Computer Science Journal of Moldova, vol.**31**, no.3(93), 2023

Supplementing elearning systems with adaptive content generation elements

Alexandr Parahonco, Mircea Petic

Abstract

The paper describes automatic summarization as one of the topic that helps elearning system to be more adaptable on content generation. This article treat automatic summarization with approaches that provide the ability to summarize texts for different languages. In the case of this article, it is about the English, Romanian and Russian languages. The paper contains both the description of the problem and different approaches already used by other researchers. Next, the data with which the automatic summarization experiments were carried out were described. The metrics with which we can evaluate the quality of the summarization result were presented. Finally, some thoughts were formulated regarding the results obtained in the experiment.

Keywords: elearning systems, text summarization, evaluation metrics, datasets, data analysis.

MSC 2020: 68T50, 68T05.

1 Introduction

Humanity in the 21st century has made a huge leap in computer science. Chat GPT is the best demonstration of this thesis. It would seem that lots of information on the Internet hampers human work with its variation and validity. Chat GPT has succeeded with this problem and proposed remarkable results. This opportunity throws light on content generation [1].

On the other hand, because of the enormous amount of information daily generated the manual completion of elearning system with

doi:10.56415/csjm.v31.18

appropriate content it is a difficult task for every teacher. With these special possibilities of automatic content generation, it becomes an interesting idea to supplement elearning platforms with a part of these huge amount of information referring to specific topics. In the conditions when there is so much information on the Internet, the need to reduce and understand it, in a limited time, becomes important. The processes of text understanding and production are directly related to the creation of summaries. That is why making a consistent summary is an important approach to understand and to select the appropriate idea to be included in educational materials on elearning platforms.

Automatic text summarization is a technique that takes a source text and extracts the most crucial information, condensing it and tailoring it to the demands of the user or job. The source text is first read, and its content is identified. The main points are then collected in a brief summary [2, pp. 2-4]

Searching approaches on automatic summarization, literature review brought us three solutions: *prompt engineering, abstractive-based summarization* and *extraction-based summarization*. The first two consider neural network technology, namely **transformers**. The last one relies on **standard NLP techniques** [3]. In this article, we consider the last solution: extraction-based summarization.

In contrast to abstractive techniques, which conceptualize and paraphrase a summary, extractive techniques accomplish summarization by selecting bits of texts and creating a summary [4].

The **purpose** of this article is to find the way of evaluating text summaries in the first place, to identify the best approach for summarization in the second place, and to investigate whether there are problems from a multilingual perspective in this procedure in the third place.

To achieve the goal of the paper, we will structure the paper as follows. Initially we will present the data we will work with, namely their type, quantity and scope. We will continue with the presentation of the methods and metrics needed to evaluate the experiment and after that the essence of the experiment and the data obtained will be presented. Finally, we will draw some conclusions based on what we obtained.

2 Overview of project articles

Content generation task can be viewed from different angles. Starting from the idea of adaptive content generation for eLearning platforms, the following can be regarded:

1. answers for student questions;
2. e-course content for teachers;
3. items from the test.

The last one refers to **adaptive assessment**. From this perspective, *the responsibility of item ordering is assigned to the software part.* There are two main categories of strategies for presenting test items: **two-step** and **multi-step** (Fig. 1) [5].

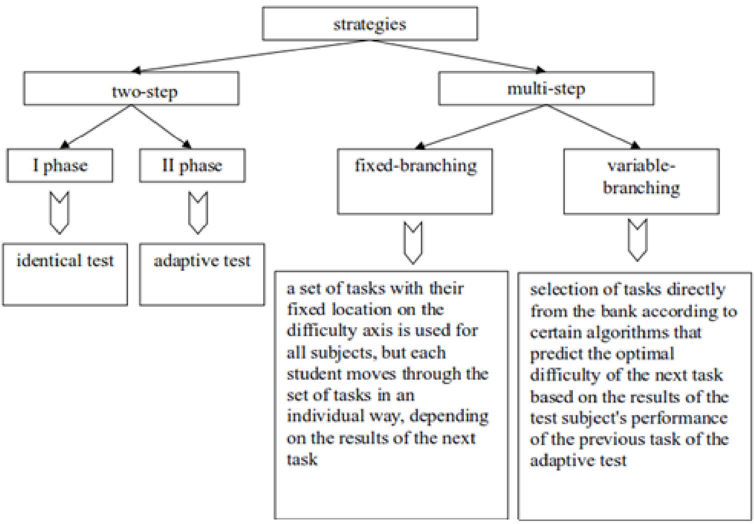

Figure 1. Adaptive testing strategies

As part of our project, it was decided to use the **Moodle** platform to implement our ideas and developments. Thus, we have developed two plugins: **TestWid** for adaptive assessment and **TestWidTheory** for content generation. The TestWid plugin is based on a **multi-step**

fixed-branching strategy. It uses a **bank of items** and **categories of items**. Let us group the items in these categories according to their complexity. Hereby, each time the student takes a test, new items are randomly selected (15 items in total). The plugin also allows you to **make retakes** for the same test with only one requirement: *there should be at least two items in each category*. So the student could obtain new items to deal with. Otherwise, the test could not be launched [5].

Another look at content generation regards **e-course development**. In contrast with adaptive assessment, where the notion of "generation" is treated as the order change of items, e-course development refers to the *fetching of information from the Internet in an advanced mode*. Our research papers [6], [7] suggest a focused web crawler based on a web-scraping approach for information extraction from the Internet and its further processing. Fig. 2 can provide a detailed view of our application for e-course development.

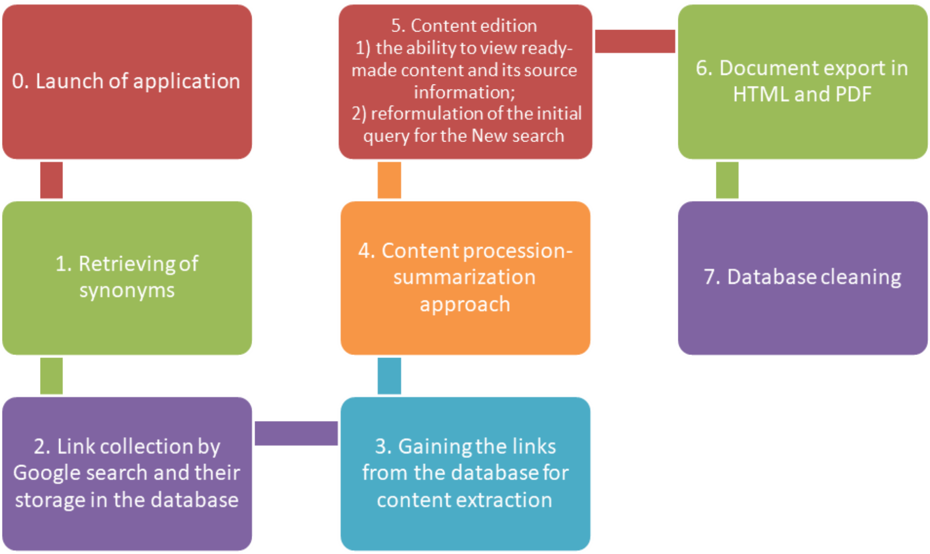

Figure 2. Scheme of the program model for the dynamic creation of e-courses

According to our approach, we have six steps. **In the first step**, some web crawlers create networks of synonyms. **In the second step**, our application uses the original request and/or their selected synonyms for advanced search using Google search. **Next**, in step three, we gain links for the requests from Google and process them (crawl, select necessary fragments), storing all the information in the database. **Step 4** is responsible for the text processing and the extraction of the most valuable information. A literature review in this domain sheds light on the summary approaches. Summarization will help us interpret large amounts of the sources and present this resume as one final product, i.e., we can summarize one or many (depending on the summary approach) text sources and provide it to the user. According to **step 5**, the summarized content should be edited and further exported in HTML or PDF formats in **step 6**.

From the student point of view, content generation can be considered a question-answering solution. Insofar as it represents the essential way of understanding the learning material throughout the querying, it is a good practice to provide such assistance.

In this case, the same application for e-course generation can be the **prototype with some little changes**. Especially now, with the appearance of ChatGPT and other similar chat-bots, it becomes profitable to have one specific AI solution on the same platform that could effectively manage learning resources on the platform and outside it, on the one hand, and provide some extra possibilities, on the other.

Finally, this application is going to be integrated into the Moodle eLearning platform as a part of the **TestWidTheory** plugin and some new one for student assistance (chat-bot). Thus, this solution will be part of the Moodle standard tool set, which will always be at the teacher's hands.

To begin with the experiment, it should be mentioned that the Internet contains an enormous amount of information that requires careful processing, selecting only valuable passages. The recent study led us to the **summary approaches** to process the information taken from the internet and present it to the teacher. It is going to be implemented **in step 4** and will be discussed further.

3 Experiment description

In order to examine the quality of the extractive summarization, six texts from different sources with different structures and domains of topic were selected (see Table 1).

Table 1. Descriptions of the selected datasets

	Domain	**Language**	**Chars**	**Words**
1	History	English	8599	1309
2	Geography	English	7939	1326
3	Biology	Russian	10323	1356
4	Literature	Russian	53328	6879
5	Informatics	Romanian	20579	3063
6	Law	Romanian	11780	1652

As part of our experiment, we investigated various types of methods for automatic summarization. Some of them are built on plain speculations and others are built on more complicated algorithms. The following summary methods described in paper [8] were investigated:

1. **Luhn's Heuristic Method** - propose that the **significance of each word** in a document **signifies how important it is**. According to this theory, *sentences that contain more of the stop-words* (words with the highest frequency) than others *do not have a greater impact* on the document's meaning [13].

2. **Edmundson Heuristic Method** - recommends **the use of a subjectively weighted mixture of features**. He took into account the features that were previously well-known and utilised in Luhn's method, but he also included a few new features such as *cue words* and *document structure* [14].

3. **Latent semantic analysis (LSA)** - is a reliable algebraic-statistical technique that can **find synonyms in the text and subjects that aren't mentioned clearly in the text**. LSA works by *breaking down the data into small, manageable spaces* [2, p. 1002].

4. **SumBasic algorithm** - produces summaries of **length n**, where **n** is the user-specified number of sentences.

5. **Kullback-Lieber (KL) Sum algorithm** - its goal is to identify a set of sentences whose length is fewer than **L words** and whose unigram distribution closely resembles that of the source text [11, pp. 522-523].

6. **Graph-based summarization (Reduction)** - employs a graph to rank the necessary sentences or words in our **unsupervised strategy**. The primary goal of the graphical method is to extract the most significant sentences from a single source.

7. **LexRank algorithm** - is also a method related to graph based approach. It uses the cosine similarity of TF-IDF vectors;

8. **TextRank algorithm** - is also a method related to graph based approach. It uses measure based on the number of words two sentences have in common (normalized by the sentences' lengths).

9. **Term Frequency method** - enlightens us as to *which terms are most frequently used* and sheds light on the *significance of particular terms* in a given text or group of papers. The length of each document varies, thus *it is likely that a term will appear more frequently in larger documents* than in shorter ones. In order to normalize term frequency, it is frequently divided by the total number of terms in the document. Other methods of normalizing word frequencies include using the average and maximum term frequencies found in a document.

10. **Term Frequency-Inverse Document Frequency (TF-IDF)** - is a commonly used method in NLP to assess the importance of words in a document or corpus. IDF is a weight that **represents a word's usage volume**. The lower the score, the more frequently it is used throughout documents. A text vectorization procedure converts words in a text document into significance numbers. The TF-IDF vectorization/scoring method, as the name suggests, *multiplies the Term Frequency* (TF) and

Inverse Document Frequency (IDF) of a word to determine its score [15].

The first eight approaches were applied from the ***Sumy*** library for text summarization. The term frequency method was examined from ***NLTK*** and ***Spacy*** library and TF-IDF approach was examined by ***NLTK*** and ***Scikit-Learn***. Summing up, we have investigated twelve methods for text summaries.

In order to estimate the quality of each method, we used four metrics discussed in [2]:

- ROUGE (ROUGE-1, ROUGE-2, ROUGE-L) - score component provides a unique viewpoint on the effectiveness of the system-generated summary by taking various linguistic and grammatical elements into account [9, p. 74]. It defines *how much of the words in reference summaries* appeared in the candidate summaries.

- BLEU - is based on the basic idea of comparison machine translations/summarization with those regarded to be accurate by humans. Each segment (mainly sentences) is being compared with a set of qualitative reference texts. The obtained scores are then averaged over the whole corpus to reach an estimate of the translation's/summarization's overall quality [10, p. 394];

- METEOR - overpasses previous metrics, taking into account **grammar and semantics**. The metric is based on the harmonic mean of unigram precision and recall, with recall weighted higher than precision [10, p. 394].;

- F-score - also relies on precision and recall, but data are different. Precision represents the **number of sentences** taking place in both summaries divided by the number of sentences in the candidate summary The basic way how to compute the F-score is to count a harmonic average of precision and recall.

Beyond the upper metrics, we have used the metrics provided by the ***Sumy*** library, which evaluates its own algorithms with ***ROUGE***, ***F-score*** and ***Unit overlap*** metrics.

The experiment consisted of a process that had two loops: an outside loop for changing texts and an inside loop to change summary methods (Fig. 3).

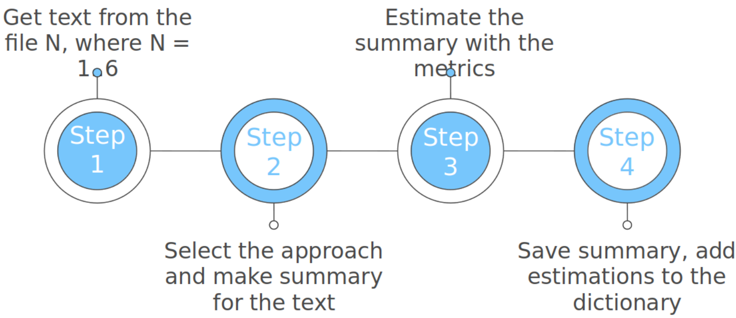

Figure 3. The course of the experiment.

At the end of the inner loop, all metrics were taken and analyzed. The experiment includes 72 iterations.

4 Data analysis

Working with the results, we have applied the **Max aggregated function** to get the highest results for each parameter. Our aim is understand which method is the best to be used and in what circumstances. That is why we analysed the methods from the following perspectives: overall effective summary approach; multilingual summary problems and approach versatility; comparison of approach realization and metrics confluence.

After applying the designed algorithm with methods to our datasets, the following results were obtained (see Table 2).

As it can be seen, from all the approaches, **the most effective are** *Luhn's heuristic method, TextRank and Term frequency method.* Having in mind **language sensitiveness**, we can use *Luhn's heuristic method for the English* language and *Term frequency for the Russian and Romanian* languages.

Table 2. Evaluation of the summary approaches. Part 1

Rank	Language	ROUGE-1	ROUGE-2	ROUGE-L	BLEU	METEOR
1	**En**	Luhn 0,8042	Luhn 0,9827	Luhn 0,8042	Luhn 0,5322	TF-IDF (NLTK) 0,59
2	En	Tex-tRank 0,784	Tex-tRank 0,74	Tex-tRank 0,784	Tex-tRank 0,499	LexRank 0,455
1	**Ru**	TF (Spacy) 0,86	TF (Spacy) 0,804	TF (Spacy) 0,86	TF (Spacy) 0,632	TF-IDF (Scikit-Learn) 0,524
2	Ru	Tex-tRank 0,836	Tex-tRank 0,793	Tex-tRank 0,836	Tex-tRank 0,584	Luhn 0,4761
1	**Ro**	TF (Spacy) 0,888	TF (Spacy) 0,822	TF (Spacy) 0,888	TF (Spacy) 0,686	TF-IDF (Scikit-Learn) 0,497
2	Ro	Luhn 0,8363	Luhn 0,7933	Luhn 0,8363	Luhn 0,5824	TF (Spacy) 0,427

It should pay attention to the METEOR results. ROUGE and BLEU results coincide, but METEOR's data differs. From all inputs, TF-IDF approach was frequently selected.

Another group of metrics is given below (see Table 3). Here F-score is based on the ROUGE and BLEU results and Unit overlapping. As we have three types of ROUGE metric in F-score formula we will get three types of F-score. Thus F-1 is for ROUGE-1, F-2 is for ROUGE-2 and, F-L is for ROUGE-L.

The last column of Table 3 (Unit overlapping) is calculated on the basis of **Sumy** library that estimates *only its methods*. Thus not all summary approaches were taken into consideration.

Table 3. Evaluation of the summary approaches. Part 2

Rank	**Language**	**F-1**	**F-2**	**F-L**	**Unit overlapping**
1	**En**	Luhn 0,641	TF-IDF (NLTK) 0,626	Luhn 0,713	LSA/KL 0,33
2	En	TextRank 0,61	TF-IDF (Scikit-Learn) 0,564	TextRank 0,686	TextRank 0,30
1	**Ru**	TF (Spacy) 0,729	Reduction 0,591	TF (Spacy) 0,789	KL 0,38440
2	Ru	TextRank 0,688	Luhn 0,59	TextRank 0,755	Luhn 0,36705
1	**Ro**	TF (Spacy) 0,774	TF-IDF (Scikit-Learn) 0,59	TF (Spacy) 0,827	LexRank 0,36
2	Ro	Luhn 0,687	TF (Spacy) 0,562	Luhn 0,754	Luhn 0,33

Looking at the results, BLEU and METEOR provide an average value of around 50% of quality. This is comparable to 50% of the summarized volume of text. In contrast, ROUGE metric provided results about 80% of quality. This is normal because these metrics complement each other. You will have high BLEU if many terms from the candidate summary appear in the reference summary, and high ROUGE if many words from the candidate summary appear in the reference summary. The F-score, in this case, provides the common result as a summarization.

As a result, **the most effective approaches** from the second group of metrics are *Term frequency method, Luhn's heuristic method and Term Frequency-Inverse Document Frequency.* For English sources should be taken *Luhn's heuristic method and TextRank*, for Russian and Romanian languages suit *Term frequency, TextRank and Luhn's heuristic method.*

Unit overlapping metric emphasizes KL and Luhn's heuristic method in most cases. Unfortunately, this approach of evaluation was implemented in Sumy library for proper algorithms. And we cannot estimate other approaches.

Unfortunately, nothing can be said about the **readability** and **coherence** of the summaries. *The applied metrics cannot estimate these parameters.* Hypothetically, **as language is flexible** and has **different ways of expanding the context of speech**, there are problems **with its preservation during sentence extraction** during the extractive approach. For example, pronouns, link words, etc. Although we have noticed that each summary approach **cuts the original text at different places**, it is impossible to judge its efficacy without an **expert review**.

The number of words, it seems, **does not play an important role** in summary ranking. We had six texts of different lengths, but their summaries *were appreciated with the same high scores as others.* This is because we have indicated the summary length **in percentage**. That is why the ratio of the original text to its summary was always the same for metrics. **They pay attention only to the amount of corresponding words in both places**. That is not reasonable for extraction-based summarization. Though different scores show that

some differences exist.

Another thing that should be considered is **summary size**. The shorter the summary, the lower the quality. The results argue for good quality in both cases, with a small difference. More shorter summaries were not taken into account as we pursued the goal of designing **ecourse content in the educational area**. The courses should not be too small but contain relevant information for students.

The most productive summary approaches are Term frequency method, Luhn's heuristic method and TextRank. However, the language influence on method list seems to be vague. The **week point** in all these methods is **tokenization** and **stopwords** list. This topic necessitate more research and experiments for strong conclusions.

5 Conclusion

This paper is the extended and revised version of the conference paper [16] presented at WIIS 2023.

In this paper, we tried to find effective methods for text summaries from a multilingual perspective. All methods prefer the English language as the default. **Tokenization** and **stopwords** lists seem to **affect the Russian and Romanian languages**. Thereby, such "simple" approaches as TF or TF-IDF have high ranks compared to more advanced approaches.

It should be emphasized that this direction of research should be pursued. Now, the most effective methods are the term frequency method, Luhn's heuristic method, and TextRank.

The research has shown that **we cannot be firmly confident** in summary efficacy relying currently on evaluation metrics. We need some **expert opinion** to investigate such parameters as **readability** and **coherence** of the shortened texts. Only thereafter we can see what **pitfalls** also should be considered and conclude whether extraction-based summarization is good for **e-course content generation** and select the best approach. Also, we should regard some other solutions as **prompt engineering** and **abstractive-based summarization**.

Acknowledgments. This article was written within the framework of the research project 20.80009.5007.22 ”Intelligent information systems for solving ill-structured problems, processing knowledge and big data”.

References

[1] T. Mali and R. Deshmukh, “Use of chat gpt in library services,” *International Journal of Creative Research Thoughts*, vol. 11, no. 4, pp. f264–f266, Paper ID - 234893, 2023, [Online]. Available: http://www.ijcrt.org/papers/IJCRT2304646.pdf. DOI: 10.1729/Journal.33816.

[2] Josef Steinberger and Karel Jezek, “Evaluation Measures for Text Summarization,” *Computing and Informatics*, vol. 28, pp. 1001-–1026, 2009. ISSN: 1335-9150.

[3] Shashank Bhargav, Abhinav Choudhury, Shruti Kaushik, Ravindra Shukla, Varun Dutt, “A comparison study of abstractive and extractive methods for text summarization,” *Advances in Intelligent Systems and Computing*, 2021, to be published.

[4] S. Hima Bindu Sri and S. R. Dutta, “A Survey on Automatic Text Summarization Techniques,” *Journal of Physics: Conference Series*, vol. 2040, no. 1, Article ID. 012044, 2021. DOI: 10.1088/1742-6596/2040/1/012044.

[5] A. Parahonco and M. Petic, “How to create an adaptive e-learning system,” in *Proceedings of the 17th International Conference “Linguistic Resources and Tools for Natural Language Processing”*, (Chisinau and Online 10-12 November 2022), 2022, pp. 153–160. ISSN 1843-911X.

[6] A. Parahonco, M. Petic, and C. Negara, “The model of Web crawler for expansion the scope of initial search,” in *Workshop on Intelligent Information Systems: Proceedings WIIS2020*, (Chisinau, 04-05 December 2020), 2020, pp. 139–150. ISBN 978-9975-68-415-6.

[7] A. Parahonco and M. Petic, “ Elearning content processing situations and their solutions,” in *Proceedings of Workshop on Intelligent Information Systems: WIIS2022*, (Chisinau, October 06-08, 2022), 2022, pp. 154–159. ISBN 978-9975-68-461-3.

[8] Vishal Gupta and Gurpreet Lehal, “A Survey of Text Summarization Extractive Techniques,” *Journal of Emerging Technologies in Web Intelligence*, vol. 2, no. 3, 2010. DOI: 10.4304/jetwi.2.3.258-268.

[9] Chin-Yew Lin, “ROUGE: A Package for Automatic Evaluation of Summaries,” in *Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004)*, (Barcelona, Spain), 2004, pp. 74–81.

[10] A. R. Fabbri, W. Kryściński, B. McCann, C. Xiong, R. Socher, and D. Radev, “SummEval: Re-evaluating Summarization Evaluation,” T*ransactions of the Association for Computational Linguistics*, vol. 9, pp. 391–409, 2021. DOI: 10.1162/tacl_a_00373.

[11] S. Sripada and J. Jagarlamudi, “Summarization Approaches Based on Document Probability Distributions,” in *Pacific Asia Conference on Language, Information and Computation*, 2009, https://api.semanticscholar.org/CorpusID:1150351.

[12] A. Li, T. Jiang, Q. Wang, and H. Yu, “The mixture of textrank and lexrank techniques of single document automatic summarization research in Tibetan,” in *2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, IEEE, vol. 1, 2016, pp. 514–519.

[13] S. Siddika and M. S. Hossen, “Automatic Text Summarization Using Term Frequency, Luhn’s Heuristic, and Cosine Similarity Approaches,” in *2022 International Conference on Recent Progresses in Science, Engineering and Technology (ICRPSET)*, IEEE, 2022, December, pp. 1–6.

[14] H. P. Edmundson, “New methods in automatic extracting,” *Journal of the ACM (JACM)*, vol. 16, no. 2, pp. 264–285, 1969.

[15] H. Christian, M. P. Agus, and D. Suhartono, "Single document automatic text summarization using term frequency-inverse document frequency (TF-IDF)," *ComTech: Computer, Mathematics and Engineering Applications*, vol. 7, no. 4, pp. 285–294, 2016.

[16] Alexandr Parahonco and Mircea Petic, "Different aspects on extractive text summarization as a part of content generation for e-courses," in *Proceedings of Workshop on Intelligent Information Systems: WIIS2023*, (Chisinau, October 19-21, 2023), 2023, pp. 169–180. ISBN 978-9975-68-492-7.

Alexandr Parahonco[1], Mircea Petic[2]

Received September 30, 2023
Accepted December 12, 2023

[1] ORCID: https://orcid.org/0009-0007-3486-5597
Vladimir Andrunachievici Institute of Mathematics and Computer Science, Alecu Russo Balti State University
E–mail: alexandr.parahonco@usarb.md

[2] ORCID: https://orcid.org/0000-0001-6044-7646
Vladimir Andrunachievici Institute of Mathematics and Computer Science
E–mail: mircea.petic@math.md