Text Document Clustering: Wordnet vs. TF-IDF vs. Word Embeddings

Michał Marcińczuk♣, Mateusz Gniewkowski♣, Tomasz Walkowiak♣, Marcin Będkowski♢

♣ Wrocław University of Science and Technology, Poland
{michal.marcinczuk,mateusz.gniewkowski,tomasz.walkowiak}@pwr.edu.pl

♢ University of Warsaw, Poland
mbedkowski@uw.edu.pl

Abstract

In the paper, we deal with the problem of unsupervised text document clustering for the Polish language. Our goal is to compare the modern approaches based on language modeling (doc2vec and BERT) with the classical ones, i.e., TF-IDF and wordnet-based. The experiments are conducted on three datasets containing qualification descriptions. The experiments' results showed that wordnet-based similarity measures could compete and even outperform modern embedding-based approaches.

1 Introduction

The aim of the paper is to evaluate different semantic distance calculation methods using clusterization by the Agglomerative Clustering method regarding qualifications collected in the Integrated Qualifications Register (IQR). It is a Polish public register supporting the Integrated Qualifications System (IQS) and regulated by the Act of 22 December 2015 on the Integrated Qualifications System. The IQR enables broad access to qualifications functioning in the national education system and enhances its transparency, as well as encourages the development of lifelong learning (IBE, 2020, p. 50).

As a repository of information about qualifications, the IQR does not meet the definition of Big Data — at least not yet — but still, it can benefit from the use of natural language processing methods allowing the calculation of similarity of documents and their clustering. The project entitled "Operating and Developing the Integrated Qualifications Register" financed by the European Social Fund aims at developing several applications supporting citizens in their career decisions and policy-makers in their strategic choices.

The main problem was how to compare and find similar qualifications from different sources, e.g., higher education (HE) diplomas and vocational education and training (VET) certificates, and group them in meaningful and interpretable clusters.

At the beginning of our work, we aimed at exploring content-based semantic similarity of qualifications, so we relied mostly on unsupervised clustering methods. Eventually, we covered both unsupervised and supervised techniques. We evaluated traditional methods and modern ones, as we wanted to test several approaches regarding their efficiency, interpretability, and feasibility. Here, we will present part of our work dealing with unsupervised methods.

2 Datasets

The dataset covers several thousand documents containing descriptions of qualifications (out of a total number of about 10000 qualifications included in the IQS and IQR). These descriptions mainly consist of so-called learning outcomes statements (LOs), which characterize the knowledge, skills, and attitudes required to obtain a given qualification. LOs can be broken down into three main components: an action verb, a skill object, and a context of the performance demonstration, e.g., "(Person) creates documents using word processing software".

Learning-outcomes-based qualifications framework is intended to "provide a common language allowing different stakeholders in education and training, as well as the labor market and society at large, to clarify skills needs and to respond to these in a relevant way" (Cedefop, 2017, p. 26). It is assumed that LOs allow for comparison of qualifications across the sector, institutional, and national borders, which was why we started with a content-based semantic similarity of qualifications and clustering techniques.

Qualification name	Category	Label
Web application and database development and administration	market qualification	IT
Computer graphics design	market qualification	IT
IT technician	VET qualification	IT
Programming, development and administration of websites and databases	VET qualification	IT
Computer science	HE diploma	IT
Game and virtual space design	HE diploma	IT
Dental technician	VET qualification	Medicine
Veterinary technician	VET qualification	Medicine
Psychooncologist	market qualification	Medicine
Supplying stores with mass-produced medical products	market qualification	Medicine
Medical rescue	HE diploma	Medicine
Medicine	HE diploma	Medicine

Table 1: Sample clusters of qualifications

Dataset name	Documents	Tokens/doc	Labels
PPKZ	633	539–17810	13
Market	362	48–888	18
Higher education	2029	29–11355	21
ALL	3024	29–17810	36

Table 2: Datasets used in the experiments

The IQR is a source of information about qualifications functioning in the IQS. However, it does not contain descriptions and learning outcomes for some qualifications, especially HE diplomas. This information is available on university and government websites, usually in PDF files. To obtain the data, we used web-scraping and OCR techniques. As a result, the IQR data has been complemented by about 2000 descriptions.

In the experiment, we used four manually labeled datasets (see Table 2). The labels denote the sectors to which the qualifications belong (see Table 1).

3 Text Similarity

3.1 Wordnet

The literature describes several metrics used to calculate the semantic similarity between two words based on their position in the wordnet structure. Here are the more known metrics:

- shortest path — the similarity is computed based on the shortest path between synsets. The similarity is in the range of 0 to 1, where 1 represents words identity;
- Leacock-Chodorow (Leacock and Chodorow, 1998) — the similarity is computed based on the shortest path between synsets and synsets' depth in the wordnet structure;
- Lin (Lin, 1998) — the similarity is computed based on *Least Common Subsumer (LCS) and* Information Content (IC). LCS is the most specific ancestor node, and IC is a measure of synset specificity (higher values are associated with more specific concepts, and lower values are more general). The similarity is in the range of 0 to 1, where 1 represents words identity;
- Wu-Palmer (Wu and Palmer, 1994) — it is a specific case of Lin measure, where the information content is the same for each synset;
- Jiang-Conrath (Jiang and Conrath, 1997), Resnik (Resnik, 1995) — other metrics which also utilize *Least Common Subsumer and* Information Content.

Budanitsky and Hirst (2006) showed that Lin metric obtained the highest correlation with human intuition. Because Polish wordnet does not contain information content, thus we could not use this metric directly. We decided to utilize the Wu-Palmer metric as it is a specific case of Lin, which does not require information content. We-Palmer metric is calculated according to Formula 1. In the formula, $depth$ is the length of the shortest path from the synset to the wordnet root.

The similarity between documents is computed according to Formula 2 (Mihalcea et al., 2006). In the formula, T_1, T_2 represent sets of synsets for the documents, and $maxSim(w, T_2)$ is the highest similarity value for a synset $w \in T_1$ and any synset from T_2. Since the clustering algorithm requires a distance matrix, we converted the similarity measure using Formula 3 (the similarity from Formula 2 is within the range 0 to 1)

In the experiments, we used Słowosieć 3.2 (Maziarz et al., 2016) (a wordnet for Polish) and

$$wu - palmer(s_1, s_2) = 2 * \frac{depth(LCS(s_1, s_2))}{depth(s_1) + depth(s_2)} \quad (1)$$

$$sim(T_1, T_2) = \frac{1}{2} * \left(\frac{\sum\limits_{w \in T_1} (maxSim(w, T_2) * idf(w))}{\sum\limits_{w \in T_1} idf(w)} + \frac{\sum\limits_{w \in T_2} (maxSim(w, T_1) * idf(w))}{\sum\limits_{w \in T_2} idf(w)} \right) \quad (2)$$

$$distance = 1 - sim \quad (3)$$

WoSeDon (Janz et al., 2018) — a tool for word sense disambiguation. To calculate the document similarity we used the *wnsim* tool[1].

3.2 TF-IDF

The most classical method for building a vector representation of texts is a bag of words. This approach's key assumption is that the text can be expressed using an unordered set of frequencies of words (terms) in text. The number of selected features (words) can be often reduced by transforming the words into their generic form (stemming, lemmatization). The text frequency (TF) representation is very often modified by the Inverted Document Frequency (Salton and Buckley, 1988) (IDF), giving a TF-IDF representation of texts. In performed experiments, we have used a tagger for Polish to lemmatize the text and TF-IDF representation of lemma 1-, 2-, and 3-grams.

3.3 Language Models

Language modeling is a modern approach to text analysis based on the assumption that individual words or even whole sentences can be represented by high-dimensional feature vectors. It is based on the hypothesis that relationships (distances) between vector representations of words or sentences can be related to semantic similarities of words/sentences. The models are built on large text corpora by observing the co-occurrence of words in similar contexts.

3.3.1 doc2vec

One of the most popular techniques of language modeling, *word2vec*, is based on neural networks (Le and Mikolov, 2014). In the so-called skip-gram approach, the aim is to predict context words from a given word. In the classical word2vec (Le and Mikolov, 2014) technique, each word (form from the text) is represented by a distinct vector, which might be a problem for a language with large vocabularies and rich inflection like Polish is. In (Bojanowski et al., 2017) authors extend the skip-gram model by building a vector representation of character n-grams and constructing the word representation as a sum of the character n-grams embeddings (for n-grams appearing in the word). It allows generating word embeddings for words not seen in the training corpus. In performed experiments, we used pre-trained vectors for Polish language (Kocoń and Gawor, 2019)[2]. Since texts differ in document length, the feature vectors representing a document were gained by averaging vector representations of individual words. This approach is known as doc2vec (Le and Mikolov, 2014).

3.3.2 BERT

The newest approaches to language modeling are inspired by deep-learning algorithms and context-aware methods. The state of the art is BERT (Devlin et al., 2018). Due to its bidirectional representation, jointly built on both the left and the right context, BERT looks at the whole sentence before assigning an embedding to each word in it. Therefore, the embeddings are context-aware. In performed experiments, we used a BERT model for Polish: Polbert[3]. The model is capable of analyzing up to 512 subwords. Therefore longer texts were cut. As a feature vector, we have used the first (with index zero) token from the last Transformer layer.

3.4 Document similarity

The TF-IDF, doc2vec, and BERT methods represent documents as vectors in multi-dimensional space. Most of the clustering methods are distance- or similarity-based. Therefore we need

[1] https://github.com/CLARIN-PL/wnsim

[2] http://hdl.handle.net/11321/606

[3] https://huggingface.co/dkleczek/bert-base-polish-cased-v1

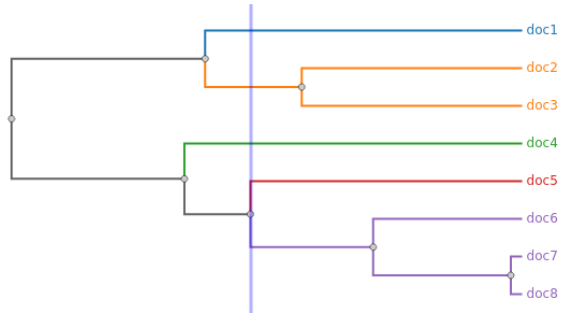

Figure 1: Example dendrogram (5 clusters)

to calculate the distance between vector-based representations of documents. We used popularly in natural language processing problems a cosine distance. It works well with sparse high-dimensional space (like TF-IDF is), and it is less noisy than Euclidean distance (Kriegel H-P., 2012). Moreover, it does not distinguish proportional vectors, which is often a desirable feature for word embedding.

4 Clustering Method

In our work, we decided to use the Agglomerative Clustering algorithm (Day and Edelsbrunner, 1984). The method iteratively joins samples into subgroups basing on a linkage criterion (in this case, an average distance).

The obtained dendrograms allowed us to determine the set of flat clusters for each different threshold defined by the joining points. A sample dendrogram with a fixed threshold is shown in Figure 1.

5 Quality Metrics

To evaluate the results, we decided to use *Adjusted Mutual Information* (Hubert and Arabie, 1985) score that allows comparison between two different clusterings. We may have used another measure (such as the Adjusted Rand Index), but according to Romano et al. (2016), AMI is the better choice because it performs well for unbalanced datasets. The score was calculated between the ground truth labels and all sets of labels obtained from the clustering algorithm.

Adjusted mutual information score is one of the information-theoretically based measures. It is based on mutual information (MI), which comes naturally from entropy.

Symbol	Description
X, Y	set of classes/clusters
H	entropy
MI	mutual information
NMI	normalized mutual information
AMI	adjusted mutual information
x_i, y_i	i-th element of X/Y (class or cluster)
$P(x_i), P(y_i)$	probability of the document being in i-th class or cluster
$P(x_i \hat{y}_j)$	intersection of $P(x_i)$ and $P(y_j)$
$E(MI)$	expected value of MI

Table 3: Symbols description

$$H(X) = \sum_i P(x_i) \log \frac{1}{P(x_i)}$$

$$MI(X,Y) = \sum_i \sum_j P(x_i \cap y_j) \log \frac{P(x_i \cap y_j)}{P(x_i)P(y_i)}$$

The problem with mutual information is that the maximum is reached not only when labels from one set (clusters) match perfectly those from the other (classes), but also when they are further subdivided. The simple solution for that is to normalize MI by mean of entropy of X and Y:

$$NMI(X,Y) = \frac{MI(X,Y)}{(H(X) + H(Y))/2}$$

Normalized mutual information can be further improved ("corrected for a chance") by subtracting the expected value of MI from nominator and denominator:

$$AMI(X,Y) = \frac{MI(X,Y) - E(MI)}{(H(X) + H(Y)/2 - E(MI)}$$

6 Evaluation

6.1 Configuration

For word2vec, TF-IDF, and Wu-Palmer methods, we used four variants with a different subset of words:

- *allposes* — all words, i.e., nouns, verbs and adjectives,
- *noun*, *verb* and *adj* — only nouns, verbs and adjective were used, respectively.

6.2 Results

For all four datasets, the BERT method obtained significantly lower results than the other methods (see Figure 2). The problem might be related to

Method	ALL			PPKZ			Market			Higher education		
	n	AMI	rank	n	AMI	rank	n	AMI	rank	n	AMI	rank
bert	179	0.360		50	0.095		104	0.344		370	0.287	
doc2vec-allposes	375	0.508	3	36	0.390	3	82	0.498	3	154	0.449	
doc2vec-verb	512	0.333		106	0.262		85	0.293		570	0.275	
doc2vec-adj	358	0.494		51	0.386	4	68	0.392		94	0.464	3
doc2vec-noun	414	0.474		75	0.343		80	0.476		128	0.438	
tfidf-allposes	81	0.497		65	0.333		24	**0.550**	1	39	0.418	
tfidf-verb	139	0.430		90	0.302		47	0.379		193	0.353	
tfidf-adj	73	0.529	2	90	0.289		37	0.460		22	**0.507**	1
tfidf-noun	106	0.501	4	65	0.317		25	0.505	2	46	0.435	
wupalmer-allposes	258	0.488		37	**0.452**	1	69	0.496	4	203	0.458	4
wupalmer-verb	208	0.321		183	0.213		156	0.217		584	0.259	
wupalmer-adj	207	**0.536**	1	36	0.441	2	63	0.398		57	0.499	2
wupalmer-noun	503	0.454		43	0.386		75	0.470		275	0.398	

Table 4: Summary of AMI scores for all dataset and method variants. The table contains the highest value of MRI score and the number of groups for which the score was obtained.

how the vector representing a document is generated — only the first 512 subwords are taken. At the same time, the documents are much longer, and some information is lost. However, experiments on supervised classification (which are not discussed herein) using the same BERT model (Polbert plus classification layer, working on the first 512 subwords) show that BERT tuned on a downstream task gives better results than doc2vec, and TF-IDF approaches. This, as well as results reported by Walkowiak and Gniewkowski (2019), could suggest that document features generated directly from the BERT language model (without retraining on a downstream task) are not suitable for the document to document similarity analysis.

In Table 4, we presented the highest AMI scores obtained for each method and dataset. For the ALL dataset, the highest scores were obtained by Wu-Palmer and TF-IDF, both using adjectives only. The AMI values were 0.536 and 0.529, respectively. A slightly lower result was obtained by doc2vec using all words — AMI value of 0.508.

For two out of three datasets, the best score was obtained by the TF-IDF. For the Market dataset, the advantage over any other method was significant and came to 0.05 points. In turn, for the Higher education dataset, the advantage over Wu-Palmer was lower than 0.01 points. For PPKZ, the Wu-Palmer method obtained the highest score, and the advantage over other methods was significant — 0.6 points (see Figure 3).

We also observed that for all methods based solely on verbs, the scores were significantly lower by 0.1–0.2 than for adjectives and nouns. For the ALL, PPKZ, and Higher education datasets the top scores were obtained on adjectives solely. The advantages over nouns and verbs were significant. Figure 4 presents the difference between Wu-Palmer variants on the ALL dataset.

6.3 Performance

We measured the computation time for two stages separately:

- document preprocessing (pre) — morphological tagging and word-sense disambiguation (for Wu-Palmer only). For preprocessing, we used CLARIN-PL web services[4] (Walkowiak, 2018) — a MorphoDita tagger (Walentynowicz, 2017) and WoSeDon (Janz et al., 2018) — a WSD tool.
- similarity computing (sim) — time required to generate the distance matrix on a single CPU thread.

Method	Pre	Sim	Total
doc2vec	2.0	3.6	5.6
TF-IDF	2.0	24.5	26.5
Wu-Palmer	7.0	563.0	570.0
BERT	2.0	1234.0	1236.0

Table 5: Processing times (in minutes) for different methods for the ALL dataset.

In Table 5, we present times required to process the ALL dataset. The fastest was doc2vec, which required only less than 6 minutes to process 3024 documents. TF-IDF was five times slower and required ca. 26 minutes. Wu-Palmer was 100 times slower than doc2vec, and BERT was 200 times slower.

[4] https://ws.clarin-pl.eu/wsd.shtml

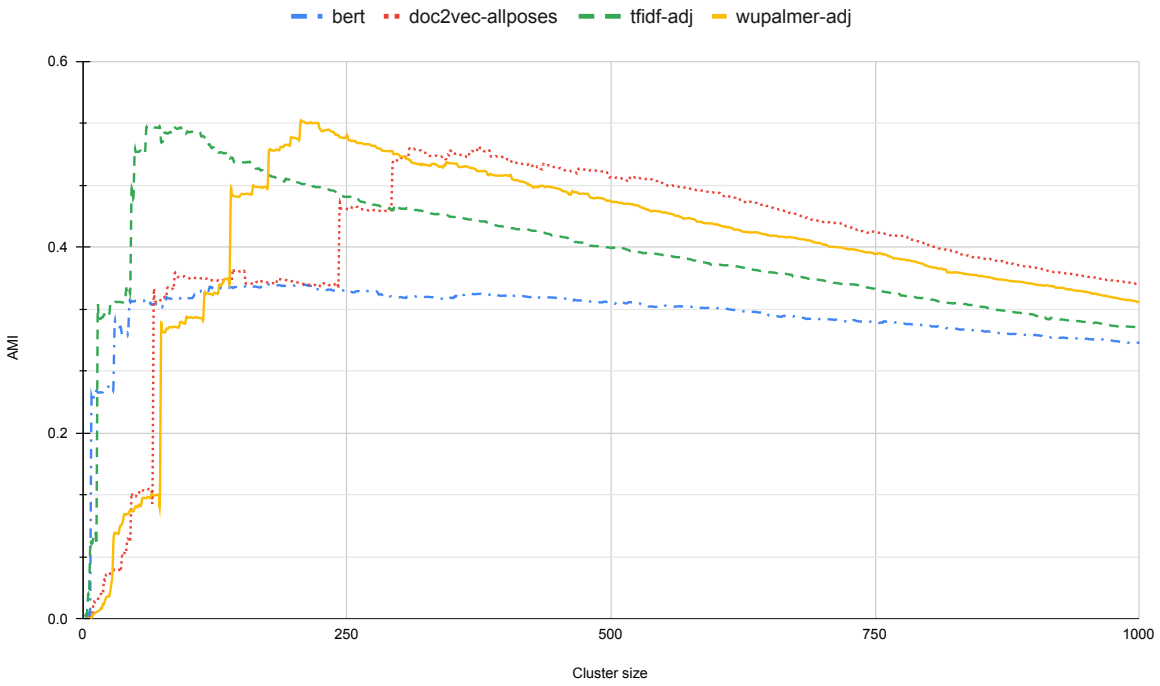

Figure 2: AMI values for the best-performing variants for each method on the ALL dataset.

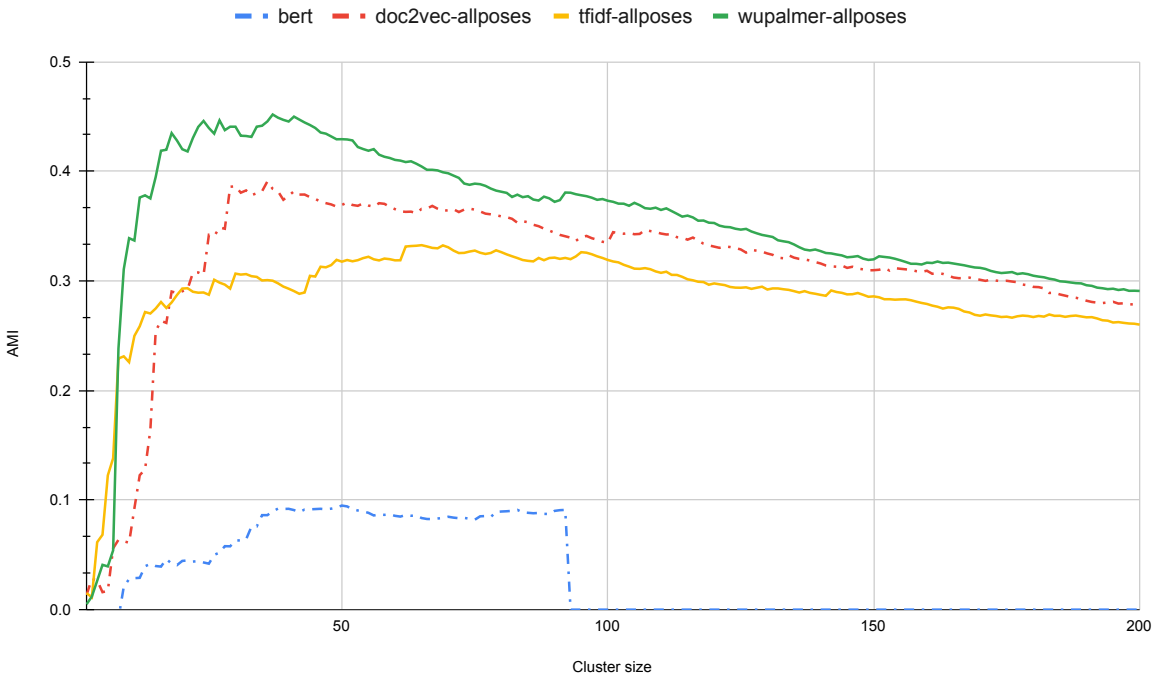

Figure 3: AMI values for the best-performing variants for each method on the PPKZ dataset.

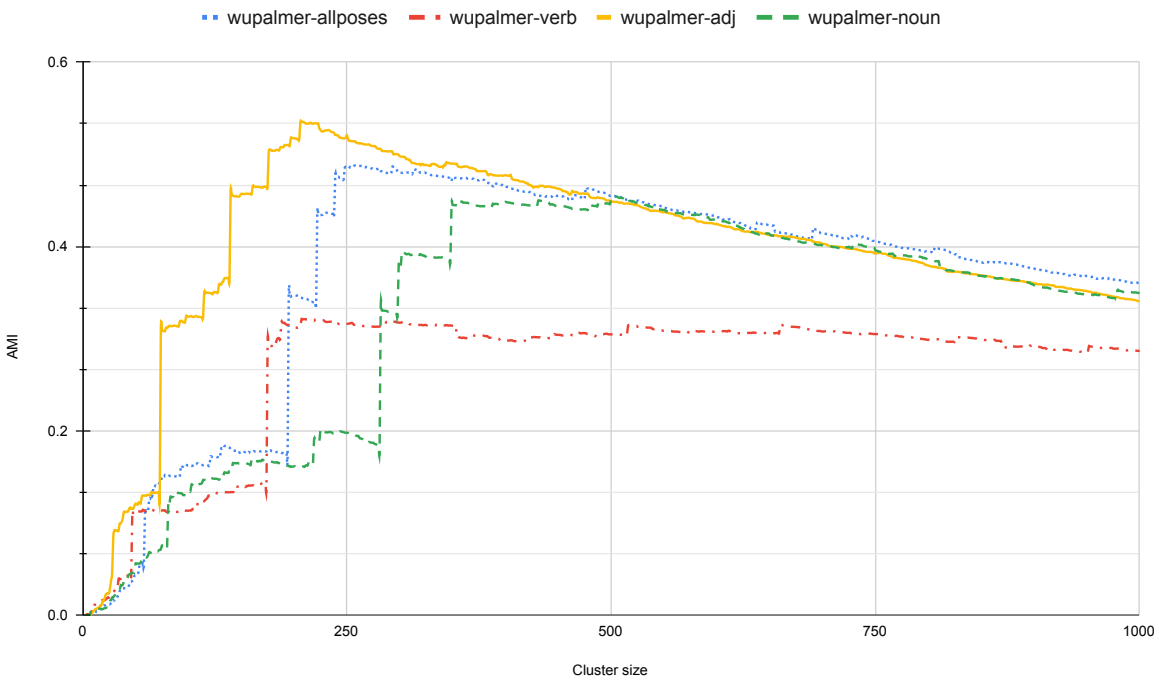

Figure 4: AMI values for each Wu-Palmer variant on the ALL dataset.

The Wu-Palmer method could be easily accelerated by paralleling — for 64 threads, the computation time can be reduced from 563 to 18 minutes. Another way to improve the processing speed would be reducing the number of synsets used to represent the document — as the number of synsets increases, processing time increases exponentially. We could apply the same technique as for TF-IDF — limit the number of synsets by defining the minimal document frequency for synsets.

7 Conclusion

The obtained results confirm the importance of developing dictionaries, knowledge bases, and domain ontologies. Wordnet-based measures of similarity may compete with embedding-based approaches in the task of text document clustering. Our research shows that the Wu-Palmer similarity metric can obtain comparable or even better (for the PPKZ dataset) results than the classical TF-IDF method and the modern doc2vec approach.

As far as the similarity of qualifications based on learning outcomes is concerned, one of the challenges discovered during our work was that the domain similarity and groups of qualifications were distorted by their source. Qualifications from the same source, e.g., from the same university or curriculum, tend to contain common, formulaic phrases. This problem will be addressed in further work.

Acknowledgements

The paper was prepared as part of the project "Operating and Developing of the Integrated Register of Qualifications" implemented by the Educational Research Institute as commissioned by the Ministry of National Education, co-financed by the European Union under the Operational Programme Knowledge Education Development.

References

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Alexander Budanitsky and Graeme Hirst. 2006. Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1):13–47.

Cedefop. 2017. Defining, writing and applying learning outcomes: A european handbook.

William H. E. Day and Herbert Edelsbrunner. 1984. Efficient algorithms for agglomerative hierarchi-

cal clustering methods. *Journal of Classification*, 1(1):7–24, Dec.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Lawrence Hubert and Phipps Arabie. 1985. Comparing partitions. *Journal of Classification*, 2(1):193–218, Dec.

IBE. 2020. Qualifications registers in selected european union countries.

Arkadiusz Janz, Paweł Kędzia, and Dominik Kaszewski. 2018. Word sense disambiguation tool WoSeDon. CLARIN-PL digital repository.

Jay J. Jiang and David W. Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. *CoRR*, cmp-lg/9709008.

Jan Kocoń and Michal Gawor. 2019. Evaluating KGR10 polish word embeddings in the recognition of temporal expressions using bilstm-crf. *CoRR*, abs/1904.04055.

Zimek A. Kriegel H-P., Schubert E. 2012. A survey on unsupervised outlier detection. *Statistical Analysis and Data Mining*, pages 363–387.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International Conference on Machine Learning*, pages 1188–1196.

C. Leacock and M. Chodorow. 1998. Combining local context and wordnet similarity for word sense identification. In Christiane Fellfaum, editor, *MIT Press*, pages 265–283, Cambridge, Massachusetts.

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of the 17th international conference on Computational linguistics*, pages 768–774, Morristown, NJ, USA. Association for Computational Linguistics.

Marek Maziarz, Maciej Piasecki, Ewa Rudnicka, Stan Szpakowicz, and Paweł Kędzia. 2016. PlWordNet 3.0 – a Comprehensive Lexical-Semantic Resource. In Nicoletta Calzolari, Yuji Matsumoto, and Rashmi Prasad, editors, *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 2259–2268. ACL, ACL.

Rada Mihalcea, Courtney Corley, and Carlo Strapparava. 2006. Corpus-based and knowledge-based measures of text semantic similarity. volume 1, 01.

Philip Resnik. 1995. Using information content to evaluate semantic similarity in a taxonomy. In *IJCAI'95: Proceedings of the 14th international joint conference on Artificial intelligence*, pages 448–453, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Simone Romano, Nguyen Xuan Vinh, James Bailey, and Karin Verspoor. 2016. Adjusting for chance clustering comparison measures. *The Journal of Machine Learning Research*, 17(1):4635–4666.

Gerard Salton and Chris Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Inf. Process. Manag.*, 24(5):513–523.

Wiktor Walentynowicz. 2017. MorphoDiTa-based tagger for polish language. CLARIN-PL digital repository.

Tomasz Walkowiak and Mateusz Gniewkowski. 2019. Evaluation of vector embedding models in clustering of text documents. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 1304–1311, Varna, Bulgaria, September. INCOMA Ltd.

Tomasz Walkowiak. 2018. Language Processing Modelling Notation – Orchestration of NLP Microservices. In Wojciech Zamojski, Jacek Mazurkiewicz, Jarosław Sugier, Tomasz Walkowiak, and Janusz Kacprzyk, editors, *Advances in Dependability Engineering of Complex Systems*, pages 464–473, Cham. Springer International Publishing.

Zhibiao Wu and Martha Palmer. 1994. Verbs semantics and lexical selection. pages 133–138, 01.