CONTENT-BASED RECOMMENDER SYSTEM FOR SCIENTIFIC PUBLICATIONS

Milovan Milivojevic, Ph.D.[1], Srdjan Obradovic B.Sc.[1], Miklos Pot, M.Sc.[2], Dragoljub Drndarevic Ph.D.[1]
[1]College of Applied Sciences, Užice, Serbia, milovan.milivojevic@vtps.edu.rs
[2]Polytechnical Engineering College, Subotica, Serbia, pmiki@vts.su.ac.rs

***Abstract**: Content-based recommender systems (CBRS) in the last decades had been facing dynamic development in various domains from e-commerce and marketing to the film industry, the games industry, music, education, medicine and other fields. In the paper the CBRS for scientific publications focused on abstracts and keywords is presented. The developed model covers a wide range of processes and methods from natural language processing, feature selection, text-centric document vectorization, clustering methods and classification techniques. In addition to numerous paradigms of content similarity, the concept of diversification has been incorporated in order to prevent overspecialization. Developed solution provides: a scientific domain analyzer based on a database of scientific papers, recommendations base for existing papers and recommendations for paper classification into appropriate scientific fields. Software solutions were implemented using R and Python platforms. The developed model is validated through the case study database of scientific papers from the SED scientific conference. The obtained results show that the CBRS based on abstracts and keywords shows better performance compared to document search only by keywords.*

***Keywords:** content-based recommender system, content similarity, TF-IDF, clustering, classification, scientific publications*

1. INTRODUCTION

In the last three decades, the Recommender Systems (RS) have undergone rapid development. There is a large number of areas in which these systems have found their practical application. E-commerce, music recommendation, movies recommendation, targeted marketing, medical treatments, education, are just some of them. Modern RS solutions often include a large number of techniques and methods from natural language processing (NLP), data science (DS), machine learning (ML), and other artificial intelligence (AI) domains. RS applications are commonly encountered in the form of: Collaborative Filtering RS (CFRS), Content Based Filtering RS (CBRS), Knowledge Based RS (KBRS) and Hybrid Based RS (HBRS) solutions [1,2].
Learning individualized profiles from descriptions of examples (content-based recommending), allows a system to uniquely characterize each user without having to match his or her interests to another's. Content-based filtering approaches utilize a series of discrete characteristics of an item or features in order to recommend additional items with similar properties [3]. A feature is a distinctive attribute that can be used to measure a process under observation [4].
There are a many excellent books and papers in which CBRS advantages and drawbacks are explained in details [5-10]. But, to the best our knowledge, there are not so many references which deal with CBRS in the fields of scientific articles. Janach et al. in [11] uses publication reviews and classifies research in recommender systems both in the field of Computer Science and Information Systems for the 2016-2011 period. In [12], authors describe the recommender system, for digital libraries which help library users find the most relevant research papers for their needs. De Nart et al. in [13] explain advantages of CBRS in exploitation of large digital libraries compared with collaborative techniques. In their paper authors consider the domain of scientific publications repositories and propose a content-based recommender based upon a graph representation of concepts built-up by linked keyphrases. This recommender is coupled with a keyphrase extraction system able to generate meaningful metadata for the documents, which are the basis for providing helpful and explainable recommendations. To help authors decide where they should submit their manuscripts, Wang D. et al. in [14], present a Content-based Journals & Conferences Recommender System for computer science. Their system recommends suitable journals or conferences with a priority order based on the abstract of a manuscript. A web crawler is employed to continuously update the training set and the learning model. To achieve interactive online response, authors propose an efficient hybrid model based on chi-square feature selection and softmax regression. The test results which they accomplished show that, the system can achieve an accuracy of 61.37% and suggest the best journals or conferences in about 5s on average.
In this paper, we present a CBRS system that also aims to recommend scientific articles that are published at scientific conferences or submitted to a scientific conference. The system is based on natural language processing, feature

selection methods, vector space modeling of scientific articles in chosen feature space, methods of partitioning around medoids and hierarchical agglomerative clustering, and Multinomial Naive Bayes classifier.
Our hypothesis is that the CBRS based on features which integrate both manuscript abstracts and keywords is able to achieve better performance than scientific articles search systems based only keywords.
The developed content-based recommender system was tested on a case study of SED[1] scientific conference.

2. SYSTEM SETUP

The construction of quality CBRS implies the realization of several successive and concurrent iterative processes. Within this paper we present our content-based recommender system (SEccoR) in which is focused on the development of the item profile while the user profile is not shown in this paper. SeccoR is developed on the basis of literary resources, modern scientific achievements (Artificial Intelligence and Machine Learning) and open source libraries in the R and Python programming languages. The solution was created as a Recommender system for recommending scientific papers that are presented and published at scientific conferences. Schematic representation of the structure of such a system is given in Fig. 1. The developed system has three key functions: Domain Knowledge Classifier (7a / Domain Analyzer), Recommendation of Similar Content Based on user's query (7b / paper recommendations) (7b), and Solving Problems of Categorizing new scientific papers into the appropriate Scientific Category (7c / session recommendations).
The following text gives a brief description of the SEccoR Recommendation System.
Scientific conferences, observed at a defined time interval, are sources of documents (1) that are primarily text-centric. Today, papers presented at scientific conferences are mostly published in the form of Conference Proceeding. Conference proceeding are usually available online, so scientific articles from them can be automatically collected in repositories (databases) (3) using web crawlers as described in [14]. Scientific papers may have complex content of different nature: text, equations, images, tables, diagrams, links to other files and/or web pages. In recent times, scientific papers may also contain multimedia content such as videos, animations, etc. Pre-processing published scientific papers (4) involves at least two phases. In the first phase, manuscript decomposition is performed for several categories: text centric segments, graphic objects such as images or equations, multimedia content, links, etc. The second phase encompasses various subprocesses, which have been improved in the past decades by the world scientific community in the direction of identifying appropriate patterns (text, music, animation, URL address etc.) and separating characteristic sequences from them. For processing textual content of documents, NLP techniques are appropriate. Some of these techniques are: tokenization, stop-word removal, lemmatization, stemming, chunking, chinking, named entity recognition, etc. The result of these processes is a set of potential features that can describe these items in vector hyper-space. However, the number of features can be too large, and some of them may be uninformative noise that needs to be eliminated. Therefore, it is necessary to implement feature selection processes (5). A set of feature selection techniques which included filter, wrapper, embedded and/or hybrid methods [8-10], were selected as suitable for the SEccoR system by the authors of this paper. This approach was based on experience and validation results obtained on the dataset of items (scientific publications).

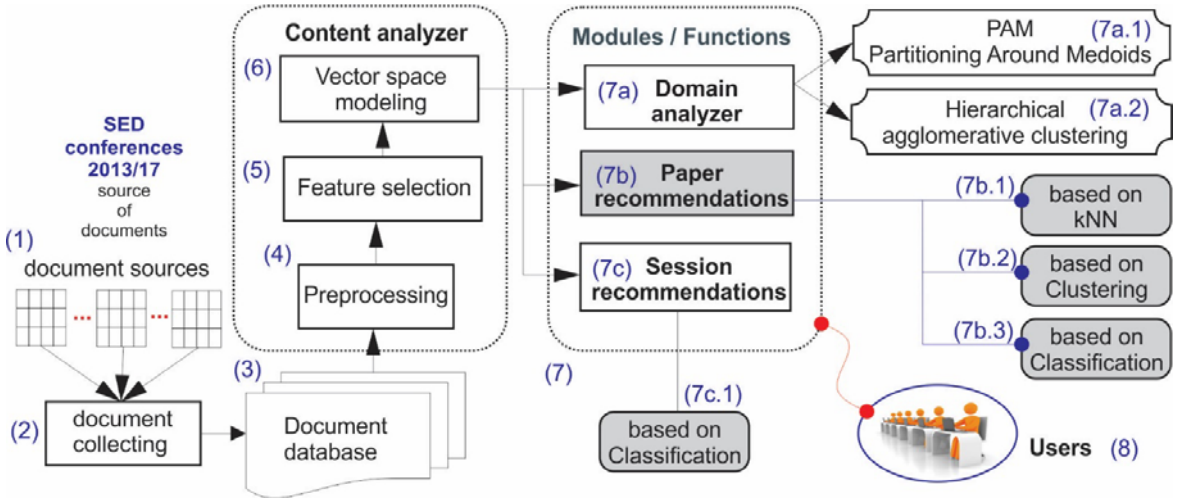


Figure 1: Structure of proposed Content-based Recommender System

As filter method representatives, methods based on chi squared (χ^2) and mutual information (MI) have been selected. In a subset of wrapper and embedding methods, more complex methods are included Boruta [15], xgboost model gain

[1] SED - Science and Higher Education in Function of Sustainable Development

[16] and DNN Gedeon [17] methods, using R packages, as well as a RFECV method (Feature ranking with recursive feature elimination) [18] developed in the scikit-learn Python package.
The usefulness of selected features is assessed using performance metrics of a Multinomial Naive Bayes classifier [19] for the task of predicting conference session affiliation for papers using available selected features. Multi-class F1 score [18] and accuracy are used as classifier performance metrics.
The selected Feature ranking methods perform dimensionality reduction and reduce the number of features to manageable size for processing and further analysis. The number of selected features is the number of vector space dimensions to which all items (documents) are mapped (6). Modeling items in vector space can be realized in different ways. The techniques applied in the SEccoR system are representing items as vectors in feature space using one of following schemes: 1) each item vector is composed from weighted feature frequencies, and the weights are produced by applying some feature ranking method, or an ensemble of base methods—by using some combination function on the ensemble, such as mean of feature importance produced from base methods; 2) each item vector is binary, in accordance with occurrence of respective features for each item; 3) each item vector consists of respective feature frequencies for each item; 4) vectorized item representations from schemes 2) or 3) are weighted using *term frequency-inverse document frequency* ($TF-IDF$) transformation [1].
Processes (3), (4) and (5) form the content analyzer of our recommender system.
SEccoR functions (7) are: *Domain analyzer*, *Paper recommender* and *Session recommender*.
Domain analyzer (7a) enables clustering of scientific papers (items, documents) based on different metrics and concepts of their similarity (by session, by scientific field, etc.). SEccoR facilitates, in a formalized way different similarity and distance metrics: euclidean, manhattan, dice, jaccard, simple matching, etc. [1]. Based on these, different distance and similarity matrices are formed, which are then used for K-medoids and Hierarchical clustering of scientific papers in vector space. Modul (7a.1) performs K-medoids clustering (PAM) [2], and module (7a.2) performs hierarchical agglomerative clustering [2].
Paper recommender (7b) is the primary function of the system. Active user reads (chooses) one document (scientific paper) according to his/her interests. Based on the paper's abstract and keywords SEccoR recommends a number of (k) additional papers. The recommendation can be produced using one of three following methods: 1) nearest neighbor approach (k-nn method) (7b.1) [2]; 2) one of two clustering methods (7b.2) (PAM or Agglomerative clustering) or 3) some classifier methods which can output class pseudoprobabilities (Multinomial Naive Bayes classifier, Logistic regression) (7b.3). Session recommender (7c) gives multiple session recommendations for the target paper based on classifier's softmax output, i.e. instead of outputting the predicted class label, the classifier outputs pseudoprobabilities of the item belonging to each of respective classes, and top k pseudoprobabilities are recommended (7c.1). Classifier input is the vector in selected feature space, derived using NLP methods on keywords and abstract of the new paper, previously unseen by the system. This automated paper classification based on its keyword and abstracts can be useful for conference editorial and organizational boards, and also reviewers.

3. METHODS

The developed system employs too many methods to be described here in full detail, so only some of them are presented in this section.
Filters based on the χ^2 statistic and Mutual information were used as measures of informativeness for ranking features in terms of their importance [1,5,7]. The result of the χ^2 filter is equal to Cramer's V coefficient between each feature f to be ranked/weighted and a categorical dependent variable c representing a class of items in some way. The $\chi^2_{(f,c)}$ statistic is calculated using a two-way contingency table, where n is the total number of items/observations, r is the number of contingency table rows, and c is the number of its columns.

$$Cramer's\ V = \sqrt{\frac{\chi_{(f,c)}}{n \cdot \min(r-1, c-1)}} \tag{1}$$

Mutual information MI, also known as information gain in literature, is used as an entropy-based filter method for feature ranking, defined for each class c and feature f as:

$$MI = H(c) + H(f) - H(c,f) \tag{2}$$

where $H(c)$, $H(f)$, and $H(c,f)$ are entropy of class c, entropy of feature f, and their joint entropy respectively.
Boruta algorithm was used as a wrapper feature selection method built around the random forest classification algorithm implemented in the R package randomForest [20]. The algorithm is described in detail in [15].
All methods of the SEccoR system are applied on items represented in vector space, and weighted using several vector weighting schemes. One of the weighting schemes we used is $TF-IDF$, where weights $w_{j,k}$ are defined for each feature f and each item vector i.

$$w_{j,k} = \frac{TF(f_j, i_k) \cdot IDF(f_j)}{\sqrt{\sum_{j=1}^{L} (TF(f_j, i_k) \cdot IDF(f_j))^2}} \quad (3)$$

Here $TF(f,i) = \frac{freq(f,i)}{maxOthers(f,i)}$ and $IDF(f) = \log(\frac{N}{n(f)})$, where $freq(f,i)$ is the number of occurrences of feature f in item i, $maxOthers(f,i)$ is the maximum frequency among all the features occurring in item i, N is the total number of items, and $n(f)$ the number of items in which the feature f appears, and L is item vector length, i.e. the number of features.

In order to give item recommendations for target item t, it is prudent to provide some measure of quality for potential recommendations p, that takes into account not only the similarity between items and potential recommendations, $Sim(t,p)$, but also aims to diversify the set of recommended items R, by rewarding the mean distance between a potential recommendation and items r_i previously recommended and added to set R. Musto et al. provide such a measure in [21], as:

$Quality(t,p,R) = Sim(t,p) * RelDiversity(p,R)$, where (4)

$$R = \varnothing \rightarrow RelDiversity(p,R) = 1; \quad R \neq \varnothing \rightarrow RelDiversity(p,R) = \frac{\sum_{i=1}^{m} (Dist(p, r_i))}{m} \quad (5)$$

As a measure of similarity $Sim(t,p)$, we used cosine similarity in $TF-IDF$ vector space, defined as:

$$Sim(t,p) = \frac{tgp}{\|t\| \|p\|} \quad (6)$$

4. CASE STUDY

The validation of the developed content-based recommender system SEccoR was realized through the Case Study of the international scientific SED conference, which has been held in the last ten years at the College of Applied Sciences Uzice (Uzice, Republic of Serbia).

4.1. Dataset

The database of documents used in the case study represents a collection of scientific papers published at the SED conference in the period from 2013 to 2017. Scientific papers were collected manually or from CD media (1,2), and loaded into the database (3). In this stage of the development of a content-based recommender, the authors have opted for a text-based analysis with a focus on paper abstracts and keywords. The database consisted of 302 scientific papers. These papers are, according thematic areas, pre-classified in eight conference sessions: Mechanical Engineering (MA) (37 papers), Information Technologies (IN) (63 papers), Ecology (ECO) (42 papers), Occupational Safety and Health (BZR) (6 papers), Civil Engineering and Architecture (GR) (29 papers), Management and Entrepreneurship (PM) (50 papers), Medicine and Healthcare (MED) (11 papers), Economics and Tourism (TTEC) (64 papers). For easier identification purposes the papers were given unique identifiers. For example, identifier *in15* denotes the 15th scientific paper in the Information Technologies conference session, while *ma24* denotes the 24th scientific paper in the Mechanical Engineering conference session, etc.

4.2. Content analyzer results and discussion

In accordance with the goal of this paper, for the given dataset, the construction of the SEccoR system is realized through three views of the input data. Following scenarios were considered: A) applying NLP and feature selection (FS) techniques only to abstract; B) application of NLP and FS methods only to keywords and C) application of FS over a coupled list generated by processes A and B.

Fig. 2 shows NLP and FS results for scenario A. Abstracts from 302 papers from SED conference from year 2013 to 2017 (1) were input for NLP processes (2) implemented using Python's spaCy [22] library. All papers are grouped into eight conference sessions.

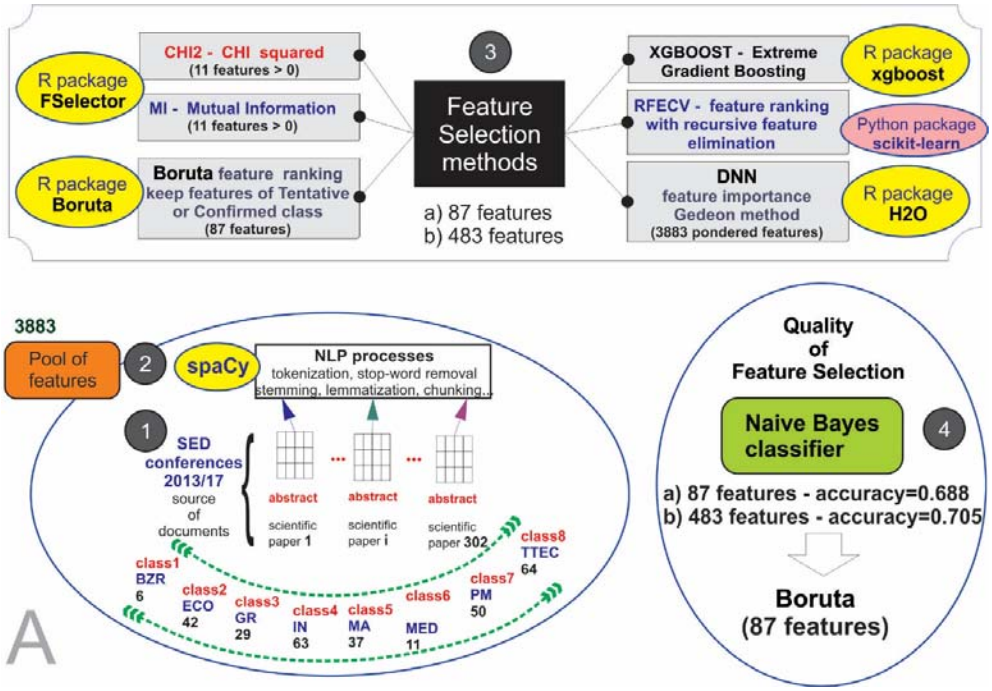


Figure 2: Scenario A: NLP and feature selection processes (FS) for abstracts

These processes resulted in a set of 3883 potential features. Wordcloud visualization of these features is given in Fig. 3. Feature ranking methods were carried out using filtering and wrapper methods (3). The basic characteristics of these methods are given in Table 1.

Figure 3: Scenario A: Features derived using spaCy NLP method.

Table 1: Characteristics of FS applied methods in SEccoR system

Method	Type	Package	Platform
χ^2	filter	FSelector	R
MI	filter	FSelector	R
Boruta	wrapper	Boruta	R
Xgboost Gain	wrapper	xgboost	R
RFECV	wrapper	scikit-learn	Python
DNN Gedeon	wrapper	H2O	R

Filter methods χ^2 and MI ranked only 11 features with importance above zero, while the Boruta algorithm kept 87 features as Tentatively or Confirmed important. These 87 features included all features ranked important by two filtering methods. We also extracted feature importance weights using xgboost's model Gain metric. Gain represents fractional contribution of each feature to the model based on the total gain of respective features splits. Recursive feature elimination wrapper method was employed using Python's scikit-learn library RFECV, and Gedeon method was used on a DNN model using the H2O R package. For these last three feature ranking methods, a heuristic of keeping as relevant only features above a certain cut-off point was chosen — $mean_{fimp} + 1.5 \cdot stdev_{fimp}$, where $mean_{fimp}$ and $stdev_{fimp}$ denoted, mean and standard deviation of respective models feature ranking vectors for 3883 features. Combined results from these three methods yielded too many features as relevant. After careful consideration, from these three methods we decided to only keep as relevant 483 features selected based on DNN Gedeon method. The quality of selected features was additionally tested on the session classification task using Multinomial Naive Bayes classifier (4) where Boruta and DNN Gedeon methods selected features as classifier inputs. We used simple accuracy as a performance metric. Based on a 70:30 train test split, the classifier achieved , $acc^{(87)}_{Boruta-test} = 0.688$ using Boruta

selected features and $acc_{H2O-test}^{(483)} = 0.705$, using DNN Gedeon method selected features. The obtained results show that the performance increase achieved with increasing the number of inputs from 87 to 483 is only slight (from 68.8% to 70.5% accuracy on the test set). Based on the previous, 87 features were selected as final for scenario A. In order to test the initial hypotheses of this paper, the methodology was repeated for scenarios B and C. In Table 2, due to scale, only some output results were provided.

Table 2: Characteristics of FS methods applied in SEccoR system

	Scenarios		
	A	B	C
Short description	items in **abstract** forms	items in **keywords** forms	A+B
NoF* after NLP processes	3883	1083	
			87+128
NoF from Chi squared, MI and Boruta methods (accuracy on test subset)	87 $acc_{Boruta-test}^{(87)} = 0.688$	35 $acc_{Boruta-test}^{(35)} = 0.59$	46 $acc_{Boruta-test}^{(46)} = 0.688$
NoF from xgboost gain, RFECV and DNN Gedeon (accuracy on test subset)	483 $acc_{Gedeon-test}^{(483)} = 0.705$	128 $acc_{Gedeon-test}^{(128)} = 0.625$	179 $acc_{Gedeon-test}^{(179)} = 0.7309$
NoF kept for further analysis (source)	**87** Boruta	**128** DNN Gedeon	
Final number of features			**179**

*NoF – number of features in subset

From Table 2, it can be seen that NLP processing produced 1083 features from paper keywords. Again, two feature selection methods were chosen as final for feature selection and based on them, and a 70:30 train-test data split, the above mentioned Multiclass Naive Bayes classifier achieved on test data— $acc_{Boruta-test}^{(35)} = 0.59$, and $acc_{xgbost-test}^{(128)} = 0.625$.

For further analysis 128 features, based on xgboost gain method were kept. As described above, in scenario C, 87 features based on scenario A and 128 features extracted from key words in scenario B are merged into one set. With this set of 215 features, the same methodology has been used for the selection of 179 features. The accuracy of the naive Bayes classifier, for the test set of items for this subgroup of 179 features was **0.7309** and was obtained using DNN Gedeon method. Classifier accuracy over the entire dataset of 320 papers was **0.84**. In our opinion this accuracy is satisfactory, and the selected **179 features** are selected for vectorization of items (scientific papers).

The vectorization of the item is realized by various methods described in section 3, and the results presented in this paper refer only to TF-IDF method.

4.3. SEccoR Function results

As described in Section 3, the first function of the SEccoR system is analysis of the scientific domain. The user can utilize one of two clustering methods for this purpose: Partitioning around medoids (K-medoids) and hierarchical agglomerative clustering. The user chooses the number of clusters (k) and a distance metric (cosine distance, Euclidean, Manhattan, Jaccard, Dice, etc.). After the items have been positioned into vector space of 179 features, results are visualized using first two principal components. Fig. 4 shows PAM clustering into three clusters using cosine distance matrix of all 302 vectorized papers.

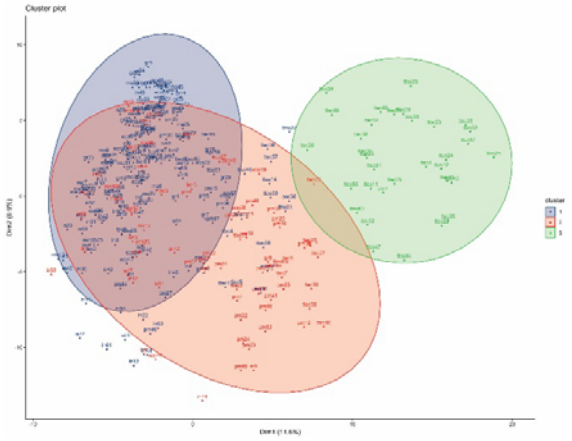
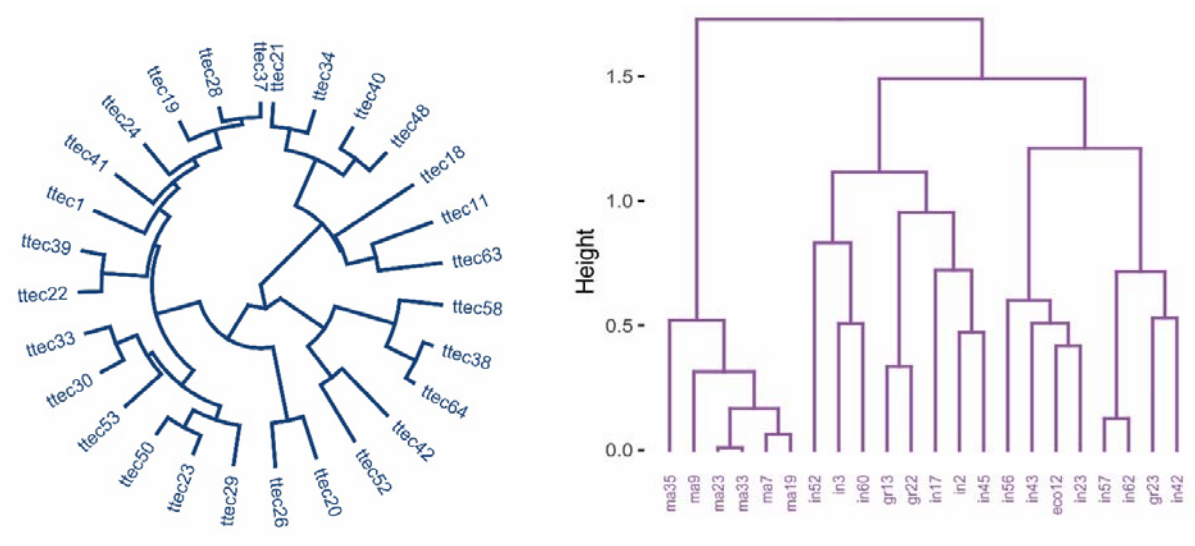


Figure 4: PAM clustering of scientific papers from SED

In Fig. 4, clusters can be loosely interpreted as follows: green cluster—economy and tourism papers; orange cluster—management papers; blue cluster—technical sciences. Visual overlap of clusters is a consequence of dimensionality reduction.

Fig. 5 shows a circular visualization of a cluster dendrogram resulting from hierarchical agglomerative clustering using ward.D2 agglomerative method. The dendrogram tree was cut into 8 clusters.

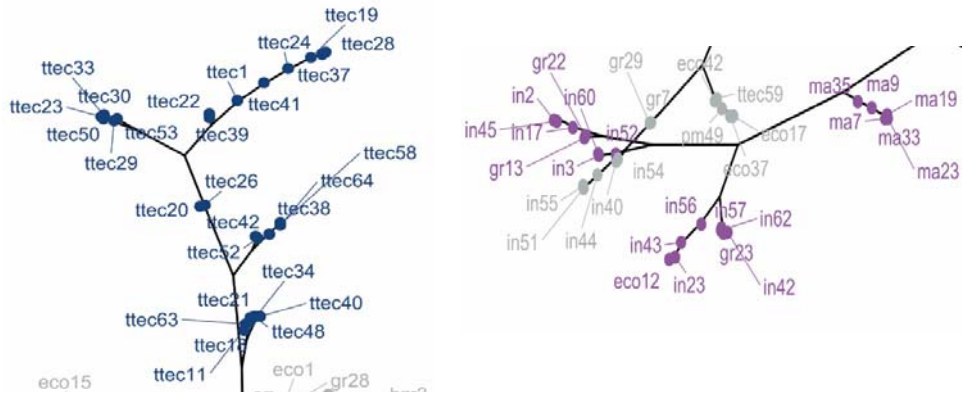
document database (k=3)



a) Cluster 3 b) Cluster 5

Figure 5: Hierarchical agglomerative clustering of scientific papers from SED document database (k=8) a) circular and b) standard dendrogram view

Fig. 6 shows the dendrograms as *phylogenic* trees.



a) part of cluster 3 b) part of cluster 5

Figure 6: Hierarchical agglomerative clustering of scientific papers from the SED document database (k=8) phylogenic trees a) part of cluster 3 b) part of cluster 5

As shown in Fig. 1, Paper recommender (7b) can give recommendations in three modes: using k-nearest neighbors method, clustering, or classification.
Based on k-nn method and diversification strategy described in Section 3, as an example, the following recommendations were given for papers ma33 and ma15, and displayed in Table 3.

Table 3: Example of SEccoR paper recommendations using k-nn method

Paper/ recommendation	MA15	MA33
1. recommendation	MA37	MA23
2. recommendation	IN4	MA35
3. recommendation	IN50	MA9
4. recommendation	TTEC2	IN23
5. recommendation	IN15	MA19

Recommendations based on the classification methods for existing items (Fig1, 7b.3), as well as recommendations for new items (Fig1, 7c.1), are based on the Multinomial Naive Bayes classifier algorithm. The difference is that for new items, it is necessary to first carry out Content Analyzer processes (Fig1). As an example, recommendations for a new scientific paper was carried out and results shown in Fig. 8.

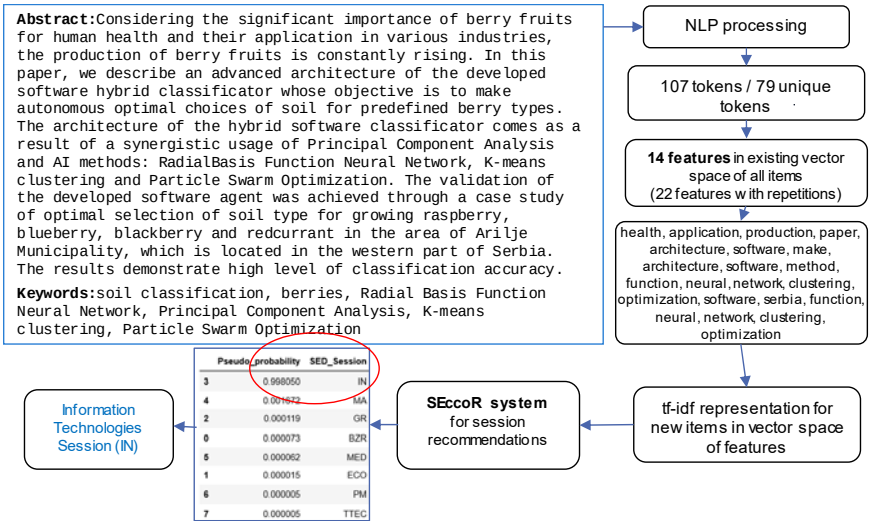


Figure 8: Session recommendation for a new scientific paper

For the example in Fig. 8, based on abstract and keywords, treated with the NLP process, 107 tokens are obtained, of which 79 are unique. Of those tokens, 14 where mutual with the vector space of the chosen 179 features, so the paper was transformed into TF-IDF form based on them. The SEccoR system based on Naive Bayes classifier recommends the Information Technologies Session (IN) for this new scientific paper.

5. CONCLUSION

Exponential growth of information in almost all domains is characteristic of human life at the end of the second decade of the XXI century. Under such conditions, software systems for recommendation of content play an increasingly important role as support for decision making. This paper presents a content-based recommender system for scientific publications focused on abstract and keywords. Developed CBRS includes wide areas expertise: NLP, feature selection, clustering and classification algorithms. In addition to numerous paradigms of similarity, the concept of diversification has been incorporated in order to prevent overspecialization. In the current development phase, developed CBRS, named SEccoR, provides: a scientific domain analyzer based on a database of scientific papers, recommendations base for existing papers and recommendations for paper classification into appropriate scientific fields. Software solutions were implemented in R and Python platforms. The results obtained through the case study for the SED Scientific Conference confirm the assumption that recommendations based on abstracts and keywords show better performance compared to document search based only on keywords. In future work, the author's attention will be directed towards improving the automated and optimal choice of the concept of similarity, applying a cross-validation strategy in selecting the number of clusters and developing a web crawler for the automatic collection of scientific articles from web and electronic sources.

REFERENCES

[1] AGGARWAL C. C., Recommender Systems: The Textbook, Springer International Publishing Switzerland, 2016.
[2] AGGARWAL C. C., Data Mining: The Textbook, Springer International Publishing Switzerland, 2015.
[3] MOONEY R. J., ROY L., Content-based book recommending using learning for text categorization, Proceedings of the Fifth ACM Conference on Digital Libraries, ACM, 2000, pp. 195–204.
[4] MWADULO .M. W., A Review on Feature Selection Methods For Classification Tasks, International Journal of Computer Applications Technology and Research Volume 5– Issue 6, 395 - 402, 2016, ISSN:- 2319–8656
[5] COVER T. M., THOMAS J. A., ELEMENTS OF INFORMATION THEORY, Second Edition, Published by John Wiley & Sons, Inc., Hoboken, New Jersey., 2006.
[6] MANNING C. D., *Stanford University*; PRABHAKAR RAGHAVAN, *Yahoo! Research*: HINRICH SCHUTZE *University of Stuttgart*, Introduction to Information Retrieval, Cambridge University Press, New York, 2008.
[7] CHAKRABARTI S., *Indian Institute of Technology, Bombay*, Mining the Web: Discovering Knowledge from Hypertext Data, Morgan Kaufmann Publishers An imprint of Elsevier Science, San Francisco, CA, 2003
[8] GIRISH CHANDRASHEKAR, FERAT SAHIN, "A survey on feature selection methods". Computers and Electrical Engineering, 2014.
[9] SAEYS Y., INZA I., LARRANAGA P., "A review of Feature Selection techniques in bioinformatics". Bioinformatics, Oxford University press, 2007.
[10] PERVEZ M. S., FARID D. M., Literature Review of Feature Selection for mining Tasks, International Journal of Computer Application, Vol 116, No. 21.
[11] JANNACH D, ZANKER M., GE M., GRONING M., Recommender Systems in Computer Science and Information Systems – A Landscape of Research, C. Huemer and P. Lops (Eds.): EC-Web 2012, LNBIP 123, pp. 76–87, 2012. Springer-Verlag Berlin Heidelberg 2012.
[12] SIMON P., SHOLA P.B., ABARI O. J., Application of Content-Based Approach in Research Paper Recommendation System for a Digital Library, International Journal of Advanced Computer Science and Applications, Vol. 5, No. 10, 2014.
[13] DE NART D., TASSO C., A Personalized Concept-Driven Recommender System for Scientific Libraries, Procedia Computer Science 38 (2014) 84 – 91.
[14] WANG D., LIANG Y., XU D., FENG X., GUAN R., A content-based recommender system for computer science publications, Knowledge-Based Systems 157 (2018) 1–9.
[15] KURSA, M. B., & RUDNICKI, W. R. Feature selection with the Boruta package. J Stat Softw, 36(11), 1-13., 2010.
[16] CHEN, T., & GUESTRIN, C., Xgboost: A scalable tree boosting system. In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining (pp. 785-794). ACM. 2016.
[17] GEDEON, T. D., Data mining of inputs: analysing magnitude and functional measures. International Journal of Neural Systems, 8(02), 209-218. 1997.
[18] https://scikit-learn.org/stable/modules/feature_selection.html#rfe (last accessed 18.05.2019)
[19] KIBRIYA, A. M., FRANK, E., PFAHRINGER, B., & HOLMES, G., Multinomial naive bayes for text categorization revisited. In Australasian Joint Conference on Artificial Intelligence (pp. 488-499). Springer, Berlin, Heidelberg, 2004.
[20] LIAW, A., & WIENER, M., Classification and regression by randomForest. R news, 2(3), 18-22, 2002.
[21] MUSTO, C., et al, Personalized finance advisory through case-based recommender systems and diversification strategies, Decision Support Systems 77: 100-111, 2015.
[22] https://spacy.io/ (last accessed 11.05.2019)