Harnessing Key Phrases in Constructing a Concept-Based Semantic Representation of Text Using Clustering Techniques

Ali Mansour[1](✉), Juman Mohammad[1], Yury Kravchenko[1], Daniil Kravchenko[1], and Nemury Silega[2]

[1] Department of Computer Aided Design, Southern Federal University, 347900 Taganrog, Russia
mansur@sfedu.ru

[2] Department of System Analysis and Telecommunications, Southern Federal University, 347900 Taganrog, Russia

Abstract. This paper introduces a modified approach for representing text as semantic vectors, building upon the Bag of Weighted Concepts (BoWC) method developed in previous research. The limitations of the BoWC method are addressed, and a proposed solution is presented. Instead of using unigrams, the authors propose extracting key phrases that best represent each document to generate high-quality concepts and reduce concept overlap. These unique key phrases are then used to construct the concept dictionary. Document vectors are created by mapping document key phrases to the concept dictionary using a modified concept weighting function that considers the weight of the key phrase within the document. To evaluate the effectiveness of the resulting vectors, they were employed in a clustering task and compared against robust baselines. Experimental studies demonstrate that the proposed modifications enhance the quality of document vector representation, as evidenced by a minimum 4% increase in clustering accuracy based on the V1 metric.

Keywords: text mining · concept extraction · keyword extraction · key phrase extraction · document vectorization · text clustering · CF-IDF

1 Introduction

The rapid growth of textual data on the web and social media platforms necessitates the use of automated text mining techniques to extract valuable information from unstructured text and enhance user experiences by providing relevant information. In the text mining process, two key factors play a crucial role: the representation of text and the choice of text mining algorithm. Document representation, in particular, is a fundamental aspect that significantly impacts performance. Its objective is to convert documents into a machine-readable format through a process known as vectorization [1].

Great efforts have been made for a long time. First of all, the most classical method is Vector Space Model (VSM) [2]. Methods that adopt this model such as Term Frequency

Y. Hernández Heredia et al. (Eds.): IWAIPR 2023, LNCS 14335, pp. 190–201, 2024.
https://doi.org/10.1007/978-3-031-49552-6_17

Inverse Document Frequency (TF-IDF) and Bag of Words (BoW) represent a document as a vector in the vector space where each word represents an independent dimension [3]. However, these methods don't consider semantic relations among different words the size of such vectors increases according to the number of words used in the documents. This affects the efficiency of text mining algorithms and makes it difficult to capture good text features.

To model the sematic relations between words, like synonymy and polysemy, improved methods are proposed. Latent Semantic Indexing (LSI) [4] approximates the source space with fewer dimensions which uses matrix algebra technique termed SVD. Latent Dirichlet Allocation (LDA) can recognize the latent topics of documents and use topic probability distribution for representations.

In recent years, a lot of efforts have been made in the field of text representation using machine learning algorithms. One well-known method for distributed representations of sentences and documents, Doc2Vec is proposed by [5]. It is based on Word2Vec [6], which trains a distributed representation in a skip-gram likelihood as the input for prediction of words from their neighbor words [7] while Doc2Vec learns distributed vector representations for variable-length fragments of texts, from a phrase or sentence to a large document.

Although the vectors produced by these embedding techniques are of low dimensions and despite the success it has achieved in some tasks, the resulting feature vectors are ambiguous and it is difficult to explain the logic of the mining algorithms based on the extracted feature vector.

In this context, the conceptual representation of documents appeared as a solution to these drawbacks. It represents the document as a vector in which each concept represents an independent dimension. Such representation is considered a linear transformation from the space of words to the space of the concept which allows controlling the size of vectors. However, the quality of the vectors depends strongly on the way concepts are extracted and weighted.

Following this approach several works have been presented, the most famous of which are the Bag of Concepts (BoC) [8], as well as our Bag of Weighted Concepts method (BoWC) presented in previous works [9, 10]. Both methods create concepts by clustering word vectors into concept clusters, then uses the frequencies of these clusters to represent document vectors. To reduce the influence of concepts that appear in most documents, BoC uses a weighting scheme similar to TF-IDF, replacing the frequency of the term TF with the frequency of the concept CF.

In contrast in BoWC authors proposed a new concept weighting function which achieves a relative balance between common and rare concepts.

Also, unlike "BoC" approach, our "BoWC" method adopts term filtering in the preprocessing stage to reduce noise in clusters when forming the concept dictionary. This, in turn, means that the cluster centroid vector will become an accurate representation of the cluster (the concept) and sufficient to calculate similarity to the document thus reducing the computational cost of the clustering process and mapping the document's words to the concept dictionary.

Thanks to this optimization, our method outperformed many strong baselines for only 200 features. This confirms that the concepts in BoWC have much greater discriminatory power than BoC.

However, these improvements in creating the concept dictionary did not prevent the existence of noisy concepts (overlap between concepts). In both methods, the concept is a cluster of words, which have been grouped together on the basis of the similarity of their embedding vectors. Although the formation of the concept on the basis of single words (unigrams) is simpler, it is considered problematic due to the word polysemy. Especially since the word will belong to one concept, although it may appear in contexts with different connotations.

Regarding these shortcomings, we propose some improvements to the BoWC method by applying a key phrase extraction algorithm to use n-grams terms instead of unigrams in document representation and concept extraction processes. The key phrase can be understood as confirming the meaning of the word and relatively revealing its ambiguity by expanding it with neighboring words from the context.

The extraction of n-grams is carried out according to the FBKE (Frequency and Bert based Keyword Extraction) method presented in [11]. The motive behind choosing FBKE method is that it produces keywords that are similar to the context of the document, which ensures that these keywords are an accurate and sufficient representation of the document. The use of key phrases in conceptual formation is expected to reduce noise in clusters and produce cleaner concepts, as well as reduce the number of matching operations between the dictionary of concepts and the document. These advantages are supposed to reflect positively on the performance of document mining algorithms.

The contributions of this article are twofold. Firstly, a pre-processing algorithm is introduced, which aims to enhance the quality of concepts generated by the concept dictionary building algorithm. This algorithm represents the document using its most significant key phrases, taking into account both frequency and semantic similarity. The goal is to improve the accuracy and relevance of the resulting concepts.

Secondly, a modified concept weighting function is proposed. This function incorporates a novel weighting coefficient that captures the relationship between the key phrase and both the document and the concept. This coefficient can be combined with other weighting functions, such as CF-EDF and CF-IDF, that follow a similar approach to the proposed method. By incorporating this new coefficient, the concept weighting function becomes more comprehensive and adaptable, allowing for more accurate and context-aware concept representation.

This paper is organized as follows. In the Materials section, an explanation of the basic BoWC method and then proposed modifications are provided, in addition to a description of the experiments that were performed to test the performance of the method against other baselines. The analysis of the results is shown in the Results section, followed by expectations and future plans in the Conclusions section.

2 Materials and Methods

The implementation of the original BoWC method includes two stages[1]: the first is the concepts extraction which includes the text processing, the unique terms extraction, the word embeddings process, and finally, the word clustering process, which creates a dictionary of concepts. The second stage is documents vectorization, which includes mapping documents into the concepts dictionary to generate the vectors. In the modified method, the concept extraction algorithm is modified to include key phrases extraction and using them instead of unigrams. In the following the stages of the original BoWC method will be explained, concerning the position of the modification (Fig. 1).

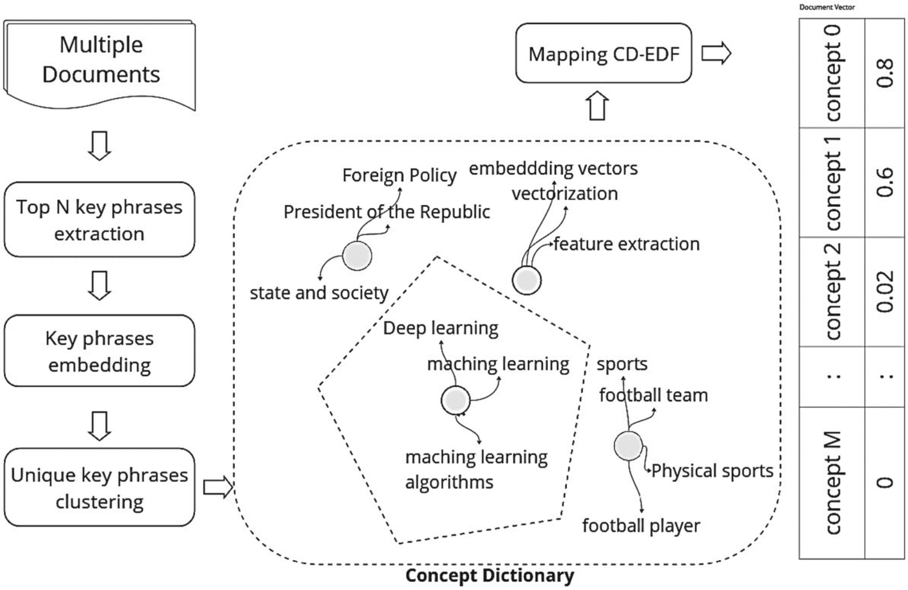

Fig. 1. General scheme of the modified BoWC method

2.1 Key Phrases Extraction

In "BoWC" each document is represented as a sequence of its words $d_i = \{w_1, w_2, w_3, \ldots, w_N\}$. The proposed modification in this work suggests that each document be represented by its most expressive key phrases. For this purpose, we apply the same methodology used in [11], which includes two phases: First, the selection of candidate key phrases, which is based on the frequency. The second stage is the weighting and ranking of the candidates using a function based on the semantic similarity between

[1] BoWC method implementation example with resources: BoWC-Method/codes/Github_FastText_BoWC.ipynb at main · Ali-MH-Mansour/BoWC-Method · GitHub.

the document and its key phrases. In this work, we will make a simple modification to improve the process of extracting key phrases, represented in the following algorithm.

Algorithm 1. *Text Preprocessing and document filtering*

```
Input    Set $\boldsymbol{D} = \{\boldsymbol{d_1}, \boldsymbol{d_2}, \ldots, \boldsymbol{d_N}\}$ of $\boldsymbol{N}$ documents
Output   $\mathbf{C}\boldsymbol{D}$ $a\ concept\ dictionary$
$\boldsymbol{D}^* = preprocessing(D)$
$\boldsymbol{CKW} = extract\_n\_grams(\boldsymbol{D}^*)$
$Freq_{UN} = extract\_frequent\_unigrams(\boldsymbol{D}^*)$
Foreach kw in CKW do:
    Kw_doc = parse_by_spacy(kw)
    Kw_root = get_root(kw_doc)
    If doc has compound dependency or kw_root in $Freq_{UN}$:
        KWS [] ← kw
    End
End
UKWS = set(KWS) //Select unique KW from corpus for clustering
UKWS_VECTORS = FASTTEXT_Encoder(UKWS)
$CD = \mathrm{sp}hericl(UKWS\_VECTORS) =$
$(kw_1^1, kw_2^1, \ldots, kw_{\mathrm{M}}^2, kw_1^2, kw_2^2, \ldots, kw_{\mathrm{M}}^2, \ldots, kw_1^K, kw_2^K, \ldots, kw_{\mathrm{M}}^K),$
```

We first extract different types of n-grams, then we identify the most frequent unigrams. Using SpaCy, we analyze the n-grams in which each unigram is found. We filter out n-gram words that are not frequent or do not form compound nouns, ensuring that the selected n-grams are meaningful keywords rather than arbitrary word sequences. Finally, we calculate the frequency for each type of n-gram, following a similar approach as the FBKE method.

$$TF_{n-gram} = \frac{n_t^i}{\sum_k n_k^i}, \tag{1}$$

where $\boldsymbol{n_k}$ refers to the total count of a specific type of n-gram (with i elements) in, while $\boldsymbol{n_t}$ represents the count of occurrences of token t in the document.

After that we choose top-N keywords to represent the document. These keywords are then encoded by an appropriate embedding model. Here for simplicity, we apply Fasttext [12]. Then FBKE selects the most relevant keyword phrases in the context of the document using the following weighting functions:

$$rel\left(d, c^j\right) = 2 \cdot \frac{tf_{c^j} \cdot S_{norm}^{j,d}}{tf_{c^j} + S_{norm}^{j,d}}, \tag{2}$$

where $S_{norm}^{j,d} = S^j \cdot e^{S^j/|c^j|}$ is the normalized value for similarity of the candidate c^j of the document d. S^j is the cosine similarity of the j-th candidate with the document. Tf is the key phrases frequency.

2.2 Concept Extraction

Aer obtaining the key phrases and their vectors for each document, the spherical k-means clustering algorithm is applied to these vectors. The output is a collection of concepts, where each concept is represented by M key phrases with similar meaning, such as synonyms, hyponym and hypernyms, which have been grouped together on the basis of the similarity of their vectors. The concept dictionary is presented as follows:

$$C = \left(kw_1^1, kw_2^1, \ldots, kw_{\mathrm{M}}^2, kw_1^2, kw_2^2, \ldots, kw_{\mathrm{M}}^2, \ldots, kw_1^K, kw_2^K, \ldots, kw_{\mathrm{M}}^K\right), \quad (3)$$

where kw_1^j is the jth cluster's (ith) key phrases. However, the concept vector is the centroid vector (the average vector of the concept's keyword vectors).

2.3 Document Representation

At this stage, the input data consists of the concepts dictionary and the documents represented by their embedding vectors. However, the extraction of key phrases during the concept extraction stage enables the modification of the CF-EDF weighting function, which can be expressed using the following formula:

$$BoWC_{c_i} = CF - EDF\left(c_i, d_j, D\right) \cdot e^{S_{c_i}} = \frac{n_{c_i}}{\sum_k n_k} \cdot e^{-\frac{|\{d \in D | c_i \in d\}|}{|D|}} \cdot e^{S_{c_i}}, \quad (4)$$

CF is concept frequency $\frac{n_{c_i}}{\sum_k n_k}$, where it is considered that the concept has appeared in the text when the value of similarity $\mathrm{S_c}$ between the centroid of the concept(s) with the document keyword exceeds a certain limit (Θ).

$$(\mathrm{S_c}) = \begin{cases} 1, \ \mathrm{S_c} > \Theta \\ 0, \ otherwise \end{cases}. \quad (5)$$

In the original method, S_{c_i}. is the average degree of similarity of the document's words with the concept to which the words belong. However, since the key phrases of the document are selected on the basis of frequency and semantic similarity between the phrase and the context of the document, the author proposes to replace the term S_{c_i}. With the average weights of the document's key phrases that belong to the concept given by the following formula:

$$S_i^j = \frac{1}{N} \sum\nolimits_{n=1}^{N} rel\left(c_{\mathrm{i}}, kw_n^j\right), \quad (6)$$

where N is the number of key phrases of the document (j-th)hat appeared in the current concept $\mathrm{c_i}$. The resulting weight expresses the relatedness between the key phrase and the document on the one hand, and between key phrase and the concept on the other hand. The final formulation of the concept weighting function according to BoWC becomes as follows:

$$BoWC_{c_i} = \frac{n_{c_i}}{\sum_k n_k} \cdot e^{-\frac{|\{d \in D | c_i \in d\}|}{|D|}} \cdot e^{S_i^j}, \quad (7)$$

As a result, the method generates a feature vector V for each given document d, where v_i^j reflects the importance (weight) of the i-th concept and k is the concepts quantity.

$$V^d = vectorization\left(\mathrm{C}, d^*, D^*\right) = \left\{v_1^d, v_2^d, \ldots, v_K^d\right\}, \tag{8}$$

Thus, we have two versions of the modified BoWC method: the first is only by changing the pre-processing stage and using keywords (key phrases) instead of unigrams while maintaining the same weighting function. The second includes the modification of the weighting function. In addition, the proposed weight S_i^j to the function of the "BoWC" method can also be used to weight the concepts in the "BoC" method, which follows the same approach.

2.4 Experimental Research

These experiments aim to check the representativeness of the resulting vectors compared to the original BoWC method and to a set of reliable baselines, namely: Bag-of-words (BoW), TF-IDF, averaged Fasttext (pre-trained and self-trained), Bag-of-concepts (BoC) detailed in previous works [9, 10]. Baselines used the same settings as in previous studies, and k-means algorithm was employed as the document clustering algorithm.

Datasets

To accomplish the document clustering tasks and assess the improved BoWC method, five text datasets were used (see Table 1).

Table 1. Overview of the used datasets

Datasets	Dataset information	
	# Documents	*# Classes*
BBC	2225	5
REUTERS (RE)	8491	8
OHSUMED (OH)	5380	7
20Newsgroups (20NG)	18821	20
WebKB	4199	4

BBC data set [13], comprises 2225 documents sourced from the BBC news website, encompassing news stories across five distinct topical areas during the period of 2004–2005.

The Reuters (RE) data set consists of articles obtained from the Reuters news feed. For this study, the R8 partition of the Reuters data set was utilized, which includes a total of 8491 documents.

The 20Newsgroups (20NG) dataset consists of 18,821 documents that have been classified into 20 distinct newsgroup categories, ensuring a roughly equal distribution among the categories.

The OHSUMED (OH) dataset is derived from a subset of clinical paper abstracts sourced from the Medline database. For this study, a partition containing 5380 documents from the OHSUMED dataset was utilized.

The WebKB dataset comprises web pages obtained from different sections of computer science, gathered through the World Wide Knowledge Base (Web->Kb) project conducted by the CMU text learning group [14]. These web pages have been categorized into seven distinct classes, including students, faculty, staff, and more. For this study, a preprocessed version of the WebKB dataset was employed, consisting of four different classes and a total of 4199 documents [14].

For the BoC method, which follows the same approach as the BoWC method, two versions of it will be tested, the first applies the CF-IDF weighting function and it is denoted as "$\text{BoC}_{\text{CF-EDF}}$" in the Table 2. The second version applies a modified concept weighting function expanded by the key phrase weight function used in BoWC method and it is denoted as "$\text{BoC}_{\text{CF-EDF-FBKE}}$" in the Table 2.

Documents are processed by lowercasing and deleting stop-words, symbols and numbers. FBKE method is used to extract key phrases, so for that n grams are extracted (with a length of 2–4) and then these phrases are filtered using Spacy to get only nominal phrases whose root is a word with a high frequency (Spacy is applied to n grams and not to the sentences). The embedding vectors were generated using the FASTTEXT embedding model with 300-word vectors. The word embedding model is trained on the data itself with a window size 15. The FBKE weighting function (1) is applied to select the top 20 most important key phrases to represent the document. The second step is to generate the concept dictionary by applying a clustering algorithm to the vectors of the key phrases with a similarity threshold of 0.6. Results may vary depending on the clustering algorithm used. In this work we use spherical k-means algorithm.

As mentioned, each cluster represents a concept. The threshold of similarity of the key phrase with the centroid of the cluster was set experimentally to 0.42. Document vectors are generated using the mapping function eq. (6) using [100, 200] concepts for each dataset.

For the evaluation metric, the V-measure evaluation metric was utilized. It is computed as the harmonic mean of homogeneity (H) and completeness (C) calculations [15].

$$V - measure = \frac{(1+\beta)\cdot \mathrm{H}\ \cdot\ \mathrm{C}}{(\beta\cdot \mathrm{H}\,) +\ \mathrm{C}}, \tag{9}$$

Here, completeness (C) represents the extent to which every member of a class is assigned to a single cluster, while homogeneity (H) indicates the degree to which each cluster exclusively contains members of a single class. The beta coefficient is set to one as the default value for this particular study.

3 Results and Discussion

This section discusses the test results that were performed with the aim of testing the quality of the vectors generated by our proposed methods as well as a set of robust baselines. Figure 2 shows the performance comparison of the original and modified

BoWC method when performing clustering tasks on five datasets. Overall, the modified BoWC (with only 100 concepts) is vastly superior to the original BoWC (with only 200 concepts).

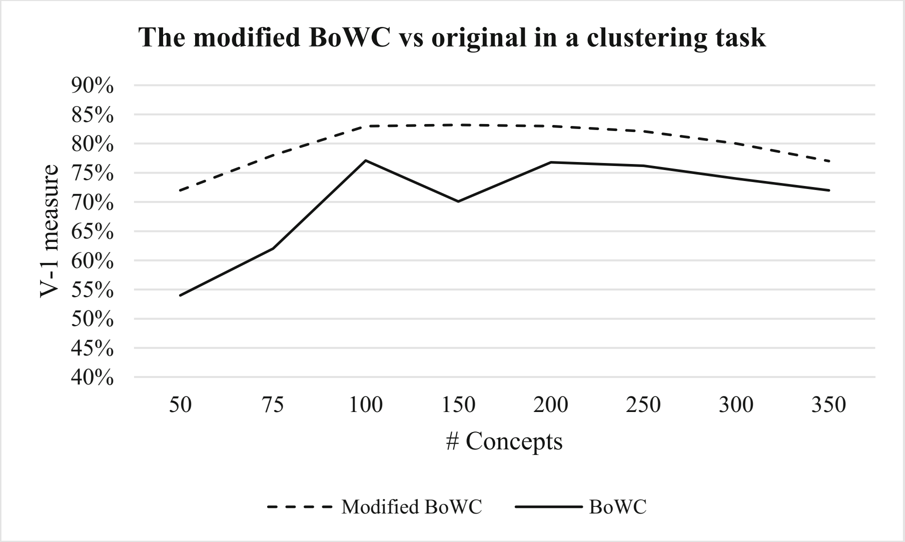

Fig. 2. The V1-score comparison between the modified and original methods in clustering tasks on BBC Dataset

It is also worth noting that the weighting of the concepts by means of the average weights of the keywords belonging to the concept led to the improvement of not only the “BoWC” method, but also the “BoC” method (Fig. 3, Table 2). This confirms the importance of this proposed coefficient in improving the quality of the resulting vectors and making them more discriminative.

Regarding the effect of the number of concepts on clustering accuracy, it is noted that BoWC begins to provide comparable performance to other methods for only 100 concepts (Fig. 2, Table 2). We can also notice that for a number of concepts greater than 200, the accuracy starts to decrease slightly, given that the concepts become overlapping and therefore non-discriminatory.

As a general note, the adjustments result in a significant improvement in clustering accuracy, which is up to 6% as with the BBC dataset.

Despite the good results achieved by our proposed encoding method compared to other approaches, it is still not efficiently applicable for encoding short texts. According to the findings in this research and previous studies that have addressed this method, it can be observed that the performance of the vectors generated by the BoWC method is better with longer texts. This is due to the ability to capture a larger number of distinctive concepts for each document and weigh them more effectively than in short texts.

However, the proposed concept extraction algorithm in this study offers applicability beyond document vectorization. It can be utilized in various fields, such as expanding

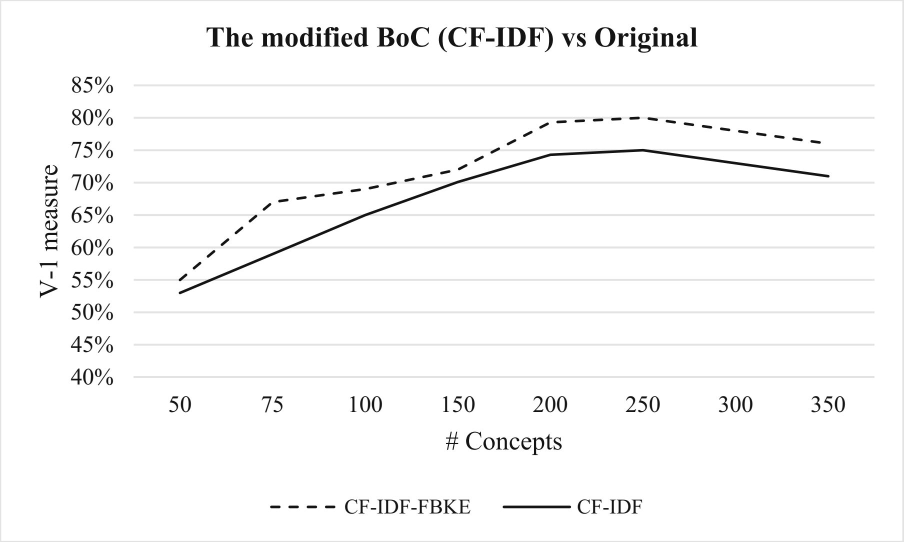

Fig. 3. The V1-score comparison between the modified and original BoC versions in clustering tasks on BBC Dataset

Table 2. The clustering results measured by v1-measure

	V1-score				
	BBC	RE	OH	20NG	WebKB
BoWC$_{\text{Fasttext}}$ (100)	0.768	0.591	0.129	0.316	0.318
BoWC$_{\text{Fasttext}}$ (200)	0.771	0.545	0.140	0.391	0.328
BoWC-FBKE (100)	0.830	0.623	0.154	0.414	0.390
BoWC-FBKE (200)	**0.832**	**0.64**	**0.155**	**0.426**	**0.391**
BoC$_{\text{CF\_IDF}}$ (200)	0.743	0.527	0.1	0.394	0.088
BoC$_{\text{CF\_IDF-FBKE}}$ (200)	0.793	0.59	0.13	0.400	0.2
TF-IDF	0.663	0.513	0.122	0.362	0.313
BoW	0.209	0.248	0.027	0.021	0.021
Averaged Fasttext (300)	0.774	0.481	0.109	0.381	0.219
self-trained Fasttext (300)	0.631	0.557	0.115	0.325	0.324

Self-trained vectors mean embedding models trained on the owned dataset (not pre-trained).

key phrases that describe a document by extracting keywords and phrases not explicitly mentioned in the text itself. Furthermore, the results obtained from the concept extraction algorithm can be employed in constructing interactive visualizations of document concepts, which prove valuable in tasks involving visual representation of documents and generating customized summaries for lengthy and multi-subject documents [16].

Additionally, the concept extraction algorithm and weighting functions proposed in this research can be utilized in developing user profiles to address content recommendation tasks.

4 Conclusions

This study introduced a modified approach for generating low-dimensional semantic vectors for documents, building upon the BoWC method from previous research. The key idea is to extract concepts from the text documents and utilize them to represent the documents through a mapping process using a semantic similarity measure. In this work, it was proposed to represent documents using key phrases extracted from them instead of single words. This approach resulted in reduced noise in the clusters, minimized concept overlap, and decreased the number of matching operations between the concept dictionary and the document. The quality of the concepts was validated through a text clustering task, where the generated vectors were compared against four strong baselines and the original method on five benchmark datasets. Future research will focus on a detailed analysis of the proposed weighting functions and their performance under different conditions. Additionally, efforts will be made to demonstrate the advantages and interpretability of the resulting vectors, highlighting their significance in solving document visualization tasks and other text mining tasks based on conceptual indexing.

Acknowledgment. The study was performed by the grant from the Russian Science Foundation № 23-21-00089, https://rscf.ru/project/23-21-00089/ in the Southern Federal University.

References

1. Bengforth, B., et al.: Applied analysis of text data in Python (2019)
2. Salton, G., Wong, A., Yang, C.-S.: A vector space model for automatic indexing. Commun. ACM **18**(11), 613–620 (1975)
3. Liu, Z., Lin, Y., Sun, M.: Representation Learning for Natural Language Processing. Springer Nature Singapore, Singapore (2020). https://doi.org/10.1007/978-981-15-5573-2
4. Deerwester, S., et al.: Indexing by latent semantic analysis. J. Am. Soc. Inform. Sci. **41**(6), 391–407 (1990)
5. Le, Q., Mikolov, T.: Distributed representations of sentences and documents. In: International Conference on Machine Learning (2014)
6. Mikolov, T., et al.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems (2013)
7. Taddy, M.: Document classification by inversion of distributed language representations (2015)
8. Kim, H.K., Kim, H., Cho, S.: Bag-of-concepts: comprehending document representation through clustering words in distributed representation. Neurocomputing **266**, 336–352 (2017)
9. Mansour, A., Mohammad, J., Kravchenko, Y.: Text vectorization method based on concept mining using clustering techniques. In: 2022 VI International Conference on Information Technologies in Engineering Education (Inforino). IEEE (2022)
10. Mansour, A.M., Mohammad, J.H., Kravchenko, Y.A.: Text vectorization using data mining methods. Izvestia SFedU. Tech. Sci. (2), 154–167 (2021)

11. Мохаммад, Ж, et al.: Метод извлечения ключевых фраз на основе новой функции ранжирования. Информационные технологии **28**(9), 465–474 (2022)
12. Bojanowski, P., et al.: Enriching word vectors with subword information. Trans. Assoc Comput. Linguist. **5**, 135–146 (2017)
13. Greene, D., Cunningham, P.: Practical solutions to the problem of diagonal dominance in kernel document clustering. In: Proceedings of the 23rd international conference on Machine learning (2006)
14. Craven, M., et al.: Learning to construct knowledge bases from the World Wide Web. Artif. Intell. **118**(1–2), 69–113 (2000)
15. Casey, K.: The V Metric, Menopausal Hormone Therapy, and Breast Cancer Risk. Yale University (2020)
16. Zhang, X., et al.: ConceptEVA: Concept-Based Interactive Exploration and Customization of Document Summaries (2023)