NATIONAL RESEARCH UNIVERSITY
HIGHER SCHOOL OF ECONOMICS

Anna Scherbakova

COMPARATIVE STUDY OF DATA CLUSTERING ALGORITHMS AND ANALYSIS OF THE KEYWORDS EXTRACTION EFFICIENCY: LEARNER CORPUS CASE

BASIC RESEARCH PROGRAM

WORKING PAPERS

SERIES: LINGUISTICS
WP BRP 97/LNG/2020

SERIES: *LINGUISTICS*

Anna Scherbakova[1]

COMPARATIVE STUDY OF DATA CLUSTERING ALGORITHMS AND ANALYSIS OF THE KEYWORDS EXTRACTION EFFICIENCY: LEARNER CORPUS CASE[2]

The paper focuses on the task of clustering essays produced by ESL (English as a Second Language) learners. The data was taken from a learner corpus REALEC. The division of texts by certain characteristics can be useful to speed up the analysis of a single corpus or access to the necessary sections of a large number of documents. The study discusses not only some existing approaches to clustering text data, as well as the possibility of clustering texts produced by ESL learners, but also ways to extract keywords in order to determine the topic of the essays in each group.

JEL Classification: Z.

Keywords: learner corpus, text documents clustering, document embedding, keywords extraction, metadata enrichment.

[1] National Research University Higher School of Economics. School of Linguistics: Bachelor Student. E-mail: aniezka.sherbakova@gmail.com

[2] The research was carried out within the HSE project 2019 – Automated detection of writing inaccuracies for students of English in Russia (ADWISER).

1. Introduction

Corpora consisting of texts produced by non-native speakers present an invaluable source of linguistic data for researchers. Various studies have been conducted on the basis of learner corpora: automatic language scoring (Vajjala, 2018), identifying text complexity (Kurdi, 2020), automatic text classification within different, proper word choice task (Makarenkov et al., 2019), semantic collocation correction (Dahlmeier and Ng, 2011), lexical substitution (McCarthy and Navigli, 2009), paraphrase generation (Madnani and Dorr, 2010), grammatical error correction (Ng et al., 2014), sentence completion (Zweig and Burges, 2011), to name just a few. Also, there are many papers devoted to obtaining document embeddings ((Salton and Buckley, 1988), (Whissell and Clarke, 2011), (Mikolov and Le, 2014)), clustering algorithms ((Steinhaus, 1956), (Ester at al., 1996), (Merris, 1994)), and various techniques for keywords extraction ((Mihalcea and Tarau, 2004), (Rose et al., 2010), (Sterckx et al., 2016)).

The task of cluster analysis is an important problem in NLP and other areas of machine learning, and the need for it can arise at all levels of document processing – from combining words into groups to clustering document collections. In this study, the concept of clustering can be considered as receiving labels for each document from the corpus so documents within a cluster have high intra-similarity and low inter-similarity to other clusters (Jensi and Wiselin, 2017).

In most cases, clustering algorithms deal with vector representations of objects that should be combined into several groups according to some criteria. Thus, these are cases when there is a need to correlate each document from the collection with a vector of a certain dimension, which will reflect the main features of the document's contents and have certain properties. One of the expectations is that in the area of similar documents the distance between the corresponding vectors is less than the distance to the vector representations of texts that are very different from the former because of differences in context. On the one hand, the choice of an algorithm for obtaining embeddings have more effects on the final result of clustering, because vectors of very small dimensions may reflect semantic contents of texts only weakly, but, on the other hand, as the size of the dimension increases, so does the computational complexity, and, as a result, the processing time of the selected clustering algorithm rapidly rises.

In this paper we use the corpus called REALEC – a "systematic computerized collection of texts that are written productions of English language learners" (Vinogradova, 2016: 830). We focus on the experimental comparison of various methods for obtaining embeddings and clustering algorithms based on the text data from this learner corpus. The objective of our experiment is to determine the topic for essays from each cluster by extracting keywords using different methods and to use the results in order to enrich metadata of REALEC.

2. Dataset overview

All experiments are conducted on the basis of a publicly available Russian Error-Annotated Learner English Corpus (REALEC)[3], which consists of texts written in English by university students of the Higher School of Economics who study English as a foreign language.

Various errors are annotated in the corpus manually, while POS tags are assigned automatically; moreover, each word is associated with a lemma. However, the source texts without annotation can also be used for research. Metadata present in REALEC includes gender, year of study, etc., but there is no indication of the task, and the tasks themselves are absent. In this paper, we are focusing on the REALEC subcorpus, namely, texts written by students of the Higher School of Economics as part of the Independent English Language Examination[4]. Essays were written as answers to two types of tasks: a description of the graphical material in the task and an opinion essay on a specific topic. The study was carried out separately for the work of each year, and clustering was analysed for the texts of each task separately. It is worth noticing that due to checking the quality of the algorithms, the topic of 600 essays was manually labeled in order to choose an algorithm that will show the best result. The number of labeled and unlabeled texts for each year is presented in Table 1 by the genre. The topic breakdown was also done in order to try to match the exact topic to each set of extracted keywords for each cluster.

[3] https://realec.org/index.xhtml#/exam/

[4] https://www.hse.ru/studyspravka/indexam

Table 1. The number of opinion essays and descriptions of graphical material by year

	Year	Opinion essays	Graph descriptions
Labeled data	2017	299	301
Unlabeled data	2019	195	210
	2018	299	267
	2017	713	721
	2016	661	668
	2015	29	31
	2014	824	828

After loading the data, the subcorpus was preprocessed: after tokenization and lemmatization, we put tokens to lowercase and removed all stop words (provided by NLTK) and punctuation.

3. Algorithms description

In this paper, we observe some methods for obtaining embeddings, which showed the best result while clustering texts on the material of different corpora (Parhomenko et al., 2017). We took methods such as TF-IDF (Salton and Buckley, 1988), BM25 (Whissell and Clarke, 2011), doc2vec (PV-DM and PV-DBOW) (Le and Mikolov, 2014).

TF-IDF is the assumption that the significance of an n-gram is directly proportional to the frequency of its occurrence in a document and is inversely proportional to the proportion of documents in the set in which this n-gram occurs (Parhomenko et al., 2017). This means that the largest weight is obtained by n-gram, which is often found in one document, but not found in the rest part of it: in other words, this n-gram is an attribute that distinguishes this document from others. Such a vector representation is calculated using the following formula:

$$TF \cdot IDF(t_i, d_j, D) = tf(t_i, d_j) \cdot log \frac{|D|}{|(t_i \in d_j)|} \quad (1)$$

where $tf(t_i, d_j)$ – an n-gram frequency ti in the document d_j, D – a set of documents, $|t_i \in d_j|$ – all such documents in a set that contain n-gram t_i

BM25 is a method of weighing the meaning of words. It limits the significance of the frequency of the n-gram, and it is not only normalized by its size, but also limited from above, which avoids assigning the word too much weight (Parhomenko et al., 2017). The value of features for n-gram is calculated by the formula:

$$idf(t_i) \cdot \frac{tf(t_i, d_j) \cdot (k_1 + 1)}{k_1 \cdot (1 - b + b \cdot \frac{|d_j|}{|d_{avg}|}) + tf(t_i, d_j)} \quad (2)$$

where $|d_j|$ – length of the document, $|d_{avg}|$ – average length of documents in a set, k_1 and b – free parameters (in this paper are equal to 1.6 and 0.75, respectively)

The group of algorithms that is used to obtain vector representations of words, word2vec, was presented in (Mikolov et al., 2013). This model projects words into the space of vectors, where vectors are matched to words of similar meaning, the distance between them is the smaller, the closer these words are in meaning. Such an effect is achieved through the use of a neural network, which is trained to predict by the word vector its context (Parhomenko et al., 2017). It is worth saying that this method differs from word2vec: in the algorithms the order of words in the context is not important, while in word2vec, on the contrary, it plays a key role. Two years after the presentation of the word2vec model in 2014, (Le and Mikolov, 2014) described two methods of vectorizing documents under the general name Paragraph Vectors (doc2vec). Doc2vec is an extension of capabilities of word2vec model: word2vec learns to project words into a hidden d-dimensional space, while doc2vec attempts to learn how to project a document into a hidden d-dimensional space. As mentioned above, Paragraph Vectors approach consists of two methods for obtaining a vector representation of documents: PV-DM (Distributed Memory) and PV-DBOW (Distributed Bag of Words). The main difference between these approaches is that PV-DM takes into account the word order and their context in the document, and PV-DBOW tries to predict the words that contain the document by a vector of text.

Cluster analysis deals with the task when a certain set of objects needs to be divided into several groups according to some criteria. There are many algorithms of clustering:

partition clustering (k-means), hierarchical clustering, density-based clustering methods (DBSCAN), etc. (Parkhomenko 2017).

In the classical interpretation, most clustering methods use vectors of the same dimension which must be combined into groups, and their number is either specified in advance (k-means) or determined during the operation of the algorithm (DBSCAN). Each cluster has to consist of similar objects, that is, within a separately selected group, the distance between two vectors must be less than the distance from them to the vector of any other cluster. The solution to this problem has to overcome the following difficulties: there are many different criteria for assessing the quality of clustering; the exact number of clusters is usually not known in advance; the final result strongly depends on the distance calculation metric.

The main goals of applying cluster analysis are simplifying further data processing (dividing a set into groups of similar objects and working with them separately), reducing the amount of stored data (leaving one object of each cluster), and isolating atypical objects from the general set (determination of outliers).

This allows to define which clustering methods to apply to solve the problem of clustering vectorized text data: k-means, hierarchical clustering with different metrics for calculating distances, density-based spatial clustering of applications with noise (DBSCAN) and spectral clustering with different parameters.

K-means (Steinhaus, 1956) is a partitioning clustering algorithm that initially randomly selects the center of mass for each of the clusters and assigns to each object the label of the cluster whose distance to the center of mass from the document is less. Then an iterative process takes place: at each step the algorithm recalculates the centers of mass of the clusters and changes the document labels in accordance with the new partition. The process stops when a sufficiently small change in the centers or when the maximum number of iterations is reached (Parkhomenko 2017).

Hierarchical clustering algorithms iteratively build a system of nested partitions – a set of samples of disjoint classes. The result of such an algorithm is usually represented in the form of a taxonomic tree – a dendrogram. Dendrograms can be built top-down (agglomerative clustering) and bottom-up (divisive clustering). In this study, the agglomerative clustering approach was used. Initially, each object was allocated into a separate cluster, then at each

iteration the closest clusters were combined into one. The Ward method was used to calculate the distance between two clusters.

Density-based spatial clustering of applications with noise (DBSCAN) (Ester at al., 1996) refers to density clustering algorithms; they allocate into clusters those spaces in which there is a high density of objects. If a set of points is specified in space, then the method groups closely spaced points among themselves, and marks lone points in areas with low density as outliers.

Spectral clustering (Merris, 1994) is one of the effective clustering algorithms that can be used to solve nonlinear separable problems. This algorithm groups points using the eigenvalues of a matrix derived from the data.

To assess the quality of clustering, two types of measures are distinguished: external measures that use additional (external) information about the real distribution of objects (for example, the knowledge of real classes of objects – a marked subset of the REALEC corpus essay), and internal measures, which are calculated using information only about the obtained partition. Also in the study (Parkhomenko 2017), it is noted that the choice of an external measure for assessing the quality of partitioning into groups (for the calculation, the previously known labels of the cluster number of each object are used) weakly affects the final rating of the vectorization and clustering algorithms, therefore, two external measures can be used in the work - one that depends on the relative position of objects in groups and the one that does not depend on the numbering of clusters – Adjusted Mutual Information (AMI) and Normalized Mutual Information (NMI) (Parkhomenko 2017). As internal measures for evaluating the effectiveness of clustering when working with unlabeled data, in this work we used only those that had shown the best results in experimental comparison in the work (Arbelaitz et al. 2013), namely: Silhouette, Davies-Bouldin (in contrast to other measures, the lower value of this measure corresponds to the better quality of clustering) and Calinski-Harabaz.

Keywords describe the subject of the document in the best way., as they effectively summarize the content of the document (Škrlj et al., 2019). In this paper, various algorithms for extracting keywords were considered. We used averaging of TF-IDF vectors (Sterckx et al., 2016) in each of the clusters and the selection of the corresponding words with the maximum weight, which made it possible to look at the words that had the greatest value in each cluster.

TextRank algorithm (Mihalcea and Tarau, 2004) projects text into a graph, where both individual words and the whole sentences can be represented as nodes. For the former, the algorithm will return keywords, and for the latter, it is suitable for a short description of texts. In this method, some nodes "recommend" others, and the strength of the recommendation is calculated recursively based on the ratings of the edges.

RAKE algorithm (Rose et al., 2010) tries to determine key phrases in the text by analyzing the frequency of occurrence of a word and its compatibility with other words in the text.

4. Experiments

In this study, the task of clustering in accordance with the specifics of the data was implemented in three stages: preprocessing of documents, obtaining text embeddings (vectorization) and implementing clustering algorithms based on the obtained embeddings. Each of these steps can be done in various ways, the choice of which depends on the result of clustering.

In this paper, we compare various algorithms for vectorizing learner's essays and consider the results of the clustering algorithms based on the obtained vector representations.

To obtain clustering, firstly, we determined the best algorithm for obtaining document embeddings. Therefore, several methods of vector representation were considered on the labeled sample, on which the k-means algorithm was used with the number of clusters equal to the real one and the maximum result for several iterations (Table 2).

Further, the best algorithm for obtaining embeddings was applied on unlabeled data, where, using the k-means algorithm and the values of internal metrics, the optimal number of clusters was determined. Internal metrics were considered in the range from 2 to 12 clusters and averaged over 5 iterations. Then, the result of the k-means algorithm with a given optimal number of clusters was used to extract keywords for each group of essays.

In order to find the optimal pipeline for clustering essays by topic, several vectorization algorithms were considered, and their comparative analysis was carried out (we used the k-means algorithm with the same parameters and, based on the result of its work, we applied the external measures AMI and NMI on the test set). The results can be seen in Table 2.

Table 2. Values of the best external measures for assessing the quality of clustering k-means, comparison on 2017 essays labeled material.

	External evaluation	TF-IDF	BM25	doc2vec (PV-DM)	doc2vec (PV-DBOW)
2017 opinion essays	AMI	**0.973**	0.922	0.528	0.487
	NMI	**0.974**	0.927	0.554	0.517
2017 graph description	AMI	**0.960**	0.928	0.894	0.882
	NMI	**0.962**	0.932	0.899	0.887

The results of our experiment showed that TF-IDF handles best with the task of document vectorization, BM25 shows slightly worse results, and both algorithms of the doc2vec method cannot separate opinion essays by topics at all, but they cope well with the descriptions of graphs. This is due to the fact that there is not enough data for training and the algorithm does not have time to train on the corpus; moreover, the essays are written according to a certain structure with a large number of template phrases (for example, phrase '*on the other hand*'), and in this situation, the method of averaging document vectors that uses doc2vec cannot highlight the topic and divide texts' vectors. It is worth noting that the selection of hyperparameters for the doc2vec algorithms was carried out using the optuna framework[5].

The best results were shown by the TF-IDF algorithm. It is explained by the fact that TF-IDF highlights the words in the document that distinguish it from the collection; in other words, it "ignores" the template words, assigning them small weights in the document's vectors, since they appear in various essays.

Thus, on unlabeled data from REALEC, we chose to use TF-IDF as an algorithm for obtaining vector representations of essays and select the number of clusters depending on the values of internal measures of assessing the quality of clustering. We considered in detail all the stages using the example of opinion essays of the year 2018 (applying the methods that showed the best result on the test set of 2017). The averaged values of internal measures are shown in Figure 1:

[5] https://optuna.readthedocs.io/en/latest/index.html

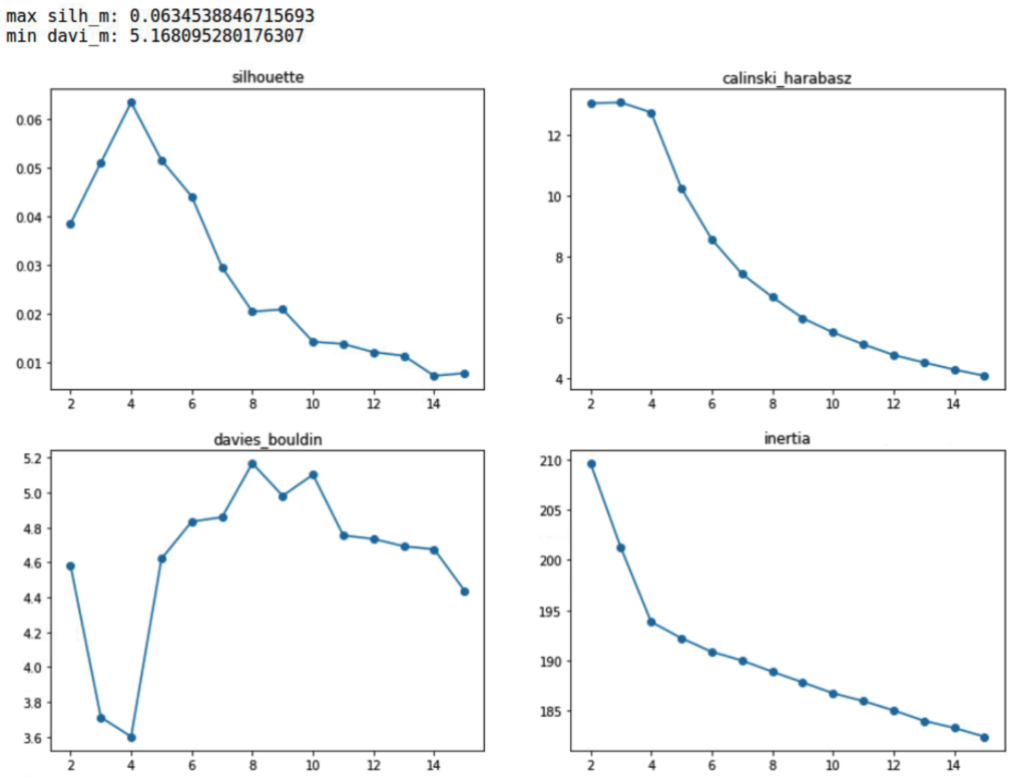

Figure 1. The averaged values of internal measures on the material of opinion essays of 2018.

The graphs show that the optimal number of clusters for the k-means algorithm is four. Then we consider the result of the work of dimension reduction algorithms in order to look at the result of k-means with the number of clusters equal to four (Figure 2):

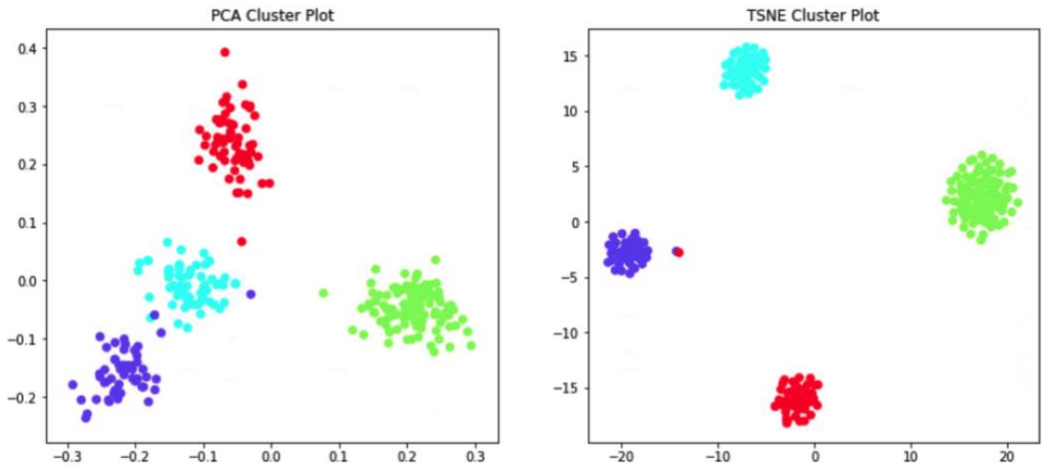

Figure 2. The result of the k-means algorithm with a decrease in the dimension of PSA and t-SNE on the material of opinion essays 2018.

It can be noted that PCA showed good division of essays by topics, and the subsequent use of t-SNE made it possible to group clusters more tightly and more clearly identify the outliers. In this case, outliers are incorrectly annotated essays: the description of graphs X_1.txt falls into the subcorpus of opinion essays, which are grouped according to the template X_2.txt.

Thus, the optimal number of clusters was chosen on the basis of internal measures of assessing the quality of clustering. Since the division into clusters is very obvious, the result of the operation of other clustering algorithms will be close to the partition by the k-means method, so we immediately consider their predictions in the form of t-SNE (Figure 3):

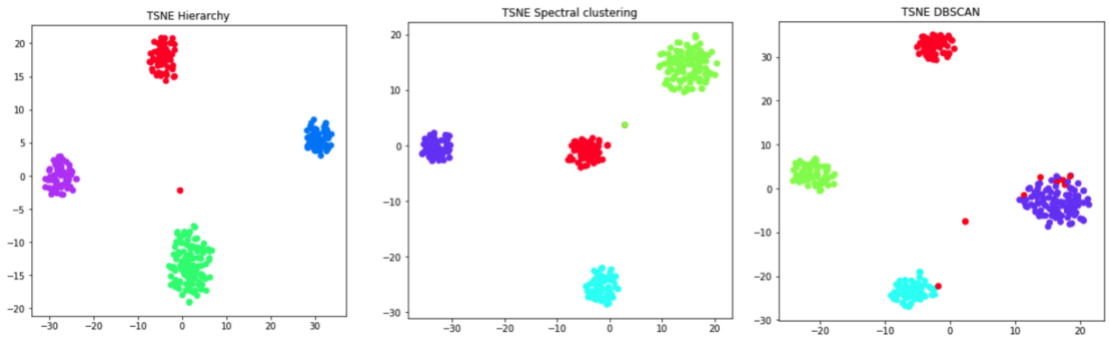

Figure 3. The result of clustering algorithms with a decrease in the dimension of t-SNE on the material of opinion essays 2018 (from left to right: hierarchical, spectral clustering and DBSCAN).

Now we can extract keywords for each cluster by summing all the TF-IDF essays vectors in each cluster and taking the words with maximum weights:

Cluster: 1

country, government, help, people, problem, world, citizen, helping, global, need, one, support, state, issue, nation, focus, war, solve, international, provide, example, others, live, money, life, situation, lot, many, political, point

Cluster: 2

money, time, free, people, work, earn, life, spend, le, prefer, lot, working, person, much, family, thing, leisure, job, friend, earning, hard, want, hand, health, without, one, would, happy, salary, enough

Cluster: 3

building, city, architect, beautiful, people, art, purpose, architecture, look, house, important, work, make, beauty, serve, place, like, think, would, one, also, create, live, construction, built, opinion, many, appearance, time, lot

Cluster: 4

child, family, parent, influence, outside, life, people, friend, home, school, development, factor, role, kid, play, person, different, ha, important, powerful, first, character, teacher, view, one, society, part, member, age, also

We also looked at the results of keywords extraction using other algorithms as an example. We selected the most popular words for each cluster by extracting the keywords of each essay separately and chose the 20 words most often encountered. We used the RAKE algorithm for this, which for extracting keywords takes into account the frequencies of individual words and their joint occurrence, and we obtained the following results:

Cluster: 1
countries, people, help, country, world, government, governments, problems, citizens, lot, live, order, example, money, focus, believe, support, planet, view, need

Cluster: 2
money, people, free time, time, work, life, lot, spend, person, earn, example, family, however, make, prefer, friends, order, important, want, able

Cluster: 3
buildings, building, people, art, architects, important, work, purpose, city, live, beauty, architecture, make, lot, think, time, opinion, however, example, beautiful

Cluster: 4
child, family, parents, life, children, people, friends, home, development, influence, lot, school, outside, person, view, way, important, however, example, world

As we can see, the results contain many words which the previous method based on TF-IDF also extracted. Next, we selected text from each cluster that was as close as possible to its center, assuming that it describes the topic as fully as possible, and we looked at the result of the algorithms for selecting keywords based on this essay to compare with the keywords selected for the 11 entire cluster. Here are the results of RAKE algorithm:

Cluster: 1
solve problems, better, country, world, countries, life, help, people, support, citizens, disagree, consider, live, lot, much

Cluster: 2
less free time, free time, realy like, less money, people believe, people think, earn money, time, earn, people, money, life, lot, however, important, spend, buy, work

Cluster: 3
look beautiful, really important, make buildings, also, buildings, make, important, belive, architects, people, serve, purpose, works, art, agree, opinion, boring, artists, live, think

Cluster: 4
important role, world, child, family, life, children, upbringing, lot, exactly, parents, behavior, mothers, opinion, things, way, communication, manners

As we can see, now among the keywords there are phrases and words with erroneous spelling (for example, '*belive*'), which are not present when "averaging" keywords in the cluster. It can be concluded that when extracting the most common keywords from the entire

cluster, words and phrases with errors are less common than when extracting keywords from the same essay in this cluster. This is due to the fact that, on average, students do not make this kind of mistakes in essays, for example, when writing the word '*believe*'. The TextRank algorithm describes the joint occurrence of words using a graph whose edges show the importance of the corresponding word in the text, then starts a random walk on the graph and determines the most often 'visited' nodes:

Cluster: 1
country, people, help, better, live, living, lived, developed, develop, problem, world disagree, lot

Cluster: 2
time, lot people think money, earn, earned, work, got, life

Cluster: 3
important building look beautiful, think, interesting, architect, colourful, colour, people belive, purpose, agree, boring

Cluster: 4
child, people, family social, life exactly parent, world, lot, want, influence, behavior manner, girl, mother, thing, outside, play, role, crucial, way, start communicating, started communication, practise, count

The obtained keywords for each group allow to define a common topic for all essays in the cluster. For example, we can conclude that essays in cluster 3 are united by the common theme "Architecture," and in cluster 1, by "Country and citizens". Thus, a non-laborious method of high-quality clustering of essays by topic was obtained.

5. Conclusion

In this paper, we compared various algorithms for obtaining text embeddings on the REALEC dataset. The comparison allowed us to determine the best combination of vectorization and clustering algorithms – TF-IDF and k-means, since TF-IDF showed better results in comparison with other methods for obtaining vector representations of texts, and k-means allowed us to accurately distinguish between classes and find cluster centers as well as outliers.

Thus, the immediate result of the work is the division of essays into clusters and their corresponding sets of keywords, resulting in an explicit topic of each collection of documents united by one cluster.

The empirical result of this study is the list of topics for all essays from the REALEC corpus. Also, the results of this work can be used for creating a convenient basis for searching essays by keywords - the proposed way of clustering essays by topics (keywords) will allow quick search of essays on the topics.

As a continuation of this study, we can see the attempt to generate a summarising description in order to reconstruct the exact topic of each group on the basis of extracted keywords. The study should be continued in the direction of identifying the correlation between clustering constituents and lexical diversity values, that is, towards a greater focus on the research task of automated evaluation of lexical features of the text.

References

Arbelaitz, O., Gurrutxaga, I., Muguerza, J., PérezIñigo, J., Perona, I. (2013). An extensive comparative study of cluster validity indices. *Pattern Recognition,* vol. 46, no. 1. pp. 243-256.

Dahlmeier, D., Ng, H.T. (2011). Correcting semantic collocation errors with L1-induced paraphrases. In *Proceedings of the 2011 conference on empirical methods in natural language processing*, pp. 107-117.

Ester, M., Kriegel, H.-P., Sander, J., Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. *AAAI Press*, pp. 226–231.

Firoozeh, N., Nazarenko, A., Alizon, F., Daille, B. (2019). Keyword extraction: Issues and methods. *Natural Language Engineering,* pp. 1–33.

Jensi, R., Wiselin Jiji, G. (2013). A Survey on optimization approaches to text document clustering. *International Journal on Computational Sciences & Applications (IJCSA)*, vol. 3, no. 6, pp. 31-44.

Kurdi, M. Zakaria. (2020). Text Complexity Classification Based on Linguistic Information: Application to Intelligent Tutoring of ESL. *Journal of Data Mining & Digital Humanities*, no.1, pp. 1-40.

Le, Q., Mikolov, T. (2014). Distributed Representations of Sentences and Documents. *PMLR,* vol. 32, no. 2, pp. 1188-1196.

Madnani, N., Dorr, B.J. (2010). Generating phrasal and sentential paraphrases: A survey of data-driven methods. *Computational Linguistics*, vol. 36, no. 3, pp.341-387.

Makarenkov, V., Rokach, L., Shapira. B. (2019). Choosing the right word: Using bidirectional LSTM tagger for writing support systems. *Engineering Applications of Artificial Intelligence*, vol. 84, pp. 1-10.

McCarthy, D., Navigli, R. (2009). The English lexical substitution task. *Language resources and evaluation*, vol. 43, no. 2, pp.139-159.

Merris, R. (1994). Laplacian matrices of graphs: a survey. *Linear algebra and its applications*, pp. 197, 143-176.

Mihalcea, R., Tarau, P. (2004). TextRank: Bringing order into texts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Barcelona, Spain, pp. 404–411.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, pp. 3111-3119.

Ng, H.T., Wu, S.M., Briscoe, T., Hadiwinoto, C., Susanto, R.H., Bryant, C. (2014). The CoNLL-2014 shared task on grammatical error correction. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task,* pp. 1-14.

Parhomenko, P.A., Grigorev, A.A., Astrakhantsev, N.A. (2017). Obzor i eksperimentalnoe sravnenie metodov klasterizatsii tekstov [A survey and an experimental comparison of methods for text clustering: application to scientific articles]. *Trudy ISP RAN/Proc. ISP RAS*, vol. 29, no. 2, pp.161-200.

Rose S., Engel D., Cramer N., Cowley W. **(**2010). Automatic keyword extraction from individual documents. In Berry M.W. and Kogan J. (eds) *Text Mining: Applications and Theory*, pp. 1–20.

Salton, G., Buckley, C. (1988). Term-Weighting Approaches in Automatic Text Retrieval. *Inf. Process. Manag.,* vol. 24, pp. 513-523.

Škrlj, B., Repar, A., Pollak, S. (2019). RaKUn: Rank-based Keyword extraction via Unsupervised learning and Meta vertex aggregation. In *International Conference on Statistical Language and Speech Processing*, pp. 311-323.

Steinhaus, H. (1956). Sur la division des corps matériels en parties. *Bulletin de l'Académie Polonaise des Sciences*, Classe III, vol. IV, no. 12, pp. 801–804.

Sterckx, L., Caragea, C., Demeester, T., Develder, C. (2016). Supervised keyphrase extraction as positive unlabeled learning. *In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Austin, Texas, pp. 1924–1929.

Vinh, N. X., Epps, J., Bailey, J. (2010). Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *The Journal of Machine Learning Research*, vol. 1, pp. 2837– 2854.

Vajjala, S. (2018). Automated Assessment of Non-Native Learner Essays: Investigating the Role of Linguistic Features. *Int J Artif Intell Educ,* vol. 28, pp. 79–105.

Vinogradova, O.I. (2016). The Role And Applications Of Expert Error Annotation In A Corpus Of English Learner Texts. *Proceedings of the Annual International Conference "Dialogue",* vol. 15. Moscow, Russia, July 1-4, pp. 830- 840.

Whissell, J.S., Clarke, C.L.A. (2011). Improving document clustering using Okapi BM25 feature weighting. *Inf Retrieval*, vol. 14, pp. 466–487.

Zweig, G., Burges, C.J. (2011). The microsoft research sentence completion challenge. Microsoft Research, Redmond, WA, USA, Tech. Rep. MSR-TR-2011-129.

Contact details and disclaimer:

Anna Scherbakova
National Research University Higher School of Economics (Moscow, Russia).
E-mail: aniezka.sherbakova@gmail.com

Any opinions or claims contained in this Working Paper do not necessarily reflect the views of HSE.