задачу формування MPS звести до задачі лінійного програмування з сукупністю лінійних обмежень та лінійною цільовою функцією.

1 ПОСТАНОВКА ЗАДАЧІ

Розглянемо формалізацію задачі формування MPS у вигляді задачі лінійного програмування. Приймемо, що формування MPS виконується в зоні планування, яка розбита на окремі інтервали.

Змінними задачі лінійного програмування є планові кількості поповнення запасу певного продукту в певному інтервалі зони планування plp_{pt}, $p = 1, \ldots, P$, $t = 1, \ldots, T$.

Обмеження задачі:

1. Зміні задачі (планові кількості поповнень) не повинні бути менше нуля, це тривіальне обмеження.

$$plp_{pt} >= 0,\ p = 1, \ldots, P,\ t = 1, \ldots, T. \qquad (1)$$

2. Календарні запаси продуктів (product stock – запас продукту) не повинні бути негативними.

$$pst_{pt} >= 0,\ p = 1, \ldots, P,\ t = 1, \ldots, T. \qquad (2)$$

3. Сумарні об'єми виробництва повинні дорівнювати підсумковому первісному дефіциту (позначимо його pst^*_{pT+1}) – дефіциту в останньому інтервалі зони планування для кожного продукту p.

$$\sum_{t=1}^{T} plp_{pt} = -pst^*_{pT+1},\ p = 1,\ldots,P, \qquad (3)$$

де pst^*_{p1} – прогнозний запас продукту p в першому інтервалі зони планування; $pst^*_{pt} = pst^*_{pt-1} - (sal_{pt} + sop_{pt})$ – запас продукту p в довільному інтервалі $t = 2, \ldots, T$.

DOI 10.15588/1607-3274-2024-2-17

4. Потреба в часі робочого центру не є більшою ніж його фонд часу – фонд робочого центру для всіх робочих центрів (w = 1, …, W) та для всіх інтервалів зони планування (t = 1, …, T).

$$rqw_{wt} <= fnw_{wt},\ w = 1, \ldots, W,\ t = 1, \ldots, T, \qquad (4)$$

де $rqw_{wt} = \sum_{s=1}^{t} \sum_{p=1}^{P} \frac{plp_{ps}}{nw_{pw}}, w = 1,...,W,\ p = 1,...,P$,

$$fnw_{wt} = \sum_{s=1}^{t} fnt_{ws}, w = 1,...,W,\ p = 1,...,P.$$

Таким чином, fnw_{wt} – сумарна, починаючи з інтервалу 1 до інтервалу t, потреба в робочому часі робочого центру w.

stm_{m1} – прогнозний рівень запасу матеріалу m = 1, …, M в першому інтервалі зони планування, визначається попереднім плановим періодом, для інтервалів 2, …, T:

– якщо t менше чи дорівнює циклу постачання dc_m, $stm_{mt} = stm_{mt-1}$;

– якщо t більше dc_m,тоді $stm_{mt} = stm_{mt-1} + plm_m$.

5. Планові кількості витрат матеріалів (material requirement) на виробництво є не більшими ніж планові запаси матеріалів (material stock) для кожного інтервалу t = 1, …, T зони планування та кожного матеріалу m:

$$rqm_{mt} <= stm_{mt},\ m = 1, \ldots, M,\ t = 1, \ldots, T, \qquad (5)$$

де $rqm_{mt} = \sum_{s=1}^{t} \sum_{p=1}^{P} plp_{ps} * nm_{pm}, m = 1,...,M, t = 1,...,T.$

Запас матеріалу на протязі циклу постачання є незмінним, він дорівнює запасу першого інтервалу; для інтервалів, які лежать правіше циклу постачання, запас збільшується на величину планової партії поставки.

6. Цільова функція задачі – максимізація завантаження потужності всіх робочих центрів:

$$\sum_{w=1}^{W} \sum_{p=1}^{P} \sum_{t=1}^{T} \frac{plp_{pt}}{nw_{pw}} \to \max. \qquad (6)$$

2 ОГЛЯД ЛІТЕРАТУРИ

Після створення методики MRP е 70-х роках минулого сторіччя паралельно розвивались інші підходи до управління виробництвом. Відмітимо Теорію обмежень (Theory of Constraints, ToC) [4, 5], та JIT/Lean [6, 7] які подекуди сприймаються як альтернативи MRP II. На нашу думку, Теорія обмежень концентрується лише на одному, щоправда, важливому аспекті управління виробництвом: пошуку і оптимізації «вузьких місць» серед наявних робочих центрів, і не претендує на цілісну методику управління з розрахунком вичерпних планів, як MRP II. Що стосується JIT/Lean, то в фокусі цього підходу – ритмічне постачання матеріалів на робочі місця; це працює, якщо на складі вже є потрібні матеріали; відповідно, хтось має подбати про матеріали з довгими циклами закупівлі. Таким чином, JIT/Lean працює в рамках, створених завдяки MRP.

У XXI сторіччі методика MRP II має кілька напрямків подальшого розвитку. Мають місце спроби вписати її в контекст Industry 4.0 [8], застосовуючи сучасні математичні, технічні і програмні засоби: кібер-фізичні системи (cyber-physical systems, CPSs), Інтернет речей (the Internet of Things, IoT), хмарні обчислення, великі дані, імітаційне моделювання, штучний інтелект. Якщо, наприклад, очевидна доцільність застосування імітаційного моделювання в дослідженнях алгоритмів MRP для порівняння варіантів плану, то доцільність застосування цього методу в поточному процесі планування викликає сумніви навіть у авторів [8], адже оперативність і зрозумілість процесу MRP теж є його суттєвими перевагами.

В роботі [9] автори показали шляхом експериментів, що в разі великої кількості різних замовлень попередня їх класифікація по трьох групах (ABC inventory analysis) дозволяє зменшити кількість переналадок на 16%.

Має місце спроба об’єднати в одному методі MRP, JIT/Lean і ToC, що призвело до появи методики управління виробництвом Demand Driven MRP (DDMRP) [10–13]. Вона полягає у тому, що на основі даних потоку виробничих замовлень для продуктів і робочих центрів створюються місця накопичення продуктів (буфери), не лише для «вузьких місць», як у ToC. Динамічно розраховуються три рівні запасів для кожного буфера (мінімальний, середній, максимальний). У процесі планування планові замовлення на виробництво або постачання матеріалів поповнюють планові залишки буфера, спрощено кажучи, з мінімального до максимального рівня, аналогічно тому, як у JIT порожній буфер ініціює поставку матеріалу з попереднього робочого центру.

На наш погляд, в умовах непередбачуваності і невизначеності у виробничій сфері в Україні, які спричинені зовнішніми обставинами, складно пропагувати методи управління, в основі яких лежить Just In Time. Ймовірно, опора на очевидність і здоровий глузд, а також метод прямих послідовних розрахунків низки планів, характерні для стандарту MRP II, є вирішальними перевагами в умовах, коли необхідно довго підтримувати високий рівень організованості у виробництві. Починати треба саме зі стандарту MRP II. Проте бракує наукових робіт щодо вдосконалень самого стандарту MRP II, які би не висували додаткових організаційних вимог для підприємства. Саме на це спрямована наша стаття. У роботі [15] розглянута змістовна постановка задачі, а в даній роботі сформульована математична модель задачі і додані нові можливості.

DOI 10.15588/1607-3274-2024-2-17

3 МАТЕРІАЛИ І МЕТОДИ

Тут та далі всі позначення будемо намагатись робити мнемонічними та додатково вказувати це. Наприклад, *plp* (production planning – планування виробництва).

Запас продукту p на початок довільного інтервалу $t = 2, \ldots, T$ зони планування дорівнює запасу в попередньому інтервалі (pst_{pt-1}) за відніманням потреби за планом продажів (sal_{pt}) та планом продажів та операцій (sop_{pt}) та додаванням плану виробництва (plp_{pt}):

$$pst_{pt} = pst_{p1} - \sum_{s=1}^{t}(sal_{ps} + sop_{ps} - plp_{ps}).$$

Обмеження (3) надає задачі властивості: виробляти стільки, скільки вимагають план продажів та план продажів і операцій.

Запас продукту *pst** без урахування поповнення поступово переходить у дефіцит зі знаком мінус.

Представлення цільової функції у вигляді (6) має мало сенсу, тому що з урахуванням (3)

$$\sum_{w=1}^{W}\sum_{p=1}^{P}\sum_{t=1}^{T}\frac{plp_{pt}}{nw_{pw}} = \sum_{w=1}^{W}\sum_{p=1}^{P}\frac{-pst^*_{pt}}{nw_{pw}} = \text{const}.$$

Але можна ввести вагові коефіцієнти k_t (умовно назвемо їх «пріоритетами зсуву плану виробництва») та розглянути цільову функцію

$$\sum_{w=1}^{W}\sum_{p=1}^{P}\sum_{t=1}^{T} k_t \frac{plp_{pt}}{nw_{pw}} \to \max. \qquad (6a)$$

Якщо, наприклад, $k_t = 2^t$, план виробництва буде розподілятися в зоні планування в більш ранніх інтервалах. Якщо, наприклад, $k_t = 2^{(T-t+1)}$, план виробництва буде розподілятися в зоні планування в більш пізніх інтервалах. Таким чином, ми можемо враховувати, що для підприємства краще: виконувати план якомога пізніше (перший ряд коефіцієнтів, це відповідає принципам JIT/Lean), чи якомога раніше (другий ряд коефіцієнтів, це відповідає врахуванню часової вартості грошей).

Задача (1–5, 6a) є базовою задачею формування MPS у вигляді задачі лінійного програмування (назвемо її задачею варіанту 0).

Сформулюємо ще додаткові обмеження задачі, які дозволять отримати додаткові можливості обмеження розміщення поповнень продуктів в інтервалах зони планування. З метою надання дозволів або заборон на розміщення точок поповнення запасів продуктів введемо величини $plp^{\max}_{pt}$, $p = 1\ldots P$, $t=1, \ldots, T$ – верхню межу плану виробництва.

$$plp_{pt} <= plp^{\max}_{pt},\ p = 1, \ldots, P,\ t = 1\ldots, T. \qquad (7)$$

Нехай *PLP* – велика константа, а T^*_p – інтервал, що безпосередньо передує виникненню дефіциту продукту *p*.

Варіант 1. І нехай для всіх продуктів

$$plp^{\max}_{pt} = \begin{cases} 0, \text{ якщо } t < T^*_p, \\ PLP \text{ у протилежному випадку.} \end{cases} \qquad (8)$$

Тоді інтервали поповнення запасів продуктів можуть бути безпосередньо перед виникненням його дефіциту або пізніше.

Задача (1–5, 6a, 7a, 8) є розвитком базової задачі формування MPS у вигляді задачі лінійного програмування. Цю задачу назвемо задачею формування MPS з розміщенням плану відповідно до інтервалу дефіциту. Її також будемо називати задачею варіанту 1.

Варіант 2. Нехай для всіх продуктів

$$plp^{\max}_{pt} = \begin{cases} PLP, \text{ якщо } t = T^*_p \text{ та } \mathrm{mod}(t - T^*_p, T^*_p) = 0, \\ 0 \text{ у протилежному випадку.} \end{cases} \qquad (9)$$

Цю задачу назвемо задачею формування MPS з періодичним розміщенням поповнення запасів продуктів, або задачею варіанту 2.

4 ЕКСПЕРИМЕНТИ

Апробацію задачі формування MPS проведемо в середовищі MS Excel за допомогою ad-on Solver (Розв'язувач задач). Прототипування різних задач за допомогою Excel є широко розповсюдженим засобом практичної перевірки працездатності теоретичних положень [14]. Ця модель розглянута також у [15] переважно під економічним кутом зору. Поточна модель на відміну від попередньоївраховує можливу потребу зсунути план до початку або до кінця зони планування за допомогою коефіцієнтів зважування цільової функції по інтервалах. Апробацію проведем у наступному порядку.

– Проведемо апробацію задачі варіанту 0 (базової задачі). Спочатку для цільової функції (6a) з $k_t = 2^t$, $t = 1, \ldots, T$, потім з $k_t = 2^{(T-t+1)}$, $t = 1, \ldots, T$.

– Проведемо апробацію задачі варіанту 1 (з розміщенням плану відповідно до інтервалу дефіциту), це буде виконано зі урахуванням обмеження (8).

– Проведемо апробацію задачі варіанту 2 (з розміщенням плану в сукупності фіксованих точок поповнення), це буде виконано зі урахуванням обмеження (9).

У прикладі розглядається модель, яка містить три продукти (П1, П2, П3), один робочий центр та один матеріал. Зона планування – десять інтервалів. Посилання на приклад є в розділі Обговорення.

DOI 10.15588/1607-3274-2024-2-17

У комірках B2:K4 (аркуш «Базова модель») розташовані кількості плану продажів. У комірках C7:K9 – кількості початкового зменшення запасів, з урахуванням початкових запасів продуктів (B7:B9) та плану продажів. У комірках L7:L9 – кінцевий дефіцит. У комірках B12:K14 розташовані значення плану виробництва, який підлягає розрахунку (це змінні моделі). В L12:L14 – сумарна планова кількість виробництва. У комірках B18:K20 – планові кількості запасів з урахуванням плану виробництва.

В рядках 22:24 – нормативи витрат, постачання матеріалів, потрібна потужність робочого центру та його фонду часу. Комірки C23:C25 – норми витрат матеріалу на одиниці продуктів. Комірка F22 – кількість матеріалу в партії постачання. Комірка F23 – кількість інтервалів у циклі постачання. Комірки I23:I25 – норми продуктивності робочого центру по продуктах. Комірка K22 містить фонд часу робочого центру в кожному інтервалі.

В комірках B28:K30 розраховані кількості витрат матеріалу на план виробництва, а в комірках B31:K31 – підсумок потреби цих витрат. У комірках B32:K32 накопичувальний підсумок цих витрат. В комірках B33:K33 – плановий запас матеріалу. В комірці B33 – значення початкового запасу матеріалу, яке визначає незмінний рівень запасу впродовж циклу постачання, після чого плановий рівень запасу зростає у кожному інтервалі на кількість матеріалу в партії постачання.

Розглянемо апробацію варіанту 0. Вхідні дані моделі наведені на рис. 3.

В комірках B36:K38 розташовані розраховані значення завантаження (в годинах) робочого центру по продуктах відповідно до плану виробництва. В комірках B39:K39 розрахований підсумок завантаження. В комірках B40:K40 розрахований накопичувальний підсумок завантаження. В комірках B41:K41 розрахований плановий накопичувальний фонд часу, який у кожному інтервалі зростає на величину його фонду часу.

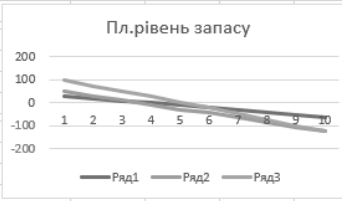
У комірках B44:K44 розташовані ваги цільової функції, а в B45:K45 – зважене завантаження робочого центру по інтервалах (добуток комірки з рядку 44 на час завантаження з рядку 39).

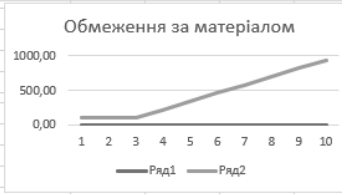
У комірці L44 цільова функція – зважена сума завантаження робочого центру по інтервалах.

У правій частині аркуша Excel розташовані графік планового падіння запасів, графік порівняння планової кількості витрат матеріалу з його плановим запасом, графік порівняння фонду часу з плановим часом завантаження робочого центру.

За допомогою Розв'язувача задач (Дані->Розв'язувач) можна визначити модель задачі лінійного програмування (рис. 4).

	A	B	C	D	E	F	G	H	I	J	K	L
1	**пл.продажів**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	
2	П1	10	12	7	10	13	7	12	10	11	9	
3	П2	19	20	22	17	15	21	19	22	18	20	
4	П3	26	22	23	27	19	27	29	27	23	22	
6	**падіння запасів**											
7	П1	**30**	20	8	1	-9	-22	-29	-41	-51	-62	62
8	П2	**50**	31	11	-11	-28	-43	-64	-83	-105	-123	123
9	П3	**100**	74	52	29	2	-17	-44	-73	-100	-123	123
11	**план виробництва**											
12	П1	0	0	0	0	0	0	0	0	0	0	0
13	П2	0	0	0	0	0	0	0	0	0	0	0
14	П3	0	0	0	0	0	0	0	0	0	0	0
17	**пл. рівень запасів**											
18	П1	**30**	20	8	1	-9	-22	-29	-41	-51	-62	
19	П2	**50**	31	11	-11	-28	-43	-64	-83	-105	-123	
20	П3	**100**	74	52	29	2	-17	-44	-73	-100	-123	
22	**нормативи**		**материал:**	пл.поповнення:		120		**робочий центр:**		фонд часу	90	
23	П1		1,2	цикл поповнення:		3			0,9			
24	П2		1,8						2,1			
25	П3		2,9						3,0			
27	**материал**											
28	П1	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	
29	П2	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	
30	П3	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	
31	всього	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	
32	накопичув.	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	
33	пл.запас	100,00	100,00	100,00	220,00	340,00	460,00	580,00	700,00	820,00	940,00	
35	**робочий центр**											
36	П1	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	
37	П2	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	**завантаження**
38	П3	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	**всього**
39	всього	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0
40	накопичув.	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	
41	фонд	90,00	180,00	270,00	360,00	450,00	540,00	630,00	720,00	810,00	900,00	
43	**к-ваги цільової функції**											
44		2	4	8	16	32	64	128	256	512	1024	взважене завантаження
45		0	0	0	0	0	0	0	0	0	0	0





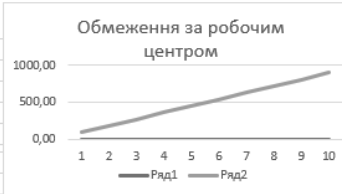


Рисунок 3 – Вхідні дані моделі варіанту 0 ([15])

DOI 10.15588/1607-3274-2024-2-17

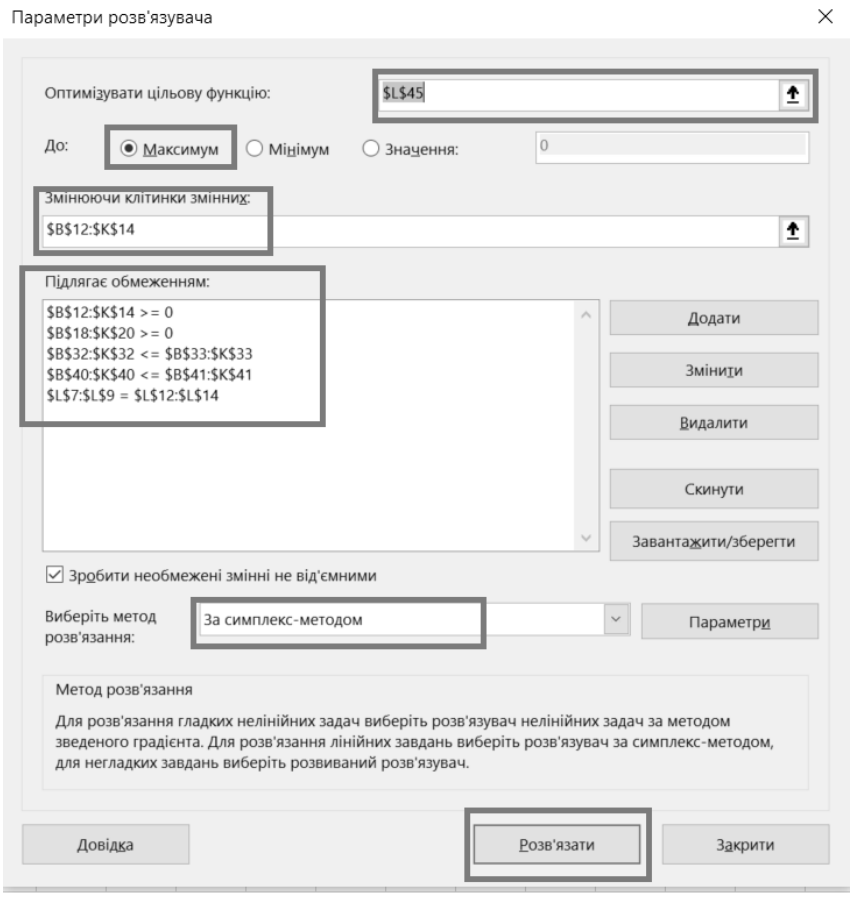


Рисунок 4 – Модель задачі варіанту 0

В такому вигляді задача сформульована в [15], лише критерій модифіковано, аби впливати на зсув плану раніше чи пізніше.

Результат розрахунку варіанту 0 MPS (зсув плану якнайпізніше) представлений на рис.8.

Після розв'язання задачі (кнопка «Розв'язати») отримуємо план поповнень запасів продуктів в комірках B12:K14.

Проведемо розрахунок плану виробництва для задачі варіанту 0 з вагами $k_t = 2^{(T-t+1)}$, $t = 1, \ldots, T$ (зсув плану якнайраніше). Для цього змінимо значення в комірках B44:K44 на рис. 3. Результат розрахунку MPS варіанту 0 зі зсувом плану якнайраніше – на рис. 9.

Апробація варіанту 1. Щоб провести апробацію моделі, яка відповідає варіанту 1, треба додати до обмежень моделі обмеження (7), (8). Для цього до вхідних даних додано комірки B48:K50 зі значеннями обмежень плану виробництва зверху (рис. 5). Змінена модель показана на рис. 6.

Апробація варіанту 2. Для проведення апробації моделі, яка відповідає варіанту 2, треба змінити обмеження моделі (8) на обмеження (9). Для цього введено дані в комірках B48:L50, які задають цикли поповнення запасу (рис. 7).

47	**обмеження зверху**										
48	П1	0	0	0	100000	100000	100000	100000	100000	100000	100000
49	П2	0	0	100000	100000	100000	100000	100000	100000	100000	100000
50	П3	0	0	0	0	100000	100000	100000	100000	100000	100000

Рисунок 5 – Вхідні дані моделі, які задають обмеження (8)

DOI 10.15588/1607-3274-2024-2-17

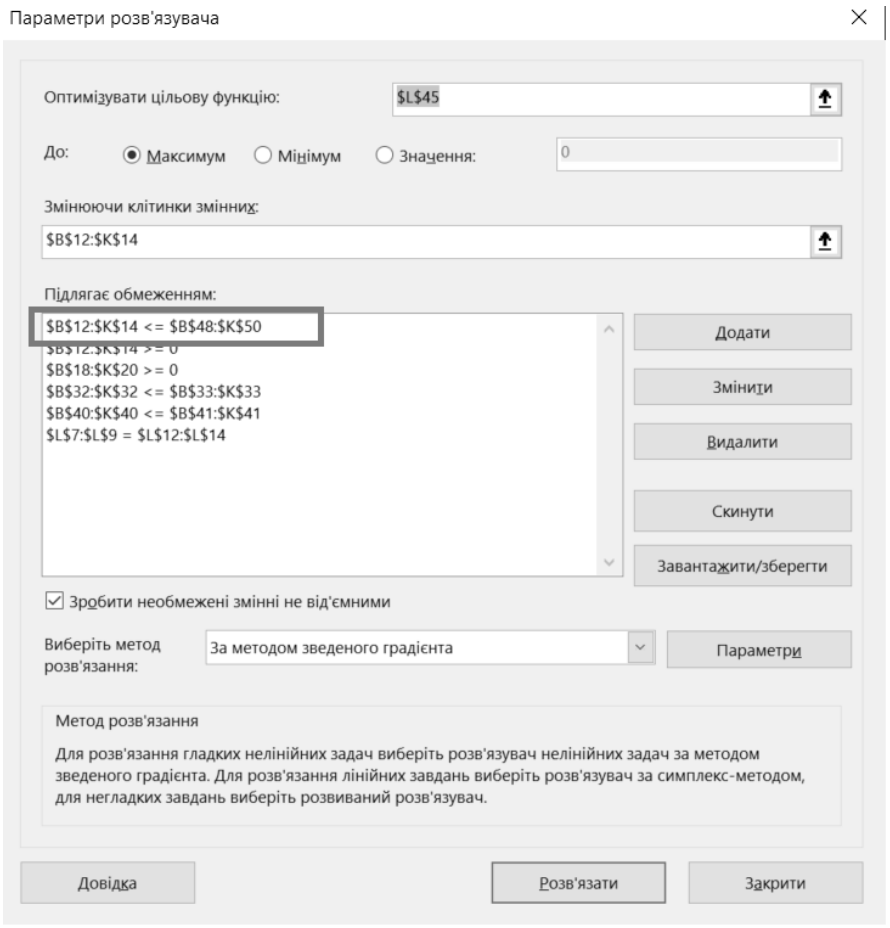


Рисунок 6 – Обмеження (7), яке додано в модель

47	обмеження зверху											цикли поповнення
48	П1	0	0	0	100000	0	0	100000	0	0	100000	2
49	П2	0	0	100000	0	0	100000	0	0	100000	0	3
50	П3	0	0	0	0	100000	0	0	100000	0	0	3
51												

Рисунок 7 – Вхідні дані моделі, які відповідають обмеженню (9) з періодичним поповненням

5 РЕЗУЛЬТАТИ

На рис. 8 надані результати розрахунку плану варіанту 0 зі зсувом його наскільки можливо пізніше.

Можна переконатись у дотриманні обмежень 1–5 з переліку обмежень з розділу 1:

1. B12:K14 >= 0;
2. B18:K20 >= 0;
3. B32:K32 <= B33:K33;
4. B40:K40 <= B41:K41;
5. L7:L9 = L12:L14.

Останні три обмеження представлені на діаграмах. Сумарні обсяги виробництва дорівнюють кінцевому дефіциту.

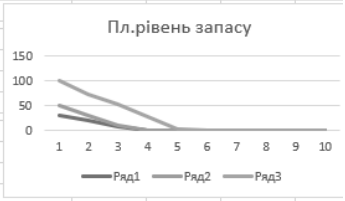
На рис. 9 представлені результати розрахунку плану варіанту 0 зі зсувом його наскільки можливо раніше. Як бачимо, поповнення продуктів передбачені з початку зони планування, якщо ресурси дозволяють (як для продукту П2), але з точки зору JIT/Lean це найбільший гріх. В той же час два останні інтервали взагалі не використовуються для виробництва, на відміну від плану на рис.8 (зсув плану якнайпізніше), де для виробництва не використовуються два перші інтервали.

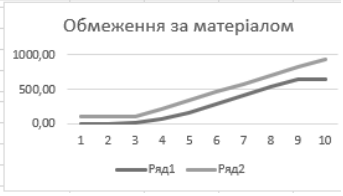
В останньому інтервалі виробництво не планується, бо у нього нема споживачів – плану продажу наступного інтервалу. Це виправиться само собою в разі «ковзаючого» планування, коли по завершенню інтервалу зона планування змішається на один інтервал.

На рис. 10 – результат розрахунку варіанту 1 плану (поповнення починаються до першого дефіциту).

DOI 10.15588/1607-3274-2024-2-17

	A	B	C	D	E	F	G	H	I	J	K	L
1	**пл.продажів**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	
2	П1	10	12	7	10	13	7	12	10	11	9	
3	П2	19	20	22	17	15	21	19	22	18	20	
4	П3	26	22	23	27	19	27	29	27	23	22	
6	**падіння запасів**											
7	П1	**30**	20	8	1	-9	-22	-29	-41	-51	-62	62
8	П2	**50**	31	11	-11	-28	-43	-64	-83	-105	-123	123
9	П3	**100**	74	52	29	2	-17	-44	-73	-100	-123	123
11	**план виробництва**											
12	П1	0	0	0	9	13	7	12	10	11	0	62
13	П2	0	0	11	17	15	21	19	22	18	0	123
14	П3	0	0	0	0	17	27	29	27	23	0	123
17	**пл. рівень запасів**											
18	П1	**30**	20	8	1	0	0	0	0	0	0	
19	П2	**50**	31	11	0	0	0	0	0	0	0	
20	П3	**100**	74	52	29	2	0	0	0	0	0	
22	**нормативи**		**материал:**	пл.поповнення:		120		**робочий центр:**		фонд часу	90	
23	П1		1,2	цикл поповнення:		3			0,9			
24	П2		1,8						2,1			
25	П3		2,9						3,0			
27	**материал**											
28	П1	0,00	0,00	0,00	10,80	15,60	8,40	14,40	12,00	13,20	0,00	
29	П2	0,00	0,00	19,80	30,60	27,00	37,80	34,20	39,60	32,40	0,00	
30	П3	0,00	0,00	0,00	0,00	49,30	78,30	84,10	78,30	66,70	0,00	
31	всього	0,00	0,00	19,80	41,40	91,90	124,50	132,70	129,90	112,30	0,00	
32	накопичув.	0,00	0,00	19,80	61,20	153,10	277,60	410,30	540,20	652,50	652,50	
33	пл.запас	100,00	100,00	100,00	220,00	340,00	460,00	580,00	700,00	820,00	940,00	
35	**робочий центр**											
36	П1	0,00	0,00	0,00	8,10	11,70	6,30	10,80	9,00	9,90	0,00	
37	П2	0,00	0,00	23,10	35,70	31,50	44,10	39,90	46,20	37,80	0,00	**завантаження**
38	П3	0,00	0,00	0,00	0,00	51,00	81,00	87,00	81,00	69,00	0,00	**всього**
39	всього	0,00	0,00	23,10	43,80	94,20	131,40	137,70	136,20	116,70	0,00	683,1
40	накопичув.	0,00	0,00	23,10	66,90	161,10	292,50	430,20	566,40	683,10	683,10	
41	фонд	90,00	180,00	270,00	360,00	450,00	540,00	630,00	720,00	810,00	900,00	
43	**k-ваги цільової функції**											
44		2	4	8	16	32	64	128	256	512	1024	взважене завантаження
45		0	0	184,8	700,8	3014,4	8409,6	17625,6	34867,2	59750,4	0	124553





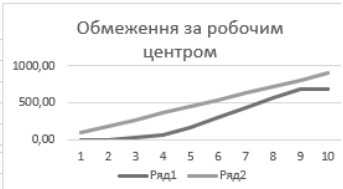
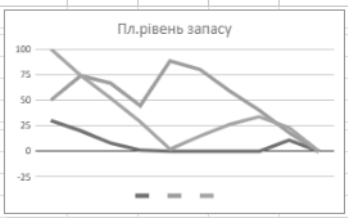
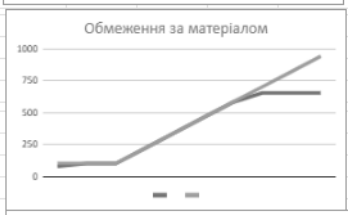


Рисунок 8 – Результати розв’язання задачі варіанту 0 з вагами $k_t = 2^t$, $t = 1, \ldots, T$ (зсув плану якнайпізніше)

	A	B	C	D	E	F	G	H	I	J	K	L
4	П3	26	22	23	27	19	27	29	27	23	22	
6	**падіння запасів**											
7	П1	**30**	20	8	1	-9	-22	-29	-41	-51	-62	62
8	П2	**50**	31	11	-11	-28	-43	-64	-83	-105	-123	123
9	П3	**100**	74	52	29	2	-17	-44	-73	-100	-123	123
11	**план виробництва**											
12	П1	0	0	0	9	13	7	12	21	0	0	62
13	П2	43	13	0	61	7	0	0	0	0	0	123
14	П3	0	0	0	0	32	38	36	16	0	0	123
17	**пл. рівень запасів**											
18	П1	**30**	20	8	1	-0	-0	-0	-0	11	-0	
19	П2	**50**	74	67	45	88	80	59	40	18	0	
20	П3	**100**	74	52	29	2	15	26	34	23	0	
22	**нормативи**		**материал**	пл.поповнення:		120		**робочий центр:**		фонд час	90	
23	П1		1,2	цикл поповнення:		3			0,9			
24	П2		1,8						2,1			
25	П3		2,9						3,0			
27	**материал**											
28	П1	0,00	0,00	0,00	10,80	15,60	8,40	14,40	25,20	0,00	0,00	
29	П2	77,14	22,86	0,00	109,20	12,20	0,00	0,00	0,00	0,00	0,00	
30	П3	0,00	0,00	0,00	0,00	92,20	111,60	105,60	47,30	0,00	0,00	
31	всього	77,14	22,86	0,00	120,00	120,00	120,00	120,00	72,50	0,00	0,00	
32	накопичув.	77,14	100,00	100,00	220,00	340,00	460,00	580,00	652,50	652,50	652,50	
33	пл.запас	100,00	100,00	100,00	220,00	340,00	460,00	580,00	700,00	820,00	940,00	
35	**робочий центр**											
36	П1	0,00	0,00	0,00	8,10	11,70	6,30	10,80	18,90	0,00	0,00	
37	П2	90,00	26,67	0,00	127,40	14,23	0,00	0,00	0,00	0,00	0,00	**завантаження**
38	П3	0,00	0,00	0,00	0,00	95,38	115,45	109,24	48,93	0,00	0,00	**всього**
39	всього	90,00	26,67	0,00	135,50	121,31	121,75	120,04	67,83	0,00	0,00	683,10
40	накопичув.	90,00	116,67	116,67	252,17	373,48	495,23	615,27	683,10	683,10	683,10	
41	фонд	90,00	180,00	270,00	360,00	450,00	540,00	630,00	720,00	810,00	900,00	
43	**k-ваги цільової функції**											
44		1024	512	256	128	64	32	16	8	4	2	взважене завантаження
45		92160	13653,333	0	17344	7764,009	3895,944	1920,662	542,6482	0	0	137280,5





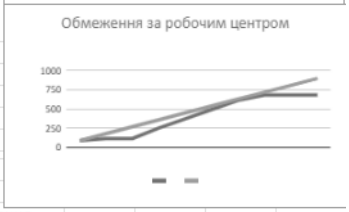


Рисунок 9 – Результати задачі варіанту 0 з вагами $k_t = 2^{T-t+1}$, $t = 1, \ldots, T$ (зсув плану якнайраніше)

DOI 10.15588/1607-3274-2024-2-17

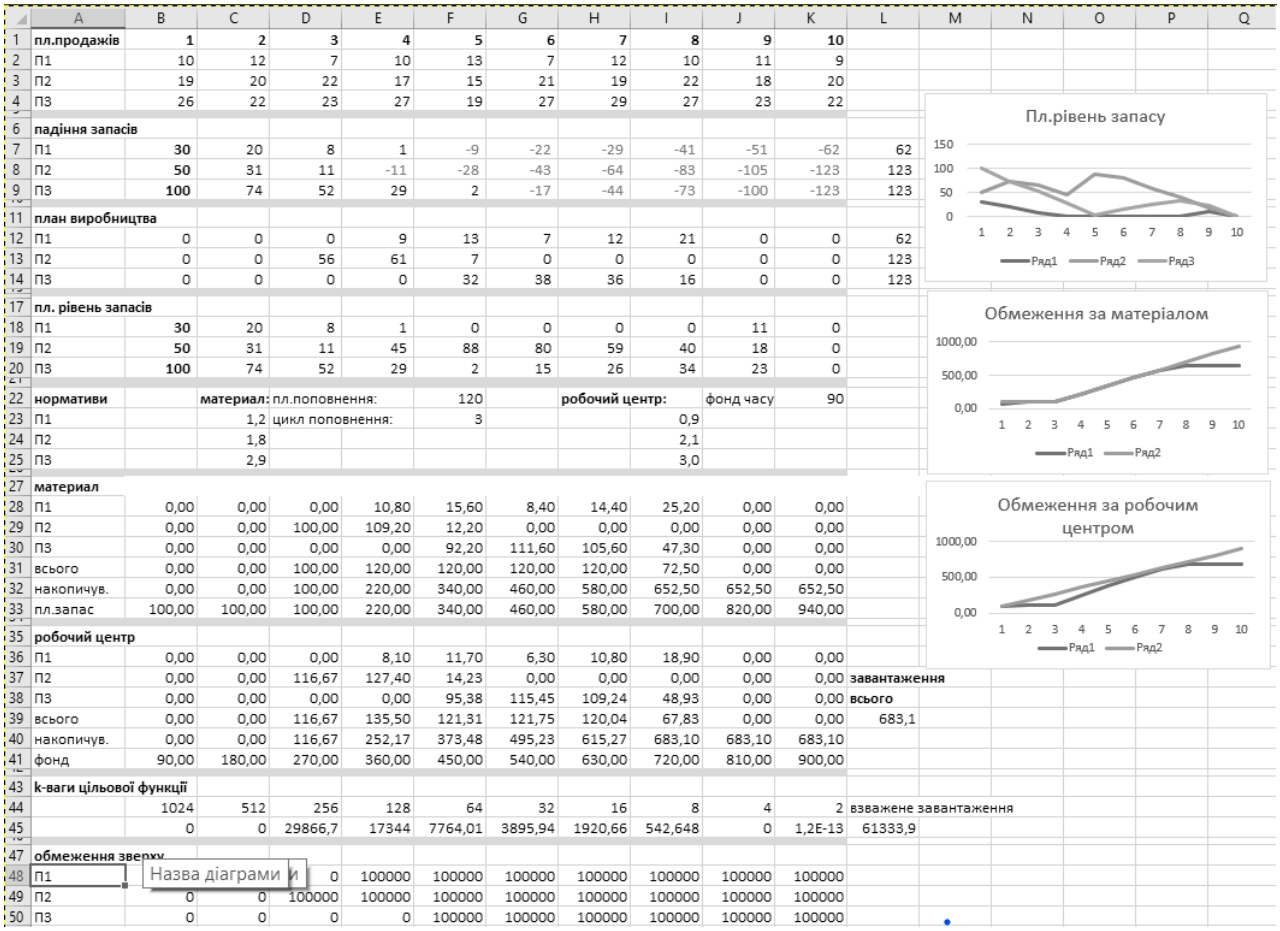


Рисунок 10 – Результати розв’язання задачі варіанту 1

Як бачимо, всі ненульові значення поповнень лежать раніше інтервалів першого дефіциту. Інші обмеження задачі (обмеження 1–5) теж не порушені.

Результати розрахунку варіанту плану 2 з періодичним поповненням представлені на рис. 11.

Як видно, обмеження задачі (обмеження 1–5) не порушені. Останні три обмеження представлені на графіках. Сумарні об’єми виробництва дорівнюють кінцевому дефіциту. План обрахований при вагах цільової функції $k_t = 2(T–t+1)$, $t = 1, \ldots, T$, що вимагає розміщати поповнення як можна раніше, але всі ненульові значення поповнень лежать в дозволених періодичних інтервалах.

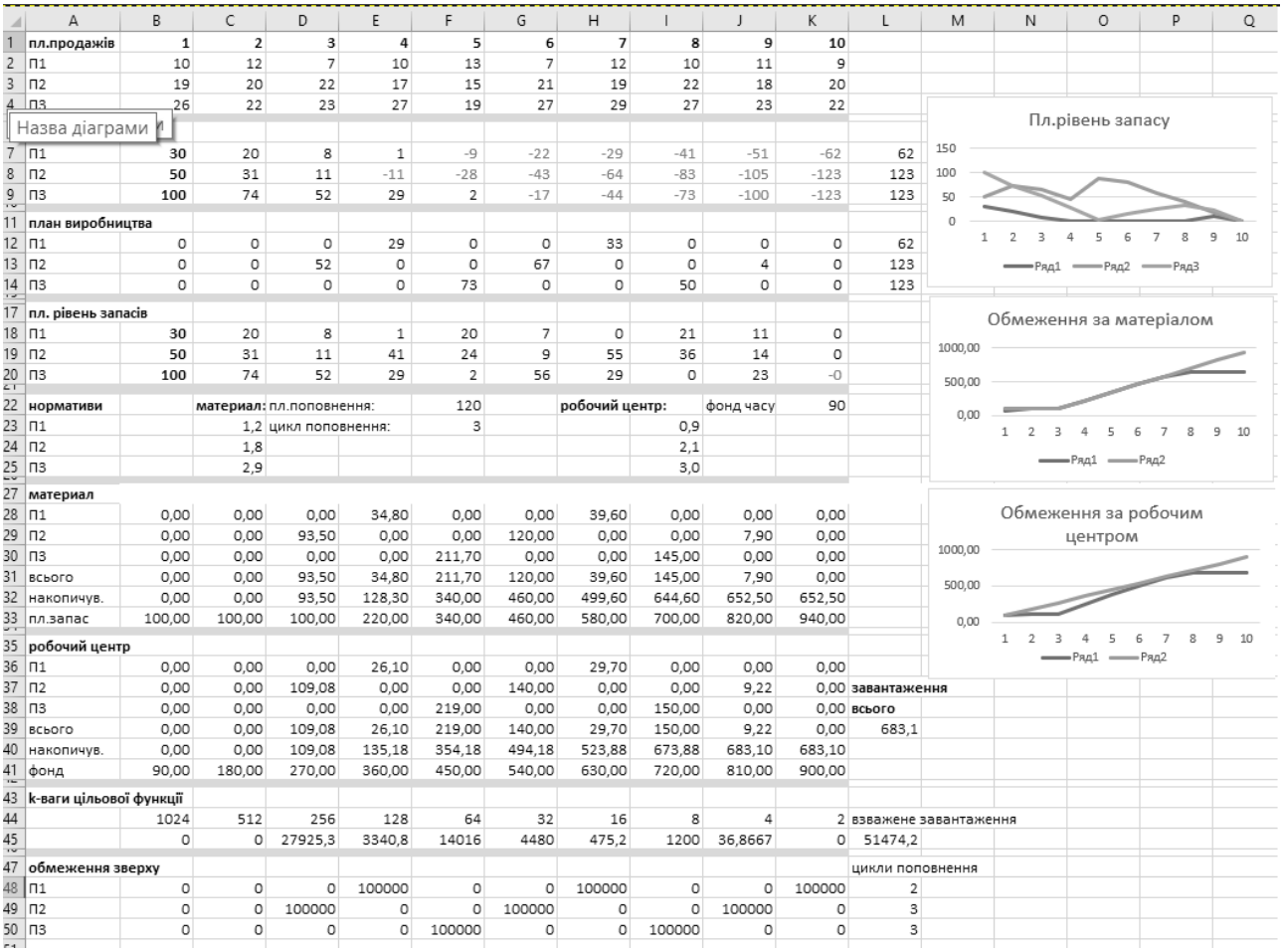
6 ОБГОВОРЕННЯ

Наявний план продажів залишає мало варіантів для варіацій MPS. У межах наявних степеней вільності результат розрахунку реагує на «тиск» цільової функції в сторону пізнішого (рис. 8) або ранішого (рис. 9) виробництва, що видно по рядках 12–14 розрахованого плану, особливо на продукті П2. Відповідно до сформульованої постановки задачі всі варіанти плану, які задовольняють обмеження, економічно рівнозначні. Симплекс-метод використовується для пошуку припустимого рішення в умовах обмежень. Аби врахувати часову вартість більш раннього виробництва продуктів, треба припустити, що покупець виплачує виробнику певну премію (надбавку) за більш раннє відвантаження продукції.

Проведена апробація розрахунків головного календарного плану виробництва не обмежується наведеними трьома варіантами. Більш привабливою здається ідея розрахунків з так званими динамічними точками поповнення запасів, коли починаючі з першої точки поповнення планується випуск продуктів з максимальним використанням виробничих ресурсів – по принципу «стільки, скільки можливо»; потім розраховується падіння запасів, виникає наступна точка поповнення і так далі. Але це є темою подальших досліджень. У процесі планування на підприємстві наступним кроком має бути перетворення плану поповнень продукції («виробляти розраховану кількість не пізніше розрахованих інтервалів») у план виробництва («виробляти розраховану кількість у розрахованих інтервалах»). Основою такого плану має бути розрахований погодинний розклад завантаження робочих центрів з урахуванням такого значного фактору як час переналадки обладнання.

Розглянутий приклад у вигляді xlsx-файлу доступний за посиланням https://docs.google.com/spreadsheets/d/1feEKz4oZQAoncibYymN0YVtk6r8DYJQb/edit?usp=sharing&ouid=114728324013759458622&rtpof=true&sd=true .

DOI 10.15588/1607-3274-2024-2-17



	A	B	C	D	E	F	G	H	I	J	K	L
1	**пл.продажів**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	
2	П1	10	12	7	10	13	7	12	10	11	9	
3	П2	19	20	22	17	15	21	19	22	18	20	
4	П3	26	22	23	27	19	27	29	27	23	22	
7	П1	**30**	20	8	1	-9	-22	-29	-41	-51	-62	62
8	П2	**50**	31	11	-11	-28	-43	-64	-83	-105	-123	123
9	П3	**100**	74	52	29	2	-17	-44	-73	-100	-123	123
11	**план виробництва**											
12	П1	0	0	0	29	0	0	33	0	0	0	62
13	П2	0	0	52	0	0	67	0	0	4	0	123
14	П3	0	0	0	0	73	0	0	50	0	0	123
17	**пл. рівень запасів**											
18	П1	**30**	20	8	1	20	7	0	21	11	0	
19	П2	**50**	31	11	41	24	9	55	36	14	0	
20	П3	**100**	74	52	29	2	56	29	0	23	-0	
22	**нормативи**		**материал:**	пл.поповнення:		120		**робочий центр:**		фонд часу	90	
23	П1		1,2	цикл поповнення:		3			0,9			
24	П2		1,8						2,1			
25	П3		2,9						3,0			
27	**материал**											
28	П1	0,00	0,00	0,00	34,80	0,00	0,00	39,60	0,00	0,00	0,00	
29	П2	0,00	0,00	93,50	0,00	0,00	120,00	0,00	0,00	7,90	0,00	
30	П3	0,00	0,00	0,00	0,00	211,70	0,00	0,00	145,00	0,00	0,00	
31	всього	0,00	0,00	93,50	34,80	211,70	120,00	39,60	145,00	7,90	0,00	
32	накопичув.	0,00	0,00	93,50	128,30	340,00	460,00	499,60	644,60	652,50	652,50	
33	пл.запас	100,00	100,00	100,00	220,00	340,00	460,00	580,00	700,00	820,00	940,00	
35	**робочий центр**											
36	П1	0,00	0,00	0,00	26,10	0,00	0,00	29,70	0,00	0,00	0,00	
37	П2	0,00	0,00	109,08	0,00	0,00	140,00	0,00	0,00	9,22	0,00	**завантаження**
38	П3	0,00	0,00	0,00	0,00	219,00	0,00	0,00	150,00	0,00	0,00	**всього**
39	всього	0,00	0,00	109,08	26,10	219,00	140,00	29,70	150,00	9,22	0,00	683,1
40	накопичув.	0,00	0,00	109,08	135,18	354,18	494,18	523,88	673,88	683,10	683,10	
41	фонд	90,00	180,00	270,00	360,00	450,00	540,00	630,00	720,00	810,00	900,00	
43	**k-ваги цільової функції**											
44		1024	512	256	128	64	32	16	8	4	2	**взважене завантаження**
45		0	0	27925,3	3340,8	14016	4480	475,2	1200	36,8667	0	51474,2
47	**обмеження зверху**											**цикли поповнення**
48	П1	0	0	0	100000	0	0	100000	0	0	100000	2
49	П2	0	0	100000	0	0	100000	0	0	100000	0	3
50	П3	0	0	0	0	100000	0	0	100000	0	0	3

Рисунок 11 – Результати розв’язання задачі варіанту 2

ВИСНОВКИ

Запропоновано вдосконалення стандарту планування ресурсів підприємства MRP II, а саме розрахунку плану MPS.

Наукова новизна результатів: застосування лінійного програмування в алгоритмі, який традиційно має характер прямого послідовного розрахунку, дозволяє об’єднати два його кроки (генерація плану і перевірка його припустимості) в один, при цьому генерується MPS, який одразу задовольняє обмеження виробничої потужності і доступності матеріалів, якщо це в принципі можливо.

Практична значимість результатів полягає в тому, що запропонована модифікація алгоритму спрощує процес створення MPS, що важливо з огляду на високі вимоги до організаційної спроможності підприємства, які висуває стандарт MRP II.

Напрямками подальших досліджень ми бачимо включення в процедуру створення MPS розрахунку денного або змінного розкладу обробки партій продукції робочими центрами з урахуванням часу переналадок, який часто є суттєвим фактором.

ПОДЯКИ

Робота виконана в рамках ініціативної наукової теми Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського» «Створення гібридної обчислювальної технології побудови квазі-формалізованої моделі прогнозування в умовах неоднорідності даних та ненормативних відхилень в системах організаційного управління» (державний реєстраційний номер 0117U002448). Автори висловлюють щиру подяку колективу київського заводу «ФАРМАК», який у свій час підтримав одного з авторів у впровадженні на підприємстві стандарту управління MRP II.

ЛІТЕРАТУРА

1. APICS Dictionary, 16th Edition. [Electronic resource]. – Access mode: https://www.ascm.org/apics-dictionary-16th-edition/ (appeal date: 09.05.2024) – Title from screen.
2. Landvater D. MRP II Standard System. A Handbook for Manufacturing Software Survival / D. Landvater, G. Christopher. – John Wiley & Sons Inc., 1989. – 315 p.
3. Browne J. Production Management Systems: An Integrated Perspective / J. Browne, J. Harhen, J. Shivnan. – Boston : Addison-Wesley, 1996. – 425 p.
4. Goldratt E. What is this Thing Called Theory of Constraints and how Should it be Implemented? / E. Goldratt. – North River Press, 1990. – 162 p.

DOI 10.15588/1607-3274-2024-2-17

5. Schragenheim E. Manufacturing at Warp Speed. Optimizing Supply Chain Financial Performance / E. Schragenheim, W. Dettmer. – London, New York, Washington : CRC Press, 2001. – 333 p.
6. Ohno Taiichi Just-In-Time for Today and Tomorrow / Taiichi Ohno, Setsuo Mito. – Productivity Press, 1988. – 145 p.
7. Yasuhiro M. Toyota Production System, An Integrated Approach to Just-In-Time / M. Yasuhiro. – Springer Science & Business Media, 2012. – 424 p.
8. Modeling of the Master Production Schedule for the Digital Transition of Manufacturing SMEs in the Context of Industry 4.0 / [E. Tobon-Valencia, S. Lamouri, R. Pellerin et al.] // Sustainability. – 2022. – № 14. – P. 12562. Access mode: https://doi.org/10.3390/su141912562
9. Setting MRP Parameters and Optimizing the Production Planning Process / [M. Malindzakova, P. Garaj, J. Trpˇcevská et al.] // Processes. – 2022. – № 10. – P. 690. Access mode: https://doi.org/10.3390/pr10040690
10. Ptak C. A. Orlicky's Material Requirements Planning, Third Edition / C. A. Ptak, C. Smith. – McGraw Hill Professional, 2011. – 352 p.
11. Demand Driven Material Requirements Planning (DDMRP): A systematic review and classification / [A. Azzamouri, P. Baptiste, G. Dessevre et al.] // Journal of Industrial Engineering and Management. – 2021. – № 14(3). – P. 439–456. Access mode: https://doi.org/10.3926/jiem.3331
12. Orue A. Demand Driven MRP – The need to standardise an implementation process / A. Orue, A. Lizarralde, A. Kortabarria // International Journal of Production Management and Engineering. – 2020. –Vol. 8, Issue 2. – P. 65–73. Access mode: https://doi.org/10.4995/ijpme.2020.12737.
13. An empirical comparison of MRPII and Demand-Driven MRP / [R. Miclo, F. Fontanili, M. Lauras et al.] // IFAC-PapersOnLine. – 2016. – Vol. 49. – Issue 12. – P. 1725–1730.
14. Effective Prototyping with Excel: A Practical Handbook for Developers and Designers / [N. Berger, M. Arent, J. Arnovitz, F. Sampson]. – Elsevier, 2009. – 240 p.
15. Новінський В. П. Застосування методу лінійного програмування в процедурах планування MRP II / В. П. Новінський, В. Д. Попенко // Економічний простір. – 2024. – № 189. – С. 196–206. Режим доступу: https://doi.org/10.32782/2224-6282/189-36

Стаття надійшла до редакції 12.03.2024.
Після доробки 10.05.2024.

UDC 658.5

FORMALIZATION OF THE MASTER PRODUCTION SHEDULE FORMATION TASK IN THE MRP II PLANNING SYSTEM

Novinskyi V. P. – PhD, Associate Professor of the Department of Informatics and Software Engineering, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine.

Popenko V. D. – PhD, Associate Professor of the Department of Information Systems and Technologies, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine.

ABSTRACT

Context. Considered the task of forming the Master Production Shedule in the process of production management based on the MRP II standard. The object of the study is the algorithm for forming this plan for further planning of materials supply for production and the organization of production itself.

Objective. Improvement of the algorithm of Master Production Shedule formation to avoid unnecessary stages of the algorithm.

Method. It is proposed to improve the algorithm of the Master Production Shedule formation. It consists in simultaneously taking into account the requirements for timely delivery of products to customers, limitations regarding the capacities of the company's work centers, and limitations regarding the duration of procurement cycles in the process of supplying materials. The MRP II standard envisages first planning the terms and quantity of product releases, and only at the next step checking the formed plan for admissibility with regard to the required time of operation of the equipment and the availability of the required materials quantity. In case of the calculated plan limitations violation, it is necessary to either plan and implement measures to overcome the specified limitations, i.e. organize additional shifts for work centers, use additional capacities, speed up the delivery of some materials, or reduce the sales plan. All these measures are associated with additional costs. In the proposed version of the planning process, this should be done only if the algorithm does not find an acceptable solution. The task of forming the Master Production Shedule, which is central to the MRP standard, is formulated by the authors as a linear programming task due to the linear nature of the specified restrictions on production capacities and materials. In particular, in the case of sufficiently severe restrictions on the work centers capacity, the plan for replenishing the remaining products from production is shifted to earlier planning intervals and only then rests against the restrictions. Several strategies are proposed for planning replenishments from the production of products stock.

Results. The developed algorithms are implemented in the form of Microsoft Excel templates and are available for use in order to deepen the understanding of the MRP II standard. They are also used in the educational process.

Conclusions. Approbation of the solution by the authors confirmed its workability, as well as the expediency of implementing the developed modification of the MRP II planning process into the software of leading ERP class systems suppliers. Prospects for further research may consist in a comparative analysis of the proposed options for placement of products replenishment from production, through economic evaluation of these options, as well as through simulation modeling.

KEYWORDS: planning, production, materials, products, intermediate products, MRP II.

DOI 10.15588/1607-3274-2024-2-17

REFERENCES

1. APICS Dictionary, 16th Edition. [Electronic resource]. Access mode: https://www.ascm.org/apics-dictionary-16th-edition/ (appeal date: 09.05.2024). Title from screen.
2. Landvater D., Christopher G. MRP II Standard System. A Handbook for Manufacturing Software Survival. John Wiley & Sons Inc., 1989, 15 p.
3. Browne J., Harhen J., Shivnan J. Production Management Systems: An Integrated Perspective. Boston, Addison-Wesley, 1996, 425 p.
4. Goldratt E. What is this Thing Called Theory of Constraints and how Should it be Implemented? North River Press, 1990, 162 p.
5. Schragenheim E., Dettmer W. Manufacturing at Warp Speed. Optimizing Supply Chain Financial Performance. London, New York, Washington, CRC Press, 2001, 333 p.
6. Ohno Taiichi, Mito Setsuo Just-In-Time for Today and Tomorrow. Productivity Press, 1988, 145 p.
7. Yasuhiro M. Toyota Production System, An Integrated Approach to Just-In-Time. Springer Science & Business Media, 2012, 424 p.
8. Tobon-Valencia E., Lamouri S., Pellerin R. et al. Modeling of the Master Production Schedule for the Digital Transition of Manufacturing SMEs in the Context of Industry 4.0, *Sustainability*, 2022, № 14, P. 12562. Access mode: https://doi.org/10.3390/su141912562
9. Malindzakova M., Garaj P., Trpˇcevská J. et al. Setting MRP Parameters and Optimizing the Production Planning Process, *Processes*, 2022, № 10, P. 690. Access mode: [https://doi.org/10.3390/pr10040690
10. Ptak C. A., Smith C. Orlicky's Material Requirements Planning, Third Edition. McGraw Hill Professional, 2011, 352 p.
11. Azzamouri A., Baptiste P., Dessevre G. et al. Demand Driven Material Requirements Planning (DDMRP): A systematic review and classification, *Journal of Industrial Engineering and Management*, 2021, № 14(3), pp. 439–456. Access mode: https://doi.org/10.3926/jiem.3331
12. Orue A., Lizarralde A., Kortabarria A. Demand Driven MRP – The need to standardise an implementation process, *International Journal of Production Management and Engineering*, 2020, Vol. 8, Issue 2, pp. 65–73. Access mode: https://doi.org/10.4995/ijpme.2020.12737.
13. Miclo R., Fontanili F., Lauras M. et al. An empirical comparison of MRPII and Demand-Driven MRP, *IFAC-PapersOnLine*, 2016, Vol. 49, Issue 12, pp. 1725–1730.
14. Berger N., Arent M., Arnovitz J., Sampson F. Effective Prototyping with Excel: A Practical Handbook for Developers and Designers. Elsevier, 2009, 240 p.
15. Novinsky V. P., Popenko V. D. Zastosuvannja metodu linijnogo programuvannja v procedurah planuvannja MRP II, *Ekonomichnyj prostir*, 2024, № 189, pp. 196–206. Rezhym dostupu: https://doi.org/10.32782/2224-6282/189-36

DOI 10.15588/1607-3274-2024-2-17

UDC 681.5: 673.1

DEVELOPMENT OF AUTOMATED CONTROL SYSTEM FOR CONTINUOUS CASTING

Sotnik S. V. – PhD, Associate Professor, Associate Professor of Department of Computer-Integrated Technologies, Automation and Robotics, Kharkiv, Ukraine.

ABSTRACT

Context. Today, automated continuous casting control systems are developing rapidly, as process of manufacturing billets (products) of same size from metal in casting mold in mass production has long been outdated and "continuous casting stage" is coming. This process is suitable for non-ferrous metals and steel. However, each time during development, task of improving quality of resulting billet arises, which directly depends on optimizing efficiency and reliability of automated systems themselves. Optimization is key stage in development process, as it is aimed at ensuring accuracy and stability of casting process, which includes development of parametric model and accurate algorithms that ensure optimal temperature, metal pouring rate, oscillation frequency, oscillation amplitude, metal level in crystallizer, and position of position of industrial bucket stopper for each casting stage. In particular, this problem has not yet been fully solved in literature known to authors, so it is necessary to formulate problem and develop algorithm for system operation for specific safety casting unit.

Objective. The aim of study is to develop automated control system to ensure accuracy and stability of casting process.

Method. The developed control system for continuous casting plant is based on proposed parametric model, which is formalized on basis of set theory. The model takes into account key parameters for particular casting process: metal pouring rate, oscillation frequency, oscillation amplitude, metal level in crystallizer, and position of industrial bucket stopper.

Results. The problem was formulated and key parameters were determined, which are taken into account in system's algorithm, which made it possible to develop control system for continuous casting plant to solve problem of improving quality of resulting billet.

Conclusions. A parametric model and generalized black box model representation were created, which are necessary for both new continuous casting projects and existing units to optimize metal casting process. To set up continuous casting system, controlled parameters such as pouring speed, oscillation frequency and amplitude, metal level in crystallizer, and position of industrial bucket stopper were determined. The algorithm of control system for continuous casting plant was developed, on basis of which system was developed that allows monitoring, regulation and control of obtaining steel process or non-ferrous billets. The developed user interface of control system is simple and easy to use.

KEYWORDS: process, continuous casting, automation, system, control.

ABBREVIATIONS

CC is a continuous casting;

CCM is a continuous casting machines.

NOMENCLATURE

A_{rc} is a vibration amplitude of crystallizer;

D_c is a distance between opposite walls of crystallizer;

F_{cc} is a cross-sectional area of crystallizer cavity;

F_{rc} is a frequency of crystallizer oscillation;

G_{cc} is a geometric parameters of crystallizer;

H_c is a height of crystallizer;

H_{Tecp} is a temperature at end of curing process;

I is a number of mold types, for example, parallel-walled molds or straight or reverse tapers for metal casting;

M_{lv} is a metal level in crystallizer;

P is a set of casting parameters;

P_{ib} is a stopper position industrial bucket;

SH^i_{cr} is a shape of crystallizer;

T is a logically ordered set of parameters required for casting;

T_{cc} is a temperature conditions;

T_{lq} is a liquidus temperature (at which first crystal falls out under equilibrium conditions);

T^s_{il} is a metal temperature in intermediate ladle;

T^s_{sl} is a metal temperature in steel ladle;

u_o is a speed of liquid metal with which it meets melt surface;

V_p is a speed of metal pouring;

W_o is a metal velocity at outlet of pouring hole;

Π is a set of all possible options for casting parameters.

INTRODUCTION

Today, problem of controlling continuous casting is relevant and key to improving production processes. The main scientific task is to develop efficient automated control system that will ensure accuracy, stability and optimal conditions during metal casting. Automation requires solving technical, engineering, and algorithmic challenges associated with complexity of process and its interaction with variable conditions.

This issue is important because it determines quality and productivity of casting process, which in turn affects quality of resulting billet. Automation of control in molding can lead to significant increase in production efficiency, resource savings, and cost reduction.

The scientific tasks include development of parametric model of CC installation and system operation algorithm, which together will ensure optimal casting conditions. Practical tasks include creation of functional and reliable system that can be implemented in industrial production.

DOI 10.15588/1607-3274-2024-2-18

The development of topic is based on analysis of current state of continuous casting production, existing problems and shortcomings in control systems. The initial data include results of preliminary research, technical requirements and specifications for creation of automated control system [1–3].

The research is critical to solving problems and achieving new levels of efficiency in field of CC. The development of automated control system not only optimizes production process, but also makes significant contribution to development of industry, improving product quality and rational use of resources.

The object of research is control process of plant for continuous casting of steel or non-ferrous metal billets.

The subject of research is continuous casting machines.

The purpose of the research is development of automated control system to ensure accuracy and stability of casting process.

1 PROBLEM STATEMENT

This paper solves problem of optimizing continuous casting process by developing automated control system. The mathematical formulation includes creation of parametric model that takes into account peculiarity of crystallizer to improve quality of produced slabs.

The input data for system includes parameters of casting process, as well as configuration of new or existing plants.

It is planned to obtain optimal values of parameter values, which will ensure high quality of manufactured billets.

Let's assume that proposed parametric model is necessary to determine key parameters for particular casting process and formalize them, which is necessary both when designing new CCMs and for already operating units to optimize casting process.

The input parameters in development of automated control system for CC plant are represented by model: $V_p=<W_o, u_o>$, $G_{cc}=<SH^i_{cr}, H_c, F_{cc}, D_c>$, $i = 1, 2, ..., I$. $T_{cc}=<T^s_{il}, T^s_{sl}, H_{Tecp}, T_{lq}>$.

As result, it is planned to take into account key elements of parametric model during development of automated system, which will further improve product quality through accurate and stable control of these casting parameters.

2 REVIEW OF THE LITERATURE

The process of producing products by casting is focus of many scientific papers [4–6], and continuous casting of metals is described in detail in [7, 8]. Continuous casting is main process of steel production today [6].

In recent years, growing complexity and global competition to improve quality of metal products have led to need to introduce new approaches to monitoring and controlling continuous steel casting process, as evidenced by work of authors [9–11].

A scheme for coordinating process of steelmaking and continuous casting is presented in [8]. The authors reviewed methods for studying planning of iron and steel production. The correlation between productivity of continuous casting process and its equipment is considered.

The integrated application of several algorithms based on operations research, heuristic algorithms, and artificial intelligence methods is effective means of solving such complex production planning problems. However, work [8] does not identify specific technological parameters, challenges, or limitations that iron and steel producers may face when implementing proposed methods.

In [9], authors present online monitoring system based on Internet of Things for continuous steel casting, which consists of four layers: sensing, network, service resources, and application layers. It integrates various data processing techniques, including protocol conversion, data filtering, and data transformation. Although authors describe how proposed system was implemented and demonstrated on real continuous casting line, paper would be more informative if mathematical framework on basis of which implementation was realized was provided.

Online monitoring of steel casting processes using multivariate statistics technologies is presented in [10]. The authors paid special attention to development of new scheme for synchronizing technological trajectories to monitor specific transient operations, such as equipment replacement or steel grade change, i.e., work is aimed at increasing productivity and reducing maintenance costs, and problem of improving quality of resulting products is not particularly addressed in paper.

Paper [11] investigates unsteady states of continuous steel casting process using industrial diagnostic systems process diagnostic system, special measuring devices, and thermal numerical model. The authors present mathematical component: thermal model based on Fourier-Kirchhoff equation. The authors graphically present study of vertical temperature profiles in mold and analyze measured dependencies of heat transfer coefficient on surface temperature, but more information about system hardware itself: sensors, devices, and data analysis methods could be provided, which could improve understanding of work.

The control and design of continuous steel casting process based on modern numerical models is described in detail in [12]. The authors described method of determining boundary conditions, initial conditions, and material parameters as most important components that provide numerical calculations based on them in finite element method, which is used most often and is important element of work. However, work does not take into account design features of foundry.

In source [13] describes modern automated system for monitoring crystallizer, which aims to improve product quality, increase process stability and increase casting speed. The source presents main functions of system and provides data that is constantly monitored and recorded.

Overall, these works make important contribution to field of continuous casting.

DOI 10.15588/1607-3274-2024-2-18

The general trend of these scientific papers is great interest in development and improvement of monitoring and control systems for continuous casting processes. They highlight important aspects such as impact of complexity and global competition on need for new control and management methods, integration of IoT technologies, and use of multivariate statistical technologies and numerical models.

Thus, this paper emphasizes regulation and control of most important parameters of CC installation, which are formalized using parametric model, which will undoubtedly be key to improving quality of products.

3 MATERIALS AND METHODS

3.1 Selection of key parameters of CC

In this study, we will not discuss all parameters, but will focus on CC installation.

In this paper, crystallizer will be considered as moulding unit, since it is one of most important functional units that determine rational operation of CCM.

In order to select CCM, it is necessary to determine type of cast billet:

1) machine for producing rectangular billets (slabs) used in production of rolled steel sheets and strips – slab machine [14];

2) machine for producing square billets (blooms) used in production of rolled plates and strips – sizing machine [15, 16].

Thus, slab continuous caster was chosen because company plans to produce slabs. A curved strip mill was chosen because they have high unit capacity and relatively low installation height.

Fig. 1 summarizes scheme of slab continuous caster, which consists of: 1 – intermediate ladle, 2 – turntable, 3 – copper liner, 4 – crystallizer (first cooling level), 5 – spray cooling with water (second cooling level), 6 – pulling and straightening unit, 7 – roller conveyor, 8 – gas cutter [1].

We'll focus on crystallizer because it's where workpiece is directly shaped.

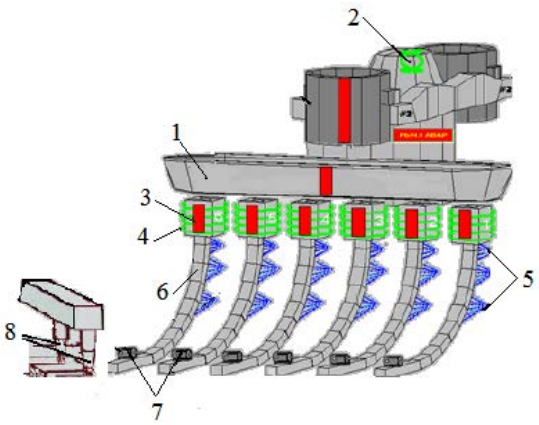


Figure 1 – Slab continuous casting machine

To determine key plant parameters that have greatest impact on quality of workpiece during continuous casting, it is necessary to analyze possible emergencies and their causes (Table 1).

Table 1 – Main emergencies in CC

Type Type of emergency / Location of emergency	Reason	Method of elimination
Slab breaker / Crystallizer	High speed pulling of workpiece	Setting speed according to pouring speed and acceleration chart
Crust thickness deformation / Crystallizer	Metal level	The required metal pouring rate, which depends on cross-section of crystallizer and flow-through section of slag dosing unit
Slab deformation / Crystallizer	Excessive wobbling of crystallizer	Distance between rollers according to thickness of workpiece or design features of rolling mechanism
Pouring steel over edges of crystallizer / Crystallizer	Improper adjustment of industrial bucket stopper position	Adjusting the position of stopper industrial bucket
Workpiece deformation / Pulling and straightening unit	High ingot drawing speed	Setting speed according to pouring speed and acceleration chart

After analyzing reasons, it was determined that following parameters should be selected for management:

1) metal pouring rate because this parameter affects frequency and amplitude of oscillations and shape of oscillations;

2) metal level in crystallizer because when controlling metal level in crystallizer, it is necessary to observe movement of industrial bucket stopper (it must be minimal) and prevent pulsation of metal jet entering crystallizer during casting process, in addition, to maintain continuous flow (jet) of metal during casting and stable metal level in crystallizer;

3) stopper position industrial bucket.

All these parameters affect required thickness of crust, which is shell of future billet.

Since key parameters of CC unit were defined above, they are written in form of parametric model as follows:

$$b=<V_p, G_{cc}, P_{ib}, T_{cc}, M_{lv}>.$$

Fig. 2 shows generalized representation of parameters in form of black box model.

The input data to "gray box" is set represented by expression above), which describes all parameters necessary and sufficient for complete selection of continuous casting parameters.

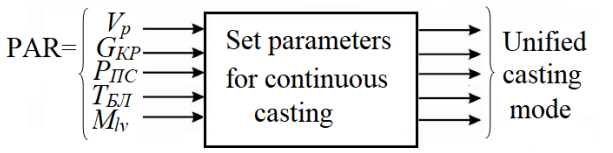


Figure 2 – Black box model

DOI 10.15588/1607-3274-2024-2-18

Inside “black box” there is set Π of all possible variants of casting parameters P, which has following properties:

1) possibility of existence $\Pi \in P$, provided that $P \to \max$;

2) set Π contains subset of $(T_1, T_2, ..., T_r) \in T$, as logically ordered set of parameters required for casting.

The output is partially unified sequence of casting mode parameters.

3.2 Technical means for controlling casting parameters

Effective control of casting parameters is crucial to ensure quality and safety of production process. In this context, use of technical means becomes necessity, and key element of such means is sensors that allow us to accurately measure and control various casting parameters, ensuring effective process control.

To begin with, metal level in crystallizer will be monitored using special sensor – XLEV eddy current level sensor, which detects level of molten metal. We chose XLEV because it has several key advantages that make it optimal choice for monitoring metal level in crystallizer:

1) it uses principle of eddy currents, which allows for non-contact measurement of level of molten metal without direct contact with it. This is important to avoid possible deformation or contamination of sensor, which can affect its accuracy;

2) XLEV has high sensitivity and measurement accuracy, which allows it to detect even slightest changes in metal levels. This ensures reliable control and response to any changes in production process [17].

Accurate speed measurements are crucial for control in continuous casting, so LSV-2100 optical speed measuring device was chosen because it also offers following advantages:

1) non-contact measurement principle with laser precision and no need for recalibration;

2) direct feedback via touchscreen display.

Digital frequency meter 10-199.9 Hz was chosen to measure oscillation frequency because it provides high measurement accuracy. Its digital nature allows for clear and stable results without distortion that can occur with analog measurements. In addition, this frequency meter has wide measuring range from 10 to 199.9 Hz, making it versatile and suitable for measurements in various crystallizer operating conditions. It is able to effectively detect even high-frequency vibrations that can occur as result of changes in metal crystallization process.

Vibrations will be measured with 640B01 sensor – industrial speed converter because it:

1) it is specifically designed for measuring vibrations in industrial environments. It has high sensitivity and measurement accuracy, which allows it to effectively detect various vibrations and vibrations that may occur during operation of crystallizer;

2) 640B01 sensor, equipped with communication interfaces that allow it to be effectively integrated into monitoring and control system [18]. This makes it convenient tool for continuous vibration monitoring and responding to any unforeseen situations.

LTR series linear encoder is linear encoder for measuring short displacements with return spring. Important feature of LTR series position sensors is presence of return spring.

3.3 Development of control system operation algorithm for continuous casting plant

Fig. 3. The developed algorithm of control system operation for continuous casting machine is shown.

First, power supply is switched on and molten metal from industrial bucket is poured into water-cooled crystallizer, where it is formed into slabs and cooled in copper sleeve.

Next, pouring speed is checked, if value is 2 (is value obtained in calculations that will be presented in further works), slab is gradually poured into the crystallizer. If the speed is less than or greater than 2, then this speed can lead to slab rupture, which means that pouring speed must be controlled via PID controller.

The next step is to check shape of oscillation, if it is sin-shaped, then oscillation frequency should be equal to 164, if this condition is met, then amplitude of oscillation should be checked.

If oscillation frequency is not 164 (is value obtained in calculations), then we need to determine whether it is equal to 426 (is value obtained in calculations).

Checking condition and, if «Yes», whether waveform is non-sinusoidal.

If oscillation frequency is generally either higher or lower than 426, then you need to control value of this parameter.

After that, amplitude of oscillations is checked, it should be 242 (is value obtained in calculations), if condition is met, then hydraulic cylinders are turned on.

The slab crystallizer of continuous casting machine is driven by two hydraulic cylinders located on both sides of movable frame.

If «No», then you need to adjust oscillation frequency.

Important task when pouring metal is to maintain given metal level in crystallizer. Therefore, it is necessary to determine level of metal in crystallizer. According to calculation results, metal level should be 0.85 (is value obtained in calculations that will be presented in further works).

If this condition is met, stopcock should turn off.

If this condition is not met, hydraulic cylinders must be switched on again.

It may happen that stopcock does not turn off, in which case you need to turn it off.

DOI 10.15588/1607-3274-2024-2-18

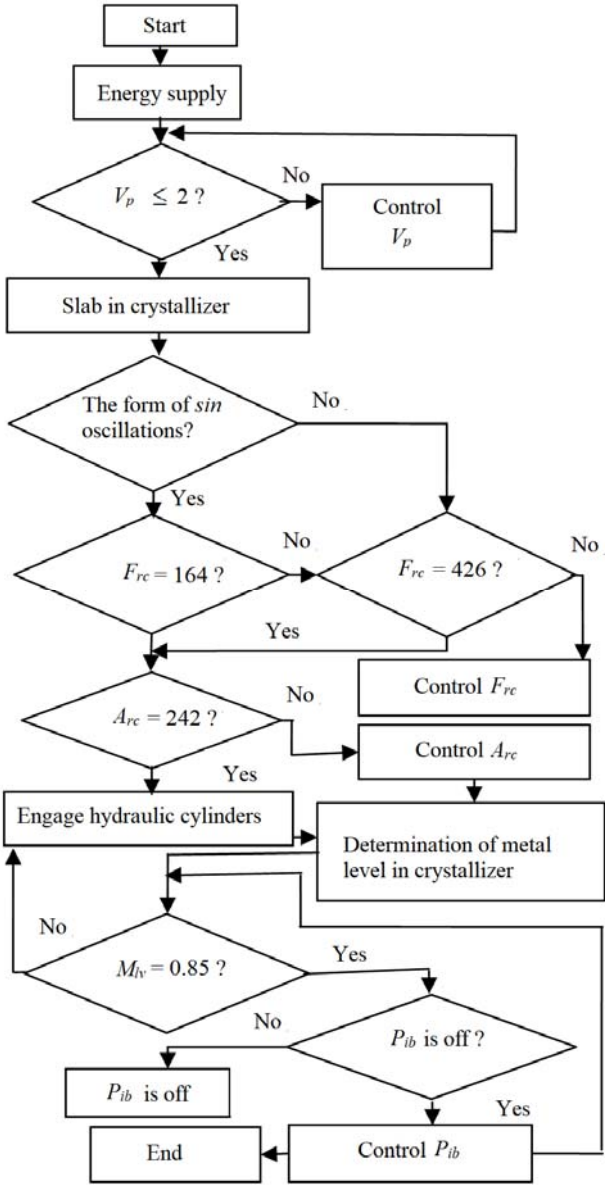


Figure 3 – Algorithm of the control system operation

Therefore, let developed system conventionally consist of two circuits: internal circuit for controlling position of stopper and circuit for controlling metal level in crystallizer.

4 EXPERIMENTS

The system is created with help of functional blocks in TraceMode system.

To begin with, you need to create project that will include operator station node and PID controller OVEN TRM 210 with RS-485 interface. Communication between operator station and device is carried out using OVEN protocol. Automatic converter from RS-485 to RS-232 interface is used as converter.

Using context menu, PLC_1 group is created in this group, and Owen RS485_group is created in this group (Figure 4).

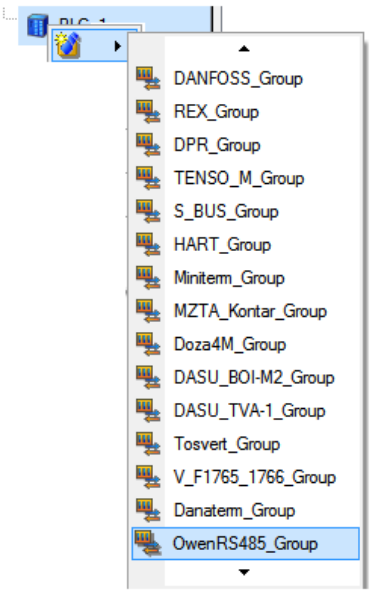


Figure 4 – The process of creating new group in Trace Mode

Fig. 5 shows window with created «Arguments» – necessary for communication with screen (interface) elements.

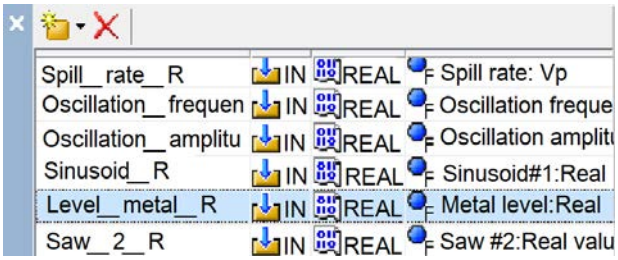


Figure 5 – The window with created «Arguments»

Another screen was also created in order to implement control of locking mechanism of CCM.

To connect elements of second screen, MODBUS_3 group and corresponding groups with names were created in «Sources/Receivers» layer:

– Read_state_Coil – reading states of discrete outputs;
– Read_State_Inp – reading states of discrete inputs;
– Write_Singl_Coil – control of discrete outputs.

In each group, components are created (Figure 6), which are connected to elements on screen through appropriate arguments:

– ARG_000 – reading status of discrete inputs;
– ARG_001 – reading status of discrete outputs;
– ARG_002 – means of controlling discrete output.

Fig. 6 shows fragment of project navigator tree with created components in «Sources/ Receivers» layer.

DOI 10.15588/1607-3274-2024-2-18

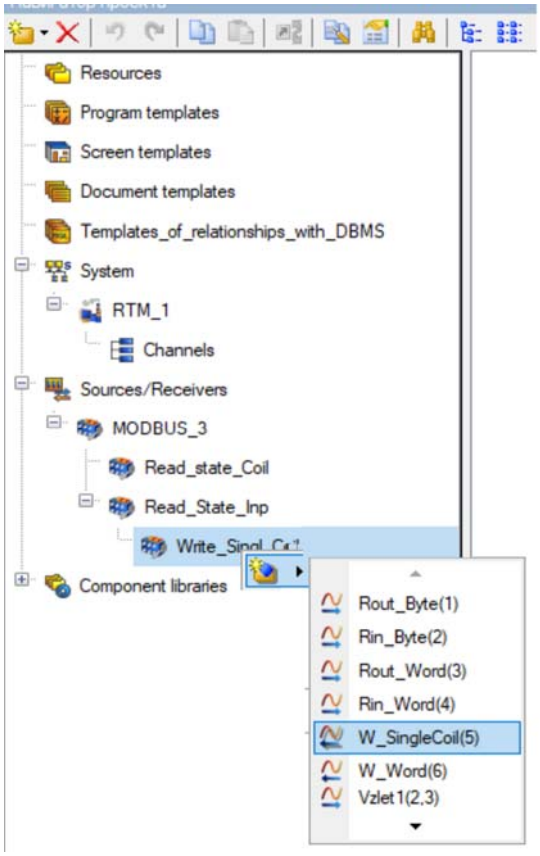


Figure 6 – Creating components in «Sources / Receivers» layer

The «Resources» layer can be used to centrally manage and monitor various resources required for effective operation of automated control system.

In «System» layer, RTM_1 project node is created, and then three groups and channels with same names are created in node: Read_state_Coil, Read_State_Inp, and Write_Singl_Coil.

Visualization of controlling process of key parameter values will be provided by graphical element «Slider».

Fig. 7 shows graphical interface (Display 1) for presenting data on frequency and amplitude of oscillation on operator's monitor.

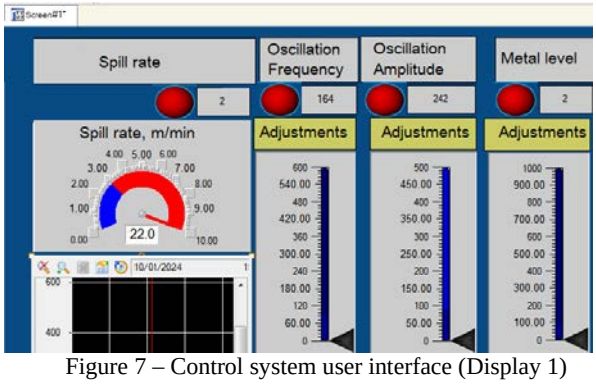


Figure 7 – Control system user interface (Display 1)

All elements: «Arrow device», «Trend», «Slider», «Text» were added to screen by drag-and-drop.

5 RESULTS

Fig. 8 shows developed user interface of control system.

To check whether binding of graphic elements to screen arguments is set correctly, you can use emulation mode. To switch to emulation mode, use special icon on toolbar. When you click on screen of graphic editor, window for setting value of argument in corresponding field is displayed (Figure 8).

Below is list of controls that are included on screen of developed system:

Process parameters: metal level in crystallizer, metal pouring rate, position of stopcock.

On screens 1 and 2, there are elements such as «Trend» graph and «Arrow device» for better perception of process dynamics, and history of changes in key parameters over time will be archived every 4 hours, while minimizing excessive amount of stored data.

The metal pour rate indicator shows current speed of molten metal. It is important to monitor this parameter because it can affect quality of casting and shape of product.

Oscillation frequency and amplitude indicators to monitor these parameters to avoid process instability and improve product quality.

If it is necessary to turn off locking valve, button changes from «ON» to «OFF», which allows operator to intervene quickly and easily if necessary.

As result, when formulating context of this work, its main goal was determined – development of automated control system to ensure accuracy and stability of casting process. As result of this work, this goal has been achieved because:

– analysis of current state of continuous casting production;

– analysis of molding machine;

– nature of cast billet was determined and required CCM was selected;

– analyzed main emergency situations in casting process;

– key parameters of continuous casting were selected with emphasis on casting machine;

– parametric model of CC is proposed;

– technical means for controlling casting parameters were selected and substantiated;

– developed algorithm for control system for continuous casting machine;

– operator interface was developed and virtual devices were connected;

– comparison with analog was made.

DOI 10.15588/1607-3274-2024-2-18

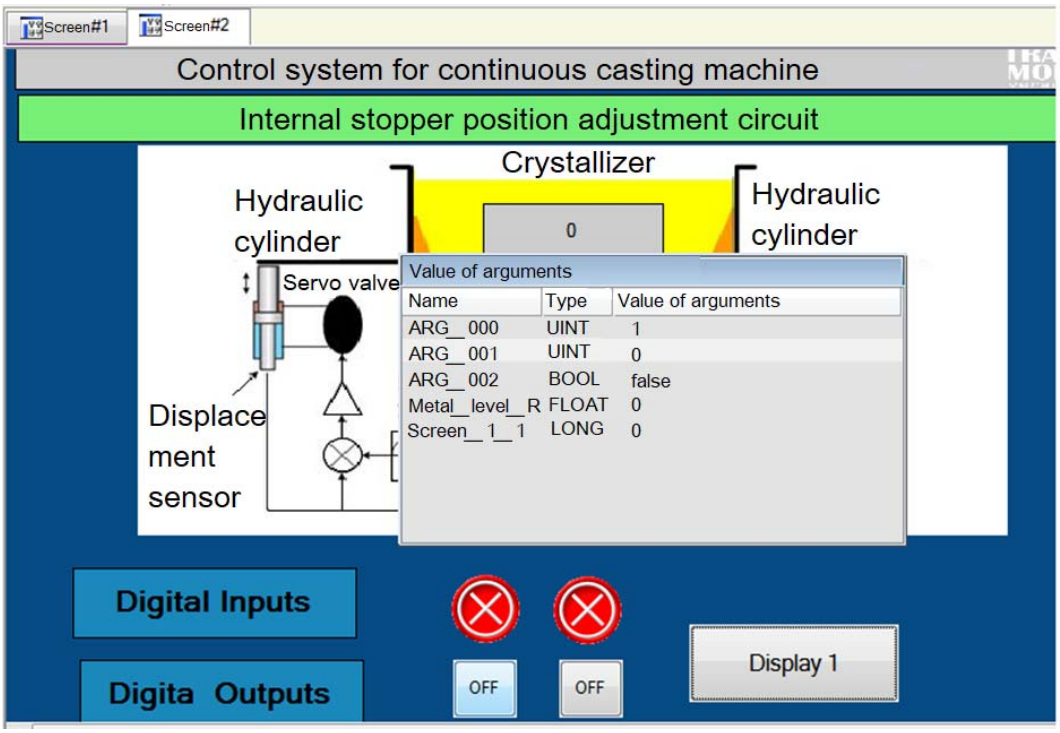


Figure 8 – Interface of developed system in process of emulation (Display 2)

6 DISCUSSION

The developed automated control system for continuous casting is represented by two screens with elements for monitoring, adjusting and controlling key parameters of mold.

A common visual representation provides clarity and intuitive navigation for operator, which is important when working with automated control systems.

To compare development with analogues, we selected program that is used to control continuous casting machine.

The software for controlling molds is classified, but there are modules for modeling metal casting.

ABAX TubeStar, automated crystallizer tracking system, is special function of standard automated control system for continuous casting machines [13].

The main functional aspects are: safe system operation, high stability of meniscus level, high operational reliability, simple operation, low investment and maintenance [13].

In ABAX TubeStar, user enters input data required for crystallizer monitoring step by step.

Project data is stored and can be displayed in reports.

The following information is monitored and recorded continuously:

– number of melts and weight of ingots cast in tons;

– steel grades of steel to be poured and ingots already poured;

– casting speed;

– heat removal;

– temperature difference in primary cooling;

– location of stream;

– current internal geometry of sleeves as measured by system.

Table 2 shows input data for steel casting process monitoring experiment.

Whole experiment is divided into two stages for comparison:

1) data entry;

2) monitoring process of forming slab product.

As comparison criterion, we will count number of module failures over certain period of time, namely two days.

As result, one system was chosen for comparison, because others are paid, and these programs are highly specialized and developed for specific size of CCM.

It was found that during two days of using these programs, developed system «worked without failures» longer than ABAX TubeStar.

Fig. 9 shows results of research in form of diagram for clarity.

Table 2 – Input data of experimental study

	Characteristics			
	Filling speed, m/ min	Oscillation frequency, min^{-1}	Oscillation amplitude, mm	Metal level, m
	1	2	3	4
Value	1	164	242	0.85
	2	164	242	0.85
	3	164	242	0.85
	4	165	243	0.86
	5	165	243	0.86
	6	165	243	0.86
	7	166	244	0.87
	8	164	244	0.87
	9	166	244	0.87
	10	166	244	0.87

DOI 10.15588/1607-3274-2024-2-18

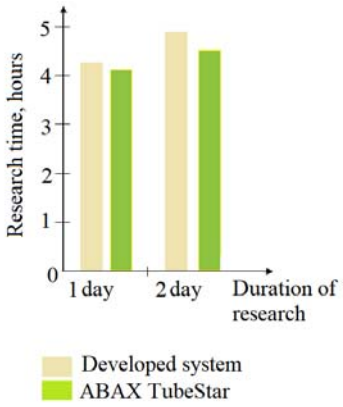


Figure 9 – Failure diagram of developed system and ABAX TubeStar module

During two-day experiment, developed system showed greater stability and efficiency, with fewer failures compared to ABAX TubeStar. The developed system allows user to enter data step by step to monitor crystallizer, ensuring high stability of meniscus level, safe operation and easy operation. Low investment and maintenance make it cost-effective option for controlling continuous metal casting plant. The experimental results show advantages of development compared to ABAX TubeStar in controlled monitoring of steel casting.

CONCLUSIONS

The urgent problem of developing automated control system for continuous casting is solved.

The scientific novelty of results obtained is that for first time parametric model for optimizing casting process has been proposed, which can be used both in design of new plants and to optimize operation of existing ones. This allows to reduce production time and costs and ensure higher quality of manufactured blanks. This model differs from existing ones in that it takes into account position of industrial bucket stopper, which leads to increase in quality of slabs produced.

Based on parametric model, Trace Mode system was used to implement automated control system for continuous casting, which is user-friendly because it consists of two screens for monitoring, regulation and control. The system is easy to use. Advantage is built-in function of auto-automated control of slab CCM, which contributes to safety and stability of casting process.

The practical value of results obtained is that software has been developed that implements control of plant during continuous casting, and experiments have been conducted to study stability of developed system. The results of experiment allow us to recommend proposed system for use in practice.

Thus, results obtained reflect important step in development of methods for controlling and monitoring continuous casting, ensuring improved product quality and optimization of production processes.

Prospects for further research are to expand functionality of system and adapt it to various production conditions. Namely, to add to proposed screens (Display 1 and 2) screen with two basic parameters of casting process temperature and pressure to expand class of practical tasks.

REFERENCES

1. Sotnik S. V., Mikitenko V. A. Obzor sovremennyih sistem upravleniya dlya nepreryivnogo littya, *M&MS 2019, 24–25 October*. Kharkiv, UKRAINE, 2019, R. 45–47.
2. Sotnik S., Tereshchuk D. O., Trokhin V. V. Development of remote control for thermoplastics dosing automation system, *The 5th International scientific and practical conference "Topical aspects of modern scientific research" (January 25–27, 2024) CPN Publishing Group*. Tokyo, Japan, 2024, pp. 179–184.
3. Sotnik S. V., Redkin K. S. Design features of control panels and consoles in automation systems, *The 9th International scientific and practical conference "Science and innovation of modern world" (May 18–20, 2023) Cognum Publishing House*. London, United Kingdom, 2023, pp. 201–205.
4. Sotnik S., Matarneh R., Lyashenko V. System model tooling for injection molding, *International Journal of Mechanical Engineering and Technology (IJMET)*, 2017, Vol. 8, № 9, pp. 378–390.
5. Al-Sherrawi M. H., Saadoon A. M., Sotnik S., Lyashenko V. Information model of plastic products formation process duration by injection molding method, *International Journal of Mechanical Engineering and Technology (IJMET)*, 2018, Vol. 9, № 3, pp. 357–366.
6. Arnu D., Yaqub E., Mocci C. et al. A reference architecture for quality improvement in steel production, Data Science – Analytics and Applications. Wiesbaden, Springer Fachmedien Wiesbaden, 2017, pp. 85–90.
7. Cemernek D. Cemernek C., Gursch H. et al. Machine learning in continuous casting of steel: A state-of-the-art survey, *Journal of Intelligent Manufacturing*, 2021, pp. 1–19.
8. LIU Q., LIU Q., YANG J. et al. Progress of research on steelmaking-continuous casting production scheduling, *Chinese Journal of Engineering*, 2020, Vol. 42, №2, pp. 144–153. DOI: 10.13374/j.issn2095-9389. 2019. 04. 30.002
9. Zhang F., Liu M. et al. An IoT-based online monitoring system for continuous steel casting, *IEEE Internet of Things Journal*, 2016, Vol. 3, № 6, pp. 1355–1363. DOI: 10.1109/JIOT.2016.2600630
10. Zhang Y., Dudzic M. S. Online monitoring of steel casting processes using multivariate statistical technologies: From continuous to transitional operations, *Journal of Process Control*, 2006, Vol. 16, №8, pp. 819–829. DOI: 10.1016/j.jprocont.2006.03.005
11. Pyszko R., Franìk Z. et al. Monitoring and simulation of the unsteady states in continuous casting, *Materiali in tehnologije / Materials and technology*, 2018, №2, pp. 111–117.
12. Miłkowska-Piszczek K., Falkus J. Control and design of the steel continuous casting process based on advanced numerical models, *Metals*, 2018, Vol. 8, №8, pp. 1–16. DOI:10.3390/met8080591
13. ABAX TubeStar [Electronic resource]. Access mode: http:// abax-steel.com/automation-systems/
14. Guthrie R. I. L., Isac M. M. Continuous casting practices for steel: Past, present and future, *Metals*, 2022, Vol. 12, №5, P. 862. DOI: 10.3390/met12050862
15. Shakhov S. I. et al. Improvement of built-in electromagnetic stirring in the molds of bloom continuous-casting machines, *Metallurgist*, 2020, Vol. 64, pp. 410–417. DOI: 10.1007/s11015-020-01010-y
16. Nick A. S., Vynnycky M. On longitudinal electromagnetic stirring in the continuous casting of steel blooms, *Journal of Engineering Mathematics*, 2020, Vol. 120, № 1, pp. 129–151. DOI: 10.1007/s10665-019-10035-5
17. XLEV-S, XLEV-L, AVEMIS Eddy Current Mould Level Sensor. [Electronic resource]. Access mode: http:// www.vesuvius.com/en/our-solutions/international/iron-and-steel/continuous-casting/flow-control-product-pages/Xlev.html
18. Platinum 4–20 ma velocity transmitter [Electronic resource]. Access mode: https:// www.pcb.com/products?m=640B01

Received 09.02.2024.
Accepted 28.04.2024.

DOI 10.15588/1607-3274-2024-2-18

УДК 681.5: 673.1

РОЗРОБКА АВТОМАТИЗОВАНОЇ СИСТЕМИ УПРАВЛІННЯ ПРИ БЕЗПЕРЕРВНОМУ ЛИТТІ

Сотник С. В. – канд. техн. наук, доцент, доцент кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки Харківського національного університету радіоелектроніки, Харків, Україна.

АНОТАЦІЯ

Актуальність. На сьогодні автоматизовані системи управління безперервним литтям стрімким чином розвиваються, оскільки процес виготовлення заготовки (виробів) одного розміру з металу у виливниці при масовому виробництві давно застарів і настає «етап безперервного лиття». Такий процес виготовлення виробів підходить для кольорових металів та сталі. Однак, при розробці кожен раз виникає задача підвищення якості отриманої заготовки, яка безпосередньо залежить від оптимізації ефективності та надійності самих автоматизованих систем.

Оптимізація є ключовим етапом в процесі розробки, оскільки вона спрямована на забезпечення точності та стабільності процесу лиття, що включає в себе розробку параметричної моделі та точних алгоритмів, які забезпечують оптимальну температуру, швидкість розливу металу, частота коливань, амплітуда коливань, рівень металу в кристалізаторі та положення стопора промковша для кожного етапу лиття. Зокрема, у відомій авторам літературі ця задача досі не вирішена у повному обсязі, тому, необхідно провести постановку задачі та розробити алгоритм роботи системи для конкретної установки безперервного лиття.

Мета. Метою дослідження є розробка автоматизованої системи управління для забезпечення точності та стабільності процесу лиття.

Метод. Розроблена система управління установкою безперервного лиття спирається на запропоновану параметричну модель, яка формалізована на базі теорії множин. Модель враховує ключові параметри для конкретного процесу лиття: швидкість розливу металу, частота коливань, амплітуда коливань, рівень металу в кристалізаторі та положення стопора промковша.

Результати. Здійснена постановка задачі та визначені ключові параметри, які враховані в алгоритмі роботи системи, а це дало можливість розробити систему управління установкою безперервного лиття для вирішення задачі підвищення якості отриманої заготовки.

Висновки. Створено параметричну модель та узагальнене представлення у вигляді моделі чорного ящика, які є необхідними як для нових проектів безперервного лиття, так і для вже існуючих агрегатів з метою оптимізації процесу лиття металу. Для налаштування системи безперервного лиття були визначені контрольовані параметри, такі як швидкість розливу, частота та амплітуда коливань, рівень металу у кристалізаторі та положення стопора промковша. Розроблено алгоритм системи управління для установки безперервного розливу на основі якого розроблено система, яка дозволяє реалізувати моніторинг, регулювання та управляння процесом отримання заготовок зі сталі або кольорових металів. Розроблено інтерфейс користувача системи управління є простим та зручним для користування.

КЛЮЧОВІ СЛОВА: процес, безперервне лиття, автоматизація, система, управляння.

ЛІТЕРАТУРА

1. Sotnik S. V. Obzor sovremennyih sistem upravleniya dlya neprery-ivnogo littya / S. V. Sotnik, V. A. Mikitenko // M&MS 2019, 24–25 October, Kharkiv, UKRAINE. – 2019. – R. 45–47.
2. Sotnik S. Development of remote control for thermoplastics dosing automation system / S. Sotnik, D. O. Tereshchuk, V. V. Trokhin // The 5th International scientific and practical conference "Topical aspects of modern scientific research" (January 25–27, 2024) CPN Publishing Group, Tokyo, Japan. – 2024. – P. 179–184.
3. Sotnik S. V. Design features of control panels and consoles in automation systems / S. V. Sotnik, K. S. Redkin // The 9th International scientific and practical conference "Science and innovation of modern world" (May 18–20, 2023) Cognum Publishing House, London, United Kingdom. – 2023. – P. 201–205.
4. Sotnik S. System model tooling for injection molding / S. Sotnik, R. Matarneh, V. Lyashenko // International Journal of Mechanical Engineering and Technology (IJMET). – 2017. – Vol. 8, № 9. – P. 378–390.
5. Information model of plastic products formation process duration by injection molding method / [M. H. Al-Sherrawi, A. M. Saadoon, S. Sotnik, V. Lyashenko] // International Journal of Mechanical Engineering and Technology (IJMET). – 2018. – Vol. 9, № 3. – P. 357–366.
6. A reference architecture for quality improvement in steel production / [D. Arnu, E. Yaqub, C. Mocci et al.] // Data Science – Analytics and Applications. – Wiesbaden: Springer Fachmedien Wiesbaden. – 2017. – P. 85–90.
7. Machine learning in continuous casting of steel: A state-of-the-art survey / [D. Cemernek, C. Cemernek, H. Gursch et al.] // Journal of Intelligent Manufacturing. – 2021. – P. 1–19.
8. Progress of research on steelmaking–continuous casting production scheduling / [Q. LIU, Q. LIU, J. YANG et al.] // Chinese Journal of Engineering. – 2020. – Vol. 42, №2. – P. 144–153. DOI: 10.13374/j.issn2095-9389. 2019. 04. 30.002
9. An IoT-based online monitoring system for continuous steel casting / [F. Zhang, M. Liu et al.] // IEEE Internet of Things Journal. – 2016. – Vol. 3, №. 6. – P. 1355–1363. DOI: 10.1109/JIOT.2016.2600630
10. Zhang Y. Online monitoring of steel casting processes using multivariate statistical technologies: From continuous to transitional operations / Y. Zhang, M. S. Dudzic // Journal of Process Control. – 2006. – Vol. 16, №8. – P. 819–829. DOI: 10.1016/j.jprocont.2006.03.005
11. Monitoring and simulation of the unsteady states in continuous casting / [R. Pyszko, Z. Franìk et al.] // Materiali in tehnologije / Materials and technology. – 2018. – №2. – P. 111–117.
12. Miłkowska-Piszczek K. Control and design of the steel continuous casting process based on advanced numerical models / K. Miłkowska-Piszczek, J. Falkus // Metals. – 2018. – Vol. 8, №8. – P. 1–16. DOI:10.3390/met8080591
13. ABAX TubeStar [Electronic resource]. – Access mode: http:// abax-steel.com/automation-systems/
14. Guthrie R. I. L., Continuous casting practices for steel: Past, present and future / R. I. L. Guthrie, M. M. Isac // Metals. – 2022. – Vol. 12, № 5. – P. 862. DOI: 10.3390/met12050862
15. Improvement of built-in electromagnetic stirring in the molds of bloom continuous-casting machines / [S. I. Shakhov et al.] // Metallurgist. – 2020. – Vol. 64. – P. 410–417. DOI: 10.1007/s11015-020-01010-y
16. Nick A. S. On longitudinal electromagnetic stirring in the continuous casting of steel blooms / A. S. Nick, M. Vynnycky // Journal of Engineering Mathematics. – 2020. – Vol. 120, № 1. – P. 129–151. DOI: 10.1007/s10665-019-10035-5
17. XLEV-S, XLEV-L, AVEMIS Eddy Current Mould Level Sensor. [Electronic resource]. – Access mode: http:// www.vesuvius.com/en/our-solutions/international/iron-and-steel/continuous-casting/flow-control-product-pages/Xlev.html
18. Platinum 4–20 ma velocity transmitter [Electronic resource]. – Access mode: https:// www.pcb.com/products?m=640B01

DOI 10.15588/1607-3274-2024-2-18

Наукове видання

Радіоелектроніка,

інформатика,

управління

№ 2/2024

Науковий журнал

Головний редактор – д-р техн. наук С. О. Субботін

Заст. головного редактора – д-р техн. наук Д. М. Піза

Комп’ютерне моделювання та верстання — С. В. Зуб
Редактор англійських текстів — С. О. Субботін

Оригінал-макет підготовлено у редакційно-видавничому відділі НУ «Запорізька політехніка»

Свідоцтво про державну реєстрацію
КВ № 24220-14060 ПР від 19.11.2019.

Підписано до друку 17.05.2024. Формат 60×84/8.
Папір офс. Різогр. друк. Ум. друк. арк. 22,09.
Тираж 300 прим. Зам. № 511.

69063, м. Запоріжжя, НУ «Запорізька політехніка», друкарня, вул. Жуковського, 64

Свідоцтво суб’єкта видавничої справи
ДК № 6952 від 22.10.2019.