User's Research Interests Based Paper Recommendation System: A Deep Learning Approach

Betül Bulut, Esra Gündoğan, Buket Kaya, Reda Alhajj, and Mehmet Kaya

1 Introduction

The information in the world of science increases constantly. Scientists constantly make new scientific studies. Academic studies are converted into publications such as books, journals and conference proceedings. This rapid development in science has led to a rapid increase in the number of publications in the digital media. Researchers follow the studies about their study fields. Numerous databases are available to help researchers access information. The databases like Google Scholar, Science Direct and IEEE Digital Library are the most preferred databases especially in engineering. Increasing the number of digital libraries over the years and increasing the number of publications covered by the existing databases is a positive development for scientists. It is a positive situation for the researcher to have so many sources of information. However, it is difficult for the researcher to choose publications that may be useful to him in an excessive amount of publications in different sources. Filtering is essential for the researcher to find the most suitable publication from a database full of publications.

B. Bulut · E. Gündoğan · M. Kaya (✉)
Department of Computer Engineering, Fırat University, Elazığ, Turkey
e-mail: kaya@firat.edu.tr

B. Kaya
Department of Electronics and Automation, Fırat University, Elazığ, Turkey
e-mail: bkaya@firat.edu.tr

R. Alhajj
Department of Computer Science, University of Calgary, Calgary, AB, Canada

Department of Computer Engineering, Istanbul Medipol University, Istanbul, Turkey
e-mail: alhajj@ucalgary.ca

M. Kaya et al. (eds.), *Putting Social Media and Networking Data in Practice for Education, Planning, Prediction and Recommendation*, Lecture Notes in Social Networks, https://doi.org/10.1007/978-3-030-33698-1_7

There are article recommendation systems for researchers to facilitate publication. The most important aim of the article recommendation is to identify the articles that are closest to the researcher. Another aim is to minimize the time spent by the researcher to find the publication in which he or she is interested.

Most of article recommendation studies compare the content of the candidate articles. In the database, the articles that are most similar to each other are recommended to each other. These studies do not consider the researcher. The same articles are recommended to all researchers without paying attention to the researcher's previous studies, the study field. However, an article may not attract the interest of all researchers studying in that field.

In classical article recommendation systems, the researcher's publications or the study field is of no importance. The recommendation is made by accepting each researcher equally. In the proposed method, article recommendation is made by considering the features of the researcher. The difference of our study from other studies is that each user recommends the publication that is closest to the field of study taking into account the features of each researcher.

Our study aims to create a user-specific article recommendation system by considering the features such as the publications in the researcher profile and study field.

In our study, a recommendation system was created by using IEEE database data. It was performed with multiple similarity algorithms. The comparison of the user profile with candidate article to be recommended was first performed with the cosine similarity algorithm. Then, it was repeated by Doc2vec method which is one of the deep learning methods. It was tested with the voluntary participation of researchers with different degrees who work in different fields. It was seen that our study was more successful than other studies.

The rest of the paper is organized as follows. Section 2 offers an overview of article recommendation approaches. Section 3 describes the basic architecture and functioning of the system. Section 4 discusses results obtained from the recommendation system. Section 5 explains the conclusions.

2 Literature Review

Articles are prepared and published as text. Therefore, we can say that all studies conducted for the purpose of article recommendation are text processing or text mining. The common feature of all studies in the literature is that they do text processing. We used cosine similarity, TF-IDF and Doc2vec methods in our study. Doc2vec is a very popular machine learning method using recursive neural networks. It pays attention to the semantic meanings of words in text processing and considers synonymous words in text similarity studies. Deep learning methods provide more advanced technology and better results than classical machine learning methods.

There are many studies on article recommendation. Some of these studies are content-based, some are labeling-based and some are hybrid.

Although there have been many studies on labeling, the number of studies dealing with academic articles is relatively low. One of the outstanding studies on labeling and academic article recommendation method is the one proposed by Choochaiwattana [1] in 2010, and the other is Bahulikar's [2] study in 2017. Contrary to the known methods Ravi et al. [3] have proposed a model has that process text using recurrent neural networks and recommends academic articles accordingly. In addition, ontology-based academic article recommendation studies have also been conducted [4, 5].

There are also hybrid studies within academic article recommendation studies. While Lee et al. [6] have proposed a new model combining 2 different approaches, content-based and graph-based; Bancu et al. [7] have proposed a study that combines content-based and collaborative filtering methods. Apart from these, there are also approaches that combine two sets of data and use it as an academic article recommendation method, as is done by Zhao et al. [8].

West et al. [9] have indicated the references with nodes. They sorted the references with the Eigenfactor algorithm and clustered the knots using MapEquation. They found the most important node in the different steps of the hierarchy of these clusters. They have used an algorithm similar to PageRank to find this node. They have recommended the article of the most important node they found.

Xia et al. [10] have conducted a study that recommended articles with same authors for articles containing more than one author. Zhou et al. [11] have tried to list the confidence index of each academic article by suggesting a paper rank algorithm similar to the page rank algorithm, different from all studies and constructed the recommendation system on the data obtained from this. Drushku et al. [12] have proposed a collaborative method based on user interactions in their study. They set similarity criteria using a set of clustering techniques by looking at the user modelling of search engines. This study is similar to our study because similarity algorithms are successful and pay attention to the user profile.

Finding similarity of different articles with the keyword can be called a label-based similarity computation algorithm. Although researchers have suggested many methods in past studies, the most simple and easily integrated TF-IDF method was used for this method. There are many other studies [13–17] that use this method for the same purpose with us.

There are also similar recommendation systems [18–20] using bipartite graphs. Ohta et al. have proposed an academic article suggestion model using bipartite graphs.

Ashraf et al. [21] have taken the data from users on social media and recommended them according to the interest of the users. This study is similar to our study because it offers recommendation based on the user's profile. However, while our study recommends articles, this study recommends news. Our study is more complicated. It is more important because of its contribution to the advancement of science.

Watanabe et al. also have collected the information like the number of times an academic article was read except for its metadata and added them to the system of recommendation [22]. Similar to our study, the academic article recommendation model of Cui et al. [23] that also deals with the relationship among the users is one of few studies on this topic.

Xue et al. and Chen et al. have suggested one of the most related studies to our study because they use both online databases and analyze the user's published studies to determine the user's interest [24, 25]. The most original feature of the study of Xue et al. is that it implements the algorithm in a real online academic database. Chen and his colleagues have suggested that researchers should work with the vision that they can specialize in more than one field because of the fact that there is not such a limitation for them to specialize in just one field.

Bai et al. [26] have made a comprehensive review of the article recommendation systems. They have compared content-based, graphic-based methods, filtering methods and hybrid methods. Du et al. [27] have used the LDA (Latent Dirichlet Allocation) model to develop an article recommendation system based on their collaboration with co-authors of the articles. The study combining the Hadoop method with the LDA method is more successful than the collaborative filtering algorithm. However, this study does not pay attention to the other features of the articles. Therefore, our study is more successful than this study. Li et al. [28] have proposed an approach to e-commerce applications by including the user's social knowledge. Also they have created an effective confidence measurement model. Hebatallah and Hassan [29] have proposed a personalized research recommendation system based on user's explicit and implicit feedback. They have used recurrent neural networks in their studies. The difference between this study and our study, this study is based on the researcher's feedback, while our study looks at the researcher's publications. Karvelis et al. [30] have proposed the topic by using the Doc2vec method and the bag of words method. They have aimed to obtain a set of correct topic terms for library books. Similar to our study, the study is to use the Doc2vec method for recommendation. Our study is more comprehensive than this study as it recommends according to the user's profile. Rabindra Nath Nandi et al. [31] have applied Doc2vec method in their study. This study compares the results of Bag of words, LDA and LSA methods. Similar to our study, Doc2vec is used. The text is processed and recommended.

There are many studies on deep-learning text processing. Lee et al. [32] have made deep learning predictive text with deep learning. In the study, a film scenario has been created and the events that would be in the next scene with emotion analysis has predicted as text. Target and predicted emotion changes have been tested with cosine similarity. This study is similar to our study in terms of using Deep-Learning. The study by Qu et al. [33] aims to recommend on deep social media platforms with deep learning methods. The study makes friends recommendation using deep learning methods. Deep Graph-Based Neural Network (DGBNN) framework has been proposed.

3 Context

In this study, the most frequently used digital databases were used by the researchers in Computer Sciences, Electrical and Electronics Engineering and Mechanical Engineering to create the data set. Academic databases used as data set are:

1. Association for Computing Machinery (ACM) Digital Library
2. IEEE Xplore Digital Library
3. The DBLP Computer Science Bibliography

ACM Digital Library was used for the metadata such as title, author name, keyword, article date and abstract. IEEE Xplore Digital Library was used for the metadata such as title, author name, keyword, article year and abstract. The DBLP Computer Science Bibliography digital library was used for the metadata such as article year, author name, article title. IEEE Xplore Digital Library and ACM Digital Library data has never been stored in our system. When we make a user-based recommendation for the article, only the selected article, the user's publications, and the public information of the articles containing the same keyword with the selected article were taken over HTTP. This information was not kept after the recommendation. Article year, article title and abstract metadata were placed in separate sections in our temporary data repository according to the articles they belonged. As author name is the most important point in our article recommendation system, it is shown both in the metadata table and a separate table.

4 Methodology

According to user profile information, the article recommendation system consists of four basic structures. The data from the datasets are transmitted to the dataset interpreter. The dataset interpreter separates the user information data from the ACM Digital Library, the IEEE Xplore Digital Library, and The DBLP Computer Science Bibliography datasets. It parses the metadata of each article. It also parses and rearranges the metadata of the articles that is recommended as a result of the searched keyword. The parsed data are kept in Datasep Interpreter. Figure 1 shows the structure of the system architecture.

In the system that recommends article according to the user profile information, we can basically divide the data from the datasets into two parts:

1. A set of metadata that is in the researcher's profile including year, title, abstract, keywords of each of the articles that they have so far.
2. A set of metadata including year, title, abstract, keywords of the articles related to the researcher's field.

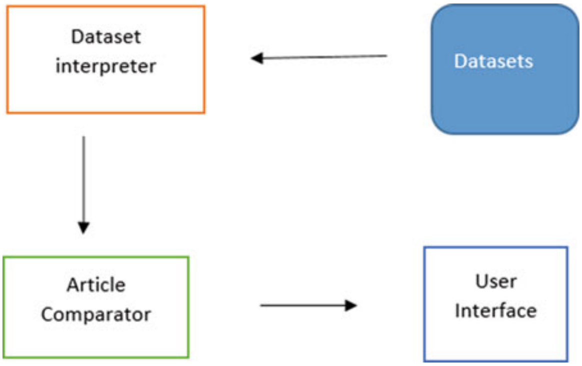

Fig. 1 System architecture

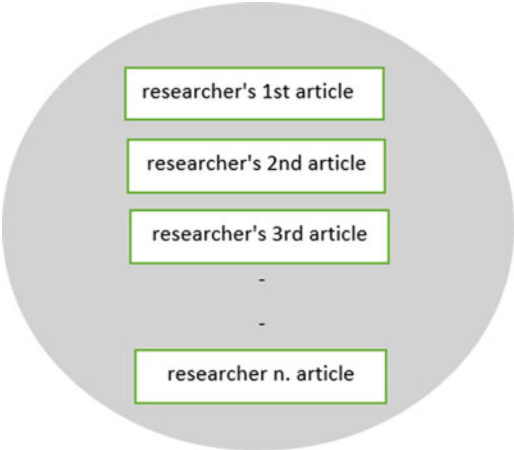

Fig. 2 Creating a profile from author's articles

In Fig. 2, the metadata of each article previously written by the researcher are combined. In this way the user profile is created.

$$info_n = title_n + year_n + authors_n + abstract_n + keywords_n \tag{1}$$

Equation (1) represents the sum of the metadata information of article number n. The formula that contains the metadata information of all the author's articles is as follows:

$$profile\_info = \sum_{k=0}^{n} info_n \tag{2}$$

Equation (2) is a string that is the sum of metadata information for all articles of the author. In other words, it is the profile information of the user. The profile_info value is the profile of the author whose article is to be recommended. $info_n$ value is the metadata information calculated for each article of the author expressed in Eq. (1).

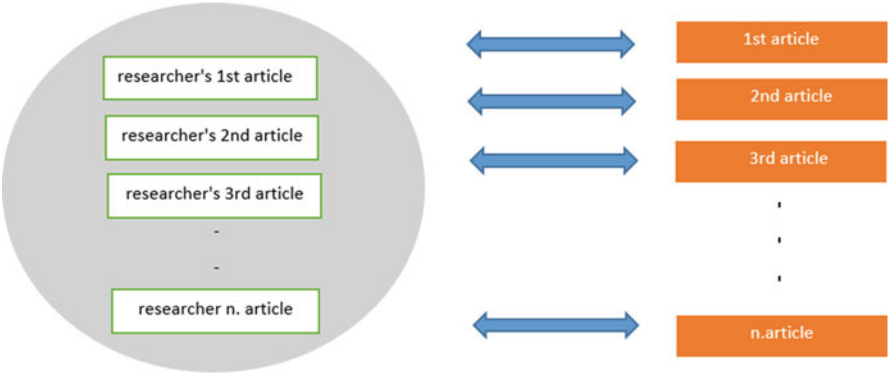

Fig. 3 Author profile and article comparison

The user's profile is created by combining the information of the n articles written by the researcher. The created profile and each of the articles are compared one by one. Figure 3 represents the comparison processes.

The processed data of the author's information and the results from the keywords are compared to the article. It compares the articles using comparative cosine similarity and TF-IDF methods. The article comparator compares the author information (*profile_info*) and each of the articles ($info_n$). The comparison result score of each article is determined. According to the comparison results from the comparator, the most related ten articles are presented to the user through User Interface.

We realized our study with more than one method. With the cosine similarity and TF_IDF method, we recommended article to the researcher and got the results. The second method of our study is the Deep Learning method which is the use of advanced technology, multi-level deep neural networks. It is very popular in high technology applications. As with machine learning methods, deep-learning method also learns system with training data. According to these data, it is obtained from the test data the results.

Gensim 3.4 and python 3 are the versions of the technologies used in our project. The Doc2vec method was used to find the most compatible article with the user's profile. For this purpose, the user's profile was created first. The user's profile consists of a combination of metadata such as the title, keyword and abstract of the articles of the user.

The Word2vec method breaks the given text into words and each word changes to a vector. In Word2vec, each word is a vector and the angle between the vectors of the words is looked at. These two words are similar to each other if the angle of the two vectors is close to each other.

The Doc2vec method is basically created based on the Word2vec method. However, in Word2vec, each word is expressed in a vector, while in Doc2vec, each text or document is a vector. The similarity between these two texts is determined

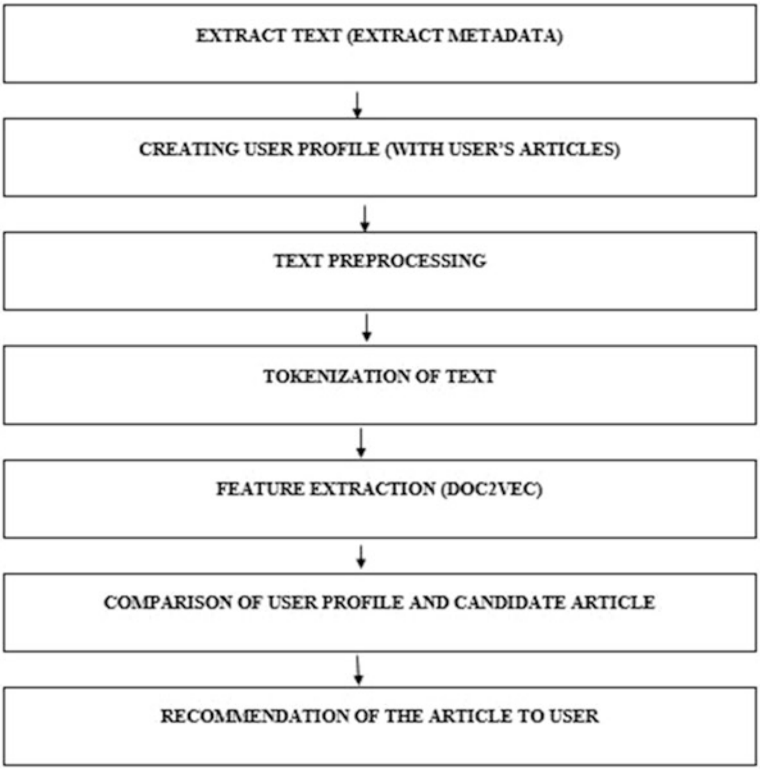

Fig. 4 User profile-based article recommendation system with Doc2vec

by looking at the degree of angle between the vectors of two texts. Figure 4 is the architecture of the article recommendation system developed with Doc2vec. The first step of the Doc2vec method is to split the text into tokens. Natural Language Toolkit platform was used for this in the study. The text containing the information obtained from the user's profile was divided into token by the NLTK tokenizer. The Doc2vec model was created with the data allocated to the tokens.

Vector_size refers to the dimensionality of the feature vectors in the Doc2vec method. In our study, we studied with more than one vector_size value. The most successful value was found to be vec_size = 10. The variable min_alpha is a float variable. It was seen that the learning rate decreased linearly to min_alfa during the study. Alpha is the first learning rate. The alpha value was selected as 0.025. The value of the epochs is the number of occurrences of the data on the corpus as the model is created. The epoch value was given as 100. The values selected for the variables are the values from which the most successful results are obtained.

All the words letters of the text we obtained from the metadata of the articles in the user profile reserved for the tokens were converted to lowercase letters. The

Doc2vec model was created with the obtained data and was recorded. The Doc2vec model obtained here can be defined as the vector of the user profile.

IEEE database was searched with the keyword taken from the researcher and the information such as title, author, keyword, and abstract of the articles was obtained. This information was indexed as elements of an array. Metadata was divided into token as it was when creating a user profile model. The data allocated to the tokens were converted to lowercase letters. The metadata of the article was converted to vector by Doc2vec method. In our study, each of these articles in the search results was expressed with a different vector. So, each element of the array was a separate vector.

The user profile vector and the vector of the article vectors are compared with the Doc2vec method. The vectors close to the user profile vector were articles that were recommended to the user. The comparison was performed with the most similar method of the Gensim library. This method takes the value of top-n as a parameter and the vector that is most similar to the number given at the value of the top-n is taken. In our study, we gave the value of top-n 10. We recommended 10 articles that were most similar to the user's profile.

5 Results

There are only a few studies that make article recommendations by paying attention to the information of the articles in the user's profile. In this study, what we tried is to determine the success criterion of the proposed method by using a few of the previously proposed methods.

In the first method, random email addresses and names of researchers were determined as experimental groups. The system sent the links and titles of ten articles that may interest the researchers based on their articles. The number of emails sent is ten, because it was anticipated that the researcher may not click on all links when a lot of articles are sent.

Table 1 shows the number of mails taken by the researcher, the number of mails read by the researcher, and the number of the researchers' own articles. The researchers read 47% of the mails and read 29% of the articles.

Another method is to conduct a survey with a research group. Name and surname of the researchers who will participate in the survey were taken. Recommended articles created from our system were taken from the system. Our system recommended 20 articles specific to each researcher. Each researcher who participated in the survey was presented the articles compiled for themselves in the form of a survey. In this two-choice survey, the researchers were asked to mark yes if the article interest them and mark no if not.

Table 2 shows the number of researchers participating in the experiment, the number of articles sent to the researchers and the answers of the researchers to this survey. The researchers have marked the articles that interest and do not interest them in the survey. According to the results, 426 of the articles submitted to the

Table 1 Experimental results

Number of emails sent to the researcher	Number of emails read by the researcher	Number of publications clicked by the researcher	Number of researcher's own publications
10	7	6	52
10	3	1	18
10	8	5	26
10	5	4	15
10	4	2	20
10	3	2	15
10	6	3	31
10	1	1	19
10	7	4	37
10	3	1	17

Table 2 Numbers of researchers and articles

Number of researchers participating in the survey	Number of articles presented in the survey	Number of articles marked as "Yes" by researchers in the survey	Number of articles marked as "No" by researchers in the survey
30	600	426	174

researcher were marked as yes, 174 as no. 71% of the articles attracted the interest of the researchers, 29% did not.

The Word2vec method represents a word with a vector. The Doc2vec method works on the same principle as the Word2vec method. Both methods are based on the semantic features of the words. Compared to the Word2vec method, the Doc2vec method can represent a whole sentence with a vector. In our study with the Doc2vec method, we showed each paragraph with a vector.

Doc2vec has two different models [31]. The first of these models is distributed memory model. This model is shown in Fig. 5.

In the Distributed Memory Model, D is the paragraph vector. W is the vector of words. Paragraph vector and word vectors are averaged. The fourth word is predicted with this average. The second model is the Distributed Bag of Words model. This model is shown in Fig. 6.

Paragraph1: "Wheather defines the state of the atmosphere. The weather is cold, rainy, dry, calm or windy. Air is a word that expresses all events taking place in the atmosphere layer."

Paragraph2: "All the natural phenomena occurring in the sky are air. All events such as rain, snow, cold, hot tornado, storm lightning define the weather."

Paragraph3: "Agriculture is the name given to the team to get the harvest of agricultural land. One of the most important branches of agriculture is organic agriculture."

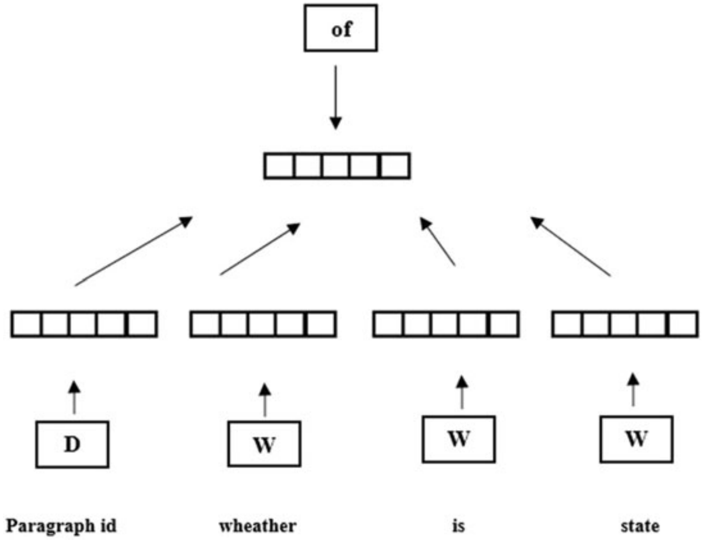

Fig. 5 The framework for distributed memory model

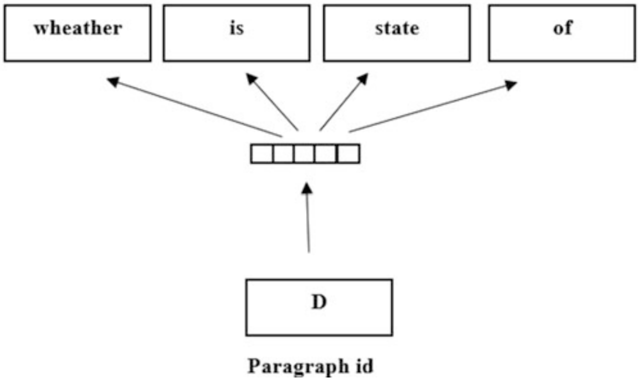

Fig. 6 The framework for distributed bag of words model

The above example is an example where the first and second paragraph should be closer to each other than to the third paragraph. In the Doc2vec model we created, these 3 paragraphs were compared. It was found that the first two paragraphs were similar to each other. In this model, the vectors of the words were combined to obtain the vector of the paragraph. We used Doc2vec's Distributed Bag of Words method in our study. The reason for this selection is that the Distributed Memory Model is useful in predicting missing words. However, there must be a vector of each paragraph to recommend article. The closest article is recommended to the researcher by comparing these vectors.

Table 3 Similarity score of two sample text in Doc2vec

Sample text I	Sample text II	Similarity score
Structured Query Language is a language that is designed to manage data stored in relational database management systems and is specific to an area being programmed	SQL is definitely not a programming language. It is used for database operations as a sub language. SQL is used to process data stored in the database. Enables data to be modelled within the intended task definition	0.363

Table 4 Similarity scores of articles with Doc2vec

User	The similarity score of 5 recommendation article per user
User1	0.727
User2	0.245
User3	0.123
User4	0.316
User5	0.211
User6	0.339
User7	0.218
User8	0.174
User9	0.120

Table 5 Success rates of Doc2vec method

Accuracy	83.4%
Precision	78.6%
Recall	47.2%
Specificity	96.7%
F1-measure	55.3%

In this study, the text obtained from the user profile consisting of 1805 words was used as the training data. The results of vector similarity for a sample text are shown in Table 3.

In this study, we selected 9 different authors and recommended 5 articles for each author. The average of similarity scores with the user profile of the 5 recommended articles is given in the Table 4.

The success rates of the Doc2vec method tested according to the survey method are shown in Table 5. According to Table 5, the accuracy rate of the system is 83.4%.

6 Conclusions

A In this study, ACM Digital Library, IEEE Xplore Digital Library, and The DBLP Computer Science Bibliography are used. The system has been tested by sending mail to the researchers. E-mails are sent in two ways. First, the researchers were sent e-mails and asked to click on these e-mails. Forty-seven percent of the researchers clicked on emails. In the second method, questionnaires were sent to the researchers via e-mail.

According to the survey technique, 71% of the emails attracted the attention of the researcher. According to these results, the survey technique is more successful than the technique based on clicking on the e-mails. Doc2vec method is a successful method for our article recommendation system with 83.4% accuracy. Unlike classical article recommendation systems, the study evaluates publications by the researcher and recommends publications that may be of interest to the researcher. The results of the system that make it are promising.

References

1. W. Choochaiwattana, Usage of tagging for research paper recommendation, in *ICACTE 2010 - 2010 3rd International Conference on Advanced Computer Theory and Engineering, Proceedings*, vol. 2 (IEEE, 2010)
2. S. Bahulikar, Analyzing recommender systems and applying a location based approach using tagging, in *2017 2nd International Conference for Convergence in Technology (I2CT)* (IEEE, Piscataway, 2017)
3. K.M. Ravi, J. Mori, I. Sakata, Cross-domain academic paper recommendation by semantic linkage approach using text analysis and recurrent neural networks, in *2017 Portland International Conference on Management of Engineering and Technology (PICMET)* (IEEE, 2017)
4. S.S. Weng, H.L. Chang, Using ontology network analysis for research document recommendation. Expert Syst. Appl. **34**(3), 1857–1869 (2008)
5. K.V. Neethukrishnan, K.P. Swaraj, Ontology based research paper recommendation using personal ontology similarity method, in *2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT)* (IEEE, 2017)
6. Y.C. Lee et al., Recommendation of research papers in DBpia: a Hybrid approach exploiting content and collaborative data, in *2016 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2016 - Conference Proceedings* (IEEE, 2017), pp. 2966–2971
7. C. Bancu, M. Dagadita, M. Dascalu, C. Dobre, S. Trausan-Matu, A.M. Florea, ARSYS—article recommender system, in *Proceedings - 14th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, SYNASC 2012* (IEEE Computer Society, 2012), pp. 349–355
8. W. Zhao, R. Wu, H. Liu, Paper recommendation based on the knowledge gap between a researcher's background knowledge and research target. Inf. Process. Manag. **52**(5), 976–988 (2016)
9. J.D. West, I. Wesley-Smith, C.T. Bergstrom, A recommendation system based on hierarchical clustering of an article-level citation network. IEEE Trans. Big Data **2**(2), 113–123 (2016)
10. F. Xia, H. Liu, I. Lee, L. Cao, Scientific article recommendation: exploiting common author relations and historical preferences. IEEE Trans. Big Data **2**(2), 101–112 (2016)
11. Q. Zhou, X. Chen, C. Chen, Authoritative scholarly paper recommendation based on paper communities, in *Proceedings - 17th IEEE International Conference on Computational Science and Engineering, CSE 2014, Jointly with 13th IEEE International Conference on Ubiquitous Computing and Communications, IUCC 2014, 13th International Symposium on Pervasive Systems* (IEEE, 2015), pp. 1536–1540
12. K. Drushku, J. Aligon, N. Labroche, P. Marcel, V. Peralta, Interest-based recommendations for business intelligence users. Inf. Syst. **86**, 79 (2019)
13. W. Zhang, T. Yoshida, X. Tang, A comparative study of TF$_*$IDF, LSI and multi-words for text classification. Expert Syst. Appl. **38**(3), 2758–2765 (2011)

14. J. Ramos, Using TF-IDF to determine word relevance in document queries, in *Proceedings of the First Instructional Conference on Machine Learning* (2003), pp. 1–4
15. T. Kenter, M. de Rijke, Short text similarity with word embeddings categories and subject descriptors, in *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management (CIKM 2015)* (ACM, New York, 2015), pp. 1411–1420
16. S. Albitar, S. Fournier, B. Espinasse, An effective TF/IDF-based text-to-text semantic similarity measure for text classification, in *International Conference on Web Information Systems Engineering*, vol. 8786 (2014), pp. 105–114
17. C.-H. Huang, J. Yin, F. Hou, A text similarity measurement combining word semantic information with TF-IDF method. Jisuanji Xuebao **34**, 856 (2011)
18. A.A. Müngen, M. Kaya, A novel method for event recommendation in meetup, in *2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)* (ACM, 2017), pp. 959–965
19. E. Gündoğan, B. Kaya, M. Kaya, Prediction of symptom-disease links in online helath forums, in *IEEE/ACM International Conference on Advances in Social Networks and Mining (ASONAM, 2017)* (ACM, 2017), pp. 876–880
20. S. Aslan, M. Kaya, Link prediction methods in bipartite networks, in *2017 International Conference on Computer Science and Engineering (UBMK)* (IEEE, 2017)
21. M. Ashraf, G.A. Tahir, S. Abrar, M. Abdulaali, S. Mushtaq, Personalized news recommendation based on multi-agent framework using social media preferences, in *2018 International Conference on Smart Computing and Electronic Enterprise (ICSCEE)* (IEEE, 2018)
22. S. Watanabe, T. Ito, T. Ozono, T. Shintani, A paper recommendation mechanism for the research support system Papits, in *Proceedings - International Workshop on Data Engineering Issues in E-Commerce, DEEC 2005*, vol. 2005 (IEEE, 2005), pp. 71–80
23. T. Cui, X. Tang, Q. Zeng, User network construction within online paper recommendation systems, in *Proceedings - 2010 IEEE 2nd Symposium on Web Society, SWS 2010* (IEEE, 2010), pp. 361–366
24. H. Xue, J. Guo, Y. Lan, L. Cao, Personalized paper recommendation in online social scholar system, in *ASONAM 2014 - Proceedings of the 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining* (IEEE, 2014), pp. 612–619
25. J. Chen, Z. Ban, Literature recommendation by researchers' publication analysis, in *2016 IEEE International Conference on Information and Automation (ICIA)* (IEEE, 2016)
26. X. Bai, M. Wang, I. Lee, Z. Yang, X. Kong, F. Xia, Scientific paper recommendation: a survey. IEEE Access **7**, 9324–9339 (2019)
27. G. Du, Y. Liu, J. Yu, Scientific paper recommendation: a survey, in *2018 IEEE Fourth International Conference on Big Data Computing Service and Applications (BigDataService)* (IEEE, 2018)
28. W. Li, X. Zhou, S. Shimizu, M. Xin, J. Jiang, H. Gao, Q. Jin, Personalization recommendation algorithm based on trust correlation degree and matrix factorization. IEEE Access **7**, 45451 (2019)
29. A. Hebatallah, M. Hassan, Personalized research paper recommendation using deep learning, in *UMAP '17 Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization* (ACM, New York, 2017), pp. 327–330
30. P. Karvelis, D. Gavrilis, G. Georgoulas, C. Stylios, Literature recommendation by researchers' publication analysis, in *2016 IEEE International Conference on Information and Automation (ICIA)* (IEEE, 2016)
31. R. N. Nandi et al., Bangla news recommendation using doc2vec. 2018 *International Conference on Bangla Speech and Language Processing* (ICBSLP), pp. 1–5, IEEE. 2018
32. S.-H. Lee, D.-M. Kim, Y.-G. Cheong, Predicting emotion in movie scripts using deep learning, in *2018 IEEE International Conference on Big Data and Smart Computing (BigComp)* (IEEE, 2018)
33. Z. Qu, B. Li, X. Wang, S. Yin, S. Zheng, An efficient recommendation framework on social media platforms based on deep learning, in *2018 IEEE International Conference on Big Data and Smart Computing (BigComp)* (IEEE, 2018)