The 15th International Scientific Conference
eLearning and Software for Education
Bucharest, April 11-22, 2019
10.12753/2066-026X-19-000

SEMANTIC AUTHOR RECOMMENDATIONS BASED ON THEIR BIOGRAPHY FROM THE GENERAL ROMANIAN DICTIONARY OF LITERATURE

Laurențiu-Marian Neagu, Teodor-Mihai Coteț, Mihai Dascălu, Ștefan Trăușan-Matu
Computer Science Department, University Politehnica of Bucharest, 313 Splaiul Independentei, Bucharest, Romania
laurentiu.neagu@cti.pub.ro, teodor_mihai.cotet@stud.acs.upb.ro, {mihai.dascalu, stefan.trausan}@cs.pub.ro

Laura Badescu, Eugen Simion
The "G. Călinescu" Institute of Literary History and Theory, Romanian Academy, Calea 13 Septembrie, Bucharest, Romania
laura.e.badescu@gmail.com, eugen.ioan.simion@gmail.com

***Abstract**: The General Romanian Dictionary of Literature is a centralized text repository which contains detailed biographies of all Romanian authors and can be used to perform various subsequent analyses. The aim of this paper is to introduce a novel method to recommend authors based on their biography from the General Romanian Dictionary of Literature (DGLR). Starting from multiple input files made available by the "G. Călinescu" Institute of Literary History and Theory, we extracted relevant information on Romanian authors covering the [A-D] letters which was indexed into Elasticsearch, a non-relational database optimized for full-text indexing and search. The relevant information considers author's full name, their pseudonym (if any), years and places of birth and of death (if applicable), brief description (including studies, cities they lived in, important people they met, brief history), writings, critical references of others, etc. The indexed information is easily accessible through a RESTful API and provides a powerful starting point which may contribute to future Romanian cultural findings. Our aim is to create an interactive map showing all Romanian literature contributors by enabling the identification of similarities and differences between them based on specific features (e.g., similar writing styles, time periods, or similar text descriptions in terms of semantic models). In order to have a clearer image on how authors relate one to another, we employed the kNN algorithm on a set of integrated features covering authors' descriptions transposed in a reduced fastText embedding space, overlap of biographic refences and professions, as well as closeness in terms of publishing periods. This paper is a proof of concept that makes use of only the first two volumes of DGLR and represents the first step for follow-up analyses performed using the indexed dictionary.*

***Keywords:** Clustering; Text Categorization; Text Mining; Analysis of General Romanian Dictionary of Literature; Author Recommendations; Adaptive Technologies.*

I. INTRODUCTION

The challenging effort for creating the General Romanian Dictionary of Literature (also referred to as "Dicționarul General al Limbii Romane", or DGLR) aims to cover the specificities of Romanian literature, from authors, publications, groups, to literary movements, anonymous writings or translators. The Dictionary addresses the Romanian literary phenomena which happened inside or outside Romanian borders, from the beginning of Romanian writing to present date, while retaining one important inclusion criteria: the writing needs to serve the Romanian literature; thus, the dictionary includes only entities who published qualitative work written in Romanian.

Our work aims to determine the similarities between the authors who contributed to the Romanian Literature based on their descriptions from DGLR, thus facilitating the construction of a a virtual map of Romanian authors with semantic connections and corresponding distances. This study targets to support education, more precisely it consists of Natural Language Processing (NLP) techniques that can be applied to the educational domain field in order to make recommendations of writings of different authors (poetry, poems, articles etc.) based on the literature read by users. The

representation within semantic spaces based on author descriptions supports generating personalized recommendations of specific authors to learners. The current research targets the eLearning domain and can be applied to educational scenarios that require the visualization of inter-textuality links between Romanian authors based on their DGLR descriptions, as well as scenarios requiring recommendation of similar authors based on multiple criteria (e.g., time period, importance, similar profession, bibliographic references).

The paper starts with presenting the motivation of the study, the work being conducted by a team of researchers on Machine Learning and a team from the Romanian Academy, with the goal of determining potential similarities between authors based on their descriptions from DGLR. The State-of-the-Art section contains technologies used to achieve this goal and presents a snapshot of the research conducted in this area, with similar solutions used for exploring semantic similarities between text documents.

The Method section presents a step-by-step approach of the entire processing pipeline, from describing the used corpora, to text parsing, indexing, pre-processing, computing semantic similarity, extracting features, and reducing space dimensionality. The following section presented the obtained results, with corresponding plots and discussions. The last section presents the concluding remarks and opens the discussions for further work which can be conducted in this field.

II. STATE OF THE ART

In the field of cognitive science, modeling the semantic similarity between documents and texts in general is of particular interest, both from theoretical and practical reasons, and several approaches were introduced. As presented in an empirical evaluation performed by Michael D. Lee [12], models can consider simple word-based, keyword-based and n-gram measures, but also some more elaborated approaches, like Latent Sematic Analysis (LSA) [9]. As the study revealed, the most complex one at that time, LSA, has proven to be the most efficient under the conducted experiment, followed by the keyword-based and n-gram models. LSA is considered a variant of the vector space model that uses a reduced-rank approximation to the term-document matrix [7]. The technique implies the creation of a large matrix with term-document associations and the construction of a semantic space in which closely associated terms and documents are placed near each other in the semantic space.

Related approaches for extracting features for documents were explored by Lilleberg [10] who ran several experiments for features extraction using word2vec [13], term-frequency inverse document-frequency (Tf-Idf) [11] or both combined, with or without stop words. Each method was evaluated for their supervised text classification task and this provides us insights on which approach is most adequate for our task. A simple average over all (including stop words) word2vec embeddings rendered an accuracy of .841, while an average of the same word2vec embeddings weighted by Tf-Idf scores without stop words produced a considerably higher accuracy, namely .895. The highest accuracy of .897 was achieved by using a concatenation of the Tf-Idf vector representation with the word2vec embeddings weighted by the same Tf-Idf scores, both representations excluding stop words. However, we decided to use the simpler method for our task because the difference in accuracy was of only .002.

Common procedures of measuring distances between data points involve local learning through affinity functions. Clustering involves creating groups of similar points by computing the distances between points and determining similar neighbors. Two of the most frequently employed algorithms are kNN (*k* Nearest Neighbors), used mainly on supervised learning, and k-Means applicable on unlabeled data [17]. Popular among clustering algorithms is also affinity propagation, which implies message exchange between data points, does not require as input the estimated number of clusters (like kNN or k-Means), and was proved to be effective for different datasets [3].

III. METHOD

Figure 1 introduces the processing pipeline used to establish the similarities between authors, which primarily takes into account the descriptions of each author, combined with the quotes about the corresponding author (if any), all texts beings represented in the embeddings space. Additional features were also extracted from the corpus, namely publishing years, bibliographic references, and author profession. The last steps from the pipeline consider determining the distances between authors and establishing the most similar authors to any given one.

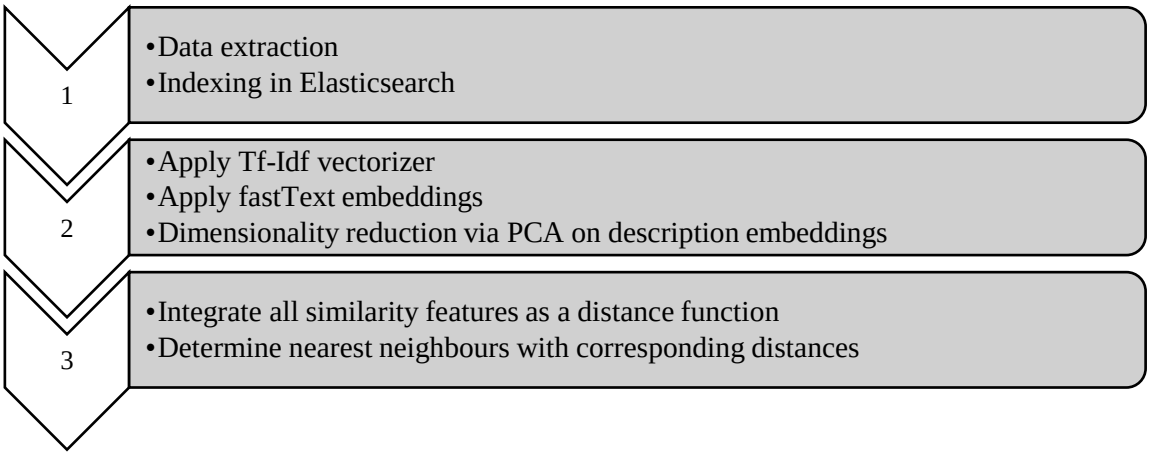


Figure 1. Processing pipeline.

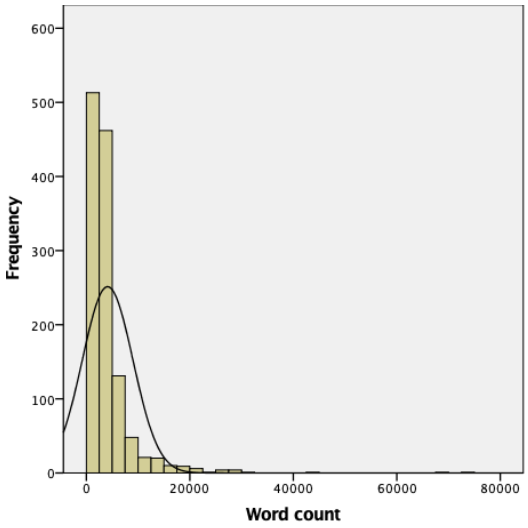
3.1 Corpus

Our text corpus is monolingual, written in Romanian language. DGLR is divided by letters (from A to Z in terms of authors' surnames or entities' names) into volumes. Our work included the parsing of 33 source files in Adobe InDesign format, converted into HTML, which cover the Romanian authors and publications ranging from A to D from the first two volumes. We identified that the authors from each HTML file have two standard node structures: one for the more important or well-known authors, and one for the others. While considering common entities, we identified two distinct sub-categories: one referring to authors and one targeting publications. Only authors were taken into consideration for the further NLP processing.

Even though the HTML tag classes from the files were mostly standardized, there were few entities that could not be matched using the criteria for parsing. For those, manual follow-up analyses will be conducted. The distribution of authors is as follows:

- 355 well-known authors, out of which 3 failed to be parsed;
- 1464 other entities, from which 881 were labeled as authors, 455 as publications and 128 failed to be parsed (being either an author or a publication).

Thus, the total number of authors that were processed is 1233. Each author is described using 4 sections. The first one is a description of the author's life and writings. The second one contains the publications of the author. The third one involves the publications in which the author was referenced. These first three sections are always present; conversely, the fourth one contains quotes about the author, and it is empty for most of the authors. Besides considering the explicit markers of "important author", the authors' importance is also reflected in the number of words from their description (see Figure 2 for plotted distribution using IBM SPSS Software). As expected, the word count is a heavy-tailed distribution, with few authors exceeding 5,000 or 10,000 words [16].



N	1,233.00
Std. Deviation	4,889.28
Skewness	6.22
Kurtosis	64.94
Minimum	102.00
Maximum	73,389.00
Percentiles	
Q1	1,809
Q2	2,850
Q3	4,512

Figure 2. Histogram of word counts.

3.2 Data extraction and indexing

The input HTML files were parsed, and each identified author was tagged as being an "important entity" or not, based on the formatting of the HTML tag. Starting from the authors' descriptions, several elements were extracted: year and place of birth, year and place of death (if the case), professions, bulk text biography, lists of writings and corresponding array of publishing years, together with critical references made by other authors or critics. Data was indexed and stored in an Elasticsearch server, which is proven to be useful for distributed search and as an analytics engine [6; 14]. The goal of this step was to extract from the parsed data only relevant atomic information useful for follow-up stages.

3.3 Tf-Idf vectorizer, fastText and dimensionality reduction on descriptions

First, the descriptions of authors are concatenated with the quotes about them. Afterwards, Tf-Idf is applied using the scikit-learn implementation[1] and spacy[2] is used for all language specific tasks (tokenization, computing lemmas, determining stop words, etc.). Only lemmas of the words were taking into consideration and stop words were removed in order to improve the feature extraction process, in line with the state-of-the-art approaches. The generated Tf-Idf matrix was further used to weight each word embedding given by fastText. The current implementation relies on fastText pre-trained word embeddings with 300 dimensions [4] that are averaged across each description.

Second, the vectors resulted from the previous step were projected into 17 dimensions using a Principal Component Analysis (PCA) [8]. The first 17 components were selected because the rest of the components captured less than 1% variance of the original vectors. The projected space retains 58% variance. We will further call these vectors *author embeddings*, as they are an orthogonal vector representation of the authors. The vectors were further projected into 3 dimensions by using a second PCA or t-SNE (t-Distributed Stochastic Neighbor Embedding) available in the TensorBoard toolkit [5]. The publishing years and bibliographic references could not use these features in the visualization as they are categorical types of data.

[1] https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html
[2] https://spacy.io/

3.4 Integrate the features and identify similar authors

Three addition features besides author embeddings are considered:

- *Publishing year* – the mean year of publishing for each author was computed. For 19 authors, no publications could be parsed; therefore, we computed the 5 nearest neighbors (using cosine distance) among *author embeddings* and filled in the missing year of publication with the average of the years of those neighbors. Afterwards, the years were scaled using a min-max normalization and the absolute difference is considered the corresponding distance function for this feature.
- *Bibliographic references* – people who wrote references about an author were extracted based on the list of references, thus generating the set of referees for each author. The distance function used for this feature was $1 - SIM(x, y)$, where $SIM(x, y)$ is the Jaccard index, namely the ratio between cardinal of the intersection and the cardinal of the union of the two respective sets. The distance respects the properties of a distance functions, as proven by Rajaraman and Ullman [15] .
- *Professions* – a set of professions was extracted for each author and Jaccard index was used to compare the overlap among pairs of authors. Some professions were grouped together because they were virtually the same (e.g., "gazetar" and "ziarist").

The distance between two authors was computed by adding each of the four previously described distances because the sum of two valid distances is also a valid distance function. After computing the distance matrix, the unsupervised *k* nearest neighbors' algorithm from scikit-learn[3] was applied.

IV. RESULTS

Some representative values of the Jaccard index applied on the authors' professions are presented in Table 1. Examples of well-known authors were used and, based on our knowledge on the area, the obtained results are relevant. Authors with identical professions have the similarity index 1.00 (as it can be seen "Dosoftei" with "Teodor Corbea"), whereas authors with only few identical professions have a lower value (e.g., "Ana Blandiana" and "Ion Creangă"). However, some associations may be misleading (e.g., "Dosoftei" and "Eugen Barbu" who had different interests), thus denoting the limitations of only considering professions as a metric for similarity. Correspondingly, the Jaccard index was applied on the bibliographic references.

Table 1. Examples of similarities based on professions for selected representative authors.

Author 1	Author 2	Similarity index
Ana Blandiana	Lucian Bureriu	1.00
Ana Blandiana	Teofil Bălaj	1.00
Ana Blandiana	Florica Bud	1.00
Barbu Delavrancea	Constantin Cheianu	1.00
Dosoftei	Teodor Corbea	1.00
Barbu Delavrancea	Ioan M. Bujoreanu	0.66
Barbu Delavrancea	Ion Marin Almăjan	0.66
Dosoftei	Ioan Barac	0.66
Ana Blandiana	Ion Creangă	0.50
Ana Blandiana	Tudor Arghezi	0.50
Dosoftei	Miron Costin	0.50
Dosoftei	George Coșbuc	0.50
Ana Blandiana	Eugen Barbu	0.40
Ana Blandiana	Ion Cristoiu	0.33
Barbu Delavrancea	Ion Creangă	0.33
Dosoftei	Eugen Barbu	0.33
Dosoftei	Antim Ivireanul	0.33
Ana Blandiana	Ion Luca Caragiale	0.25
Dosoftei	Gheorghe Asachi	0.20
Dosoftei	Vasile Alecsandri	0.20

[3] https://scikit-learn.org/stable/modules/neighbors.html

As previously mentioned, a visualization of the *authors embeddings* was realized with TensorBoard. Figure 3 depicts the most similar authors to "Miron Costin". The top distances or inverse cosine similarities are presented in Table 2. We manually explored the results through this visualization, and they seemed consistent with our knowledge of the authors. The tool is interactive while users can move in the 3-dimensional space and search for most similar authors of any given author.

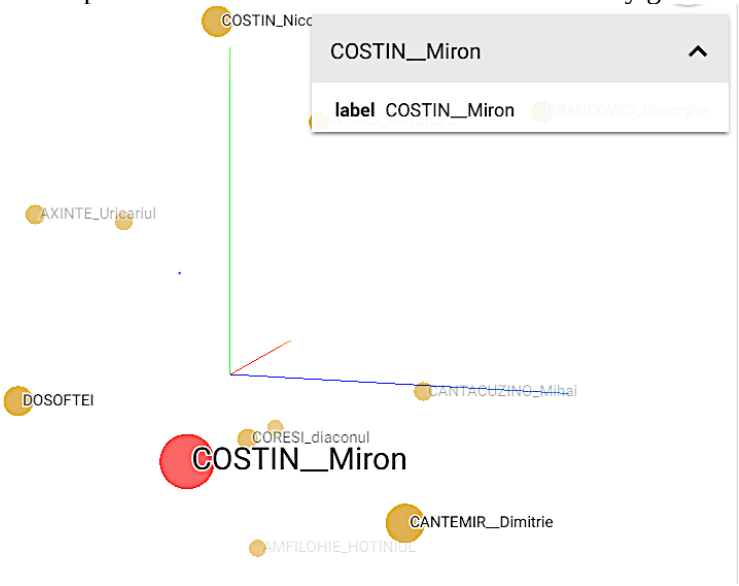


Figure 3. Miron Costin's most similar authors using the description feature.

Table 2. Miron Costin's most similar authors, using description feature.

Author name	Distance
Dimitrie Cantemir	.240
Dosoftei	.258
Nicolae Costin	.285
Uricariul Axinte	.355
Antim Ivireanul	.379
Mihai Cantacuzino	.411
Teodor Corbea	.412
Constantin Cantacuzino (stolnicul)	.421
Crimca Anastasie	.425

Table 3 displays the most similar authors to "Miron Costin" by considering all integrated features. Although the additional features have only a weight of $^{3}/_{20}$ compared to the author embeddings which have a weight of $^{17}/_{20}$, these metrics have a noticeable impact, as the most similar authors changed dramatically.

Table 3. Miron Costin's most similar authors, using all features.

Author name	Distance
Nicolae Costin	.125
Dimitrie Cantemir	.128
Gheorghe Brancovici	.151
Dosoftei	.160
Uricariul Axinte	.162
Teodor Corbea	.170
Alexandru Amiras	.184
Ioan Cantacuzino	.192
Mihai Cantacuzino	.197

Table 4 introduces the distances among some the most well-known authors having their surnames starting with [A-D] letters. The values in bold denote the closest associations, namely "Ion Barbu"– "Tudor Arghezi" (i.e., poets with less typical writings for their period) and "Ion Creangă"– "Tudor Arghezi" (i.e., writings addressing children), which make sense given their descriptions.

Table 4. Distances between the most renowned authors starting with [A-D] letters in their surname.

	Dimitrie Cantemir	**Vasile Alecsandri**	**Ion Creangă**	**Ion Luca Caragiale**	**Tudor Arghezi**	**George Bacovia**	**Ion Barbu**	**Lucian Blaga**	**Emil Cioran**
Dimitrie Cantemir	-	.393	.462	.467	.470	.620	.400	.380	.392
Vasile Alecsandri		-	.354	.206	.352	.370	.294	.376	.414
Ion Creangă			-	.330	**.195**	.355	.274	.394	.248
Ion Luca Caragiale				-	.311	.476	.318	.517	.394
Tudor Arghezi					-	.230	**.115**	.332	.350
George Bacovia						-	.211	.373	.459
Ion Barbu							-	.281	.299
Lucian Blaga								-	.273
Emil Cioran									-

V. CONCLUSIONS AND FUTURE WORK

Shifting Romanian literature to the digital era, coupled with the usage of advanced NLP technique tailored for Romanian language, are a trending research domain. The current study introduces new approaches for the online digitalization and transformation in terms of search and semantic recommendations of Romanian writings and authors based on the General Romanian Dictionary of Literature. Our prototype system processes only the letters from A to D from DGLR and will be extended, as subsequent volumes become available.

The presented work is applicable to the eLearning domain where future researchers may use our methodology to generate recommendations of lectures for specific authors, while performing exploratory searches using our semantic method. As follow-up actions, our aim is to integrate the current analyses into the *ReaderBench* framework [1; 2] that contains several advanced NLP processing techniques. The methods could be reused and enhanced by the community of developers to explore similar intertextual links between text descriptions or documents. The parsing mechanism for authors can be also improved to enhance the semantic recommendation of similar authors.

Acknowledgements

This work was supported by a grant of the Romanian National Authority for Scientific Research and Innovation, CNCS – UEFISCDI, project number PN-III 54PCCDI / 2018, INTELLIT – "Prezervarea și valorificarea patrimoniului literar românesc folosind soluții digitale inteligente pentru extragerea și sistematizarea de cunoștințe".

References

[1] Dascalu, M., Crossley, S., McNamara, D.S., Dessus, P., and Trausan-Matu, S., 2018. Please ReaderBench this Text: A Multi-Dimensional Textual Complexity Assessment Framework. In *Tutoring and Intelligent Tutoring Systems*, S. Craig Ed. Nova Science Publishers, Inc., Hauppauge, NY, USA, 251–271.

[2] Dascalu, M., Dessus, P., Bianco, M., Trausan-Matu, S., and Nardy, A., 2014. Mining texts, learner productions and strategies with ReaderBench. In *Educational Data Mining: Applications and Trends*, A. Peña-Ayala Ed. Springer, Cham, Switzerland, 345–377.

[3] Delbert Dueck, B.J.F., 2007. Non-metric affinity propagation for unsupervised image categorization. *ICCV*, 1–8.

[4] Edouard Grave, P.B., Prakhar Gupta,Armand Joulin,Tomas Mikolov, 2018. Learning Word Vectors for 157 Languages. *arXiv preprint arXiv:1802.06893*.

[5] Girija, S., 2016. Tensorflow: Large-scale machine learning on heterogeneous distributed systems.

[6] Gormley, C. and Tong, Z., 2015. *Elasticsearch: The Definitive Guide*. O'Reilly Media, Inc.

[7] Jessup, E.R. and Martin, J.H., 2001. Taking a new look at the Latent Semantic Analysis approach to information retrieval. In *Computational information retrieval*, M.W. Berry Ed. SIAM, Philadelphia, PA, 121–144.

[8] Jolliffe, I.T., 2002. Principal Component Analysis, Second Edition. *Springer*, 518.

[9] Landauer, T.K., Foltz, P.W., and Laham, D., 1998. An introduction to Latent Semantic Analysis. *Discourse Processes 25*, 2/3, 259–284.

[10] Lilleberg, J., Yun Zhu, and Yanqing Zhang, 2015. Support vector machines and word2vec for text classification with semantic features. *IEEE 14th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC)*.

[11] Manning, C.D. and Schütze, H., 1999. *Foundations of statistical Natural Language Processing*. MIT Press, Cambridge, MA.

[12] Michael D. Lee, B.P., Matthew Welsh, 2005. An Empirical Evaluation of Models of Text Document Similarity. *Proceedings of the Annual Meeting of the Cognitive Science Society 27*.

[13] Mikolov, T., Chen, K., Corrado, G., and Dean, J., Year. Efficient Estimation of Word Representation in Vector Space. In Proceedings of the Workshop at ICLR (Scottsdale, AZYear).

[14] R Kuć, M.R., 2014. ElasticSearch Server - Second Edition. *Packt Publishing Ltd.*

[15] Rajaraman, A. and Ullman, J.D., 2011. *Mining of massive datasets*. Cambridge University Press.

[16] Richard A. Groeneveld, G.M., 1984. Measuring Skewness and Kurtosis. *Journal of the Royal Statistical Society. Series D (The Statistician) 33*, 4, 391-399.

[17] Wu, X., Kumar, V., Quinlan, J.R., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G.J., Ng, A., Liu, B., Yu, P.S., Zhou, Z.-H., Steinbach, M., Hand, D.J., and Steinberg, D., 2008. Top 10 algorithms in data mining. *Knowledge and Information Systems 14*, 1, 1–37.