Review of "On the Replicability of Combining Word Embeddings and Retrieval Models"

Hedda Drexler
Informatics
Technical University of Vienna
Vienna, Austria
hedda.drexler@tuwien.ac.at

Michael Hermann-Hubler
Informatics
Technical University of Vienna
Vienna, Austria
e1254583@student.tuwien.ac.at

Kayoung Lee
Informatics
Technical University of Vienna
Vienna, Austria
e12024015@student.tuwien.ac.at

ABSTRACT

We made an attempt to reproduce a scientific paper in which the authors have replicated the experiments on comparing the mixture model of von Mises-Fisher (VMF) along with other models, such as Latent Semantic Indexing implementation (LSI) and Paragraph Vector algorithm with both implementations (PV) to name a few, by measuring their performance in three respective objectives: classification, clustering and information retrieval. From the three tasks we were able to reproduce one, while one was not fully reproducible and for one the reproduction was not possible due to missing data.

KEYWORDS

Classification, Clustering, Information Retrieval, Reproducibility, Word Embedding

1 Introduction

Scientific research suffers under the demand of "publish or perish", which leads far too often to pressured early submissions, thereby reducing the quality of the publication and its reproducibility.

Reproduced works are not well acknowledged in modern academic communities since reproducing is not considered as an original research, even though reproducing a work can take as much time and resources as a small research project..

However, with scientific papers in which the authors willingly shared the components from the PRIMAD model (Platform, Research Objective, Implementation, Method, Actors, Data), it easens reproduction of a scientific paper to validate the results from the original research or to further apply it to other settings or cases.

In this report, we want to reproduce the experiments conducted by Papariell et al. [3] so that we could either confirm the findings or challenge the conclusions.

2 Methods

The datasets used in the experiments were publicly available and the codes were saved in an open access repository. [1] The datasets have undergone the same pre-processing, such as tokenization of the texts and vectorization of the documents.

There were 8 models in total:

- Term Frequency-Inverse Document Frequency method (from here on as TF-IDF)
- Latent Semantic Indexing (LSI)
- Latent Dirichlet Allocation (LDA)
- Word2vec in the Continuous Bag-of-Word architecture (cBOW)
- Paragraph Vector with Distributed Bag-of-Words implementation (PV-dBOW)
- Paragraph Vector with Distributed Memory implementation (PV-DM)
- Fisher Vectors based on a Gaussian mixture model (FV-GMM)
- Fisher Vectors based on Mises-Fisher distributions (FV-moVMF).

2.1 Classification

The specific datasets for the classification experiment were subjectivity datasets v1.0 and sentence polarity dataset v1.0 [5]. The classification models were implemented as a separate Python code file. Both the datasets and the code file were executed through a Jupyter Notebook file.

The strength of each of the 8 models as a logistic regression classifier were measured and stored as a CSV file. The accuracy of the three models (cBOW, PV-dBOW, and PV-DM) were additionally measured, depending on the number of epochs they were trained with.

There were slight compatibility issues where the recent Python libraries did not work in the code file and had to be either changed to the version that worked or be replaced with Python code lines. Also the Jupyter Notebook file had to be executed at least two times with a bit of change in the codes to measure the performance, once as 50 dimensions and another time as 100 dimensions.

2.2 Clustering

The publicly available sklearn "20 newsgroups" data set [8] was used for the clustering. In the original paper several different

clustering methods were compared with each other. This included all aforementioned models and dimensions.

Even though the code was as a Jupyter Notebook and python file available, some parts including the preprocessing steps (tokenization) could initially not be executed without error. The package spherecluster could not be loaded without changing the original code.

2.3 Data Retrieval

The Data Retrieval was done using the TREC Robust04 retrieval dataset [5]. The main objective was the comparison of the BM25 retrieval algorithm compared to other retrieval methods, namely LSI, LDA, cBow, FV-GMM and FV-moVMF. As well as the supplementation of the BM25 algorithm with these methods.

This part of the original paper by Papariello et al. [3] was originally written in Java and Perl. As we wanted to use the same language for all parts of this paper, to improve readability, we decided to not use any of the original source code, but instead use the original code as a template and rewrite it in Python.

The queries were split, like by Papariello et al. [3] into title only, description only and title and description, to be later used in the retrieval.

We used the k1 and b values which Parapriello et al [2] found to yield the best results. (k1 = 1.2 and b = 0.75).

The retrieval scores were then normalized over all scores using the min-max normalization method.

3 Results

3.1 Classification

Mean Accuracy and Standard Deviation of the Top Values

Model	50-dim.		100-dim.	
	Subj	Sent	Subj	Sent
TF-IDF	88.6 +/- 0.5	74.8 +/- 0.4	88.6 +/- 0.5	74.8 +/- 0.4
LSI	82.5 +/- 1.0	63.2 +/- 0.8	84.2 +/- 1.3	65.3 +/- 1.1
LDA	61.6 +/- 1.7	55.9 +/- 1.5	63.8 +/- 2.2	57.7 +/- 1.4
cBow	89.0 +/- 1.1	70.1 +/- 1.3	89.2 +/- 1.1	71.6 +/- 1.4
PV-DBOW	88.6 +/- 0.8	65.5 +/- 1.6	89.2 +/- 0.8	67.9 +/- 1.2
PV-DM	83.7 +/- 1.2	68.4 +/- 1.4	88.0 +/- 1.1	69.9 +/- 1.0
FV-GMM	88.4 +/- 1.0	70.0 +/- 1.2	88.7 +/- 0.8	72.2 +/- 1.2
FV-moVMF	89.2 +/- 1.1	70.0 +/- 1.3	88.7 +/- 1.4	71.6 +/- 1.5

Table 1: Result of classification experiment with two data sets, subjectivity datasets v1.0 (subj) and sentence polarity dataset v1.0 (sent). The values shown are mean accuracy and standard deviation of the *top* values.

Mean Accuracy and Standard Deviation of the Mean Values

Model	50-dim.		100-dim.	
	Subj	Sent	Subj	Sent
TF-IDF	88.6 +/- 0.5	74.8 +/- 0.4	88.6 +/- 0.5	74.8 +/- 0.4
LSI	81.2 +/- 1.2	62.5 +/- 1.3	83.3 +/- 1.2	64.9 +/- 1.3
LDA	58.0 +/- 1.4	54.2 +/- 1.4	60.8 +/- 1.4	55.8 +/- 1.5
cBow	88.6 +/- 1.0	69.2 +/- 1.3	88.9 +/- 0.9	70.6 +/- 1.4
PV-DBOW	88.4 +/- 0.9	65.2 +/- 1.4	89.2 +/- 0.9	67.7 +/- 1.3
PV-DM	83.6 +/- 1.1	67.7 +/- 1.2	87.4 +/- 1.0	69.4 +/- 1.3
FV-GMM	88.3 +/- 1.0	69.7 +/- 1.4	88.5 +/- 0.9	71.8 +/- 1.2
FV-moVMF	88.5 +/- 0.9	69.3 +/- 1.3	86.5 +/- 1.2	70.7 +/- 1.5

Table 2: Result of classification experiment with two data sets, subjectivity datasets v1.0 (subj) and sentence polarity dataset v1.0 (sent). The values shown are mean accuracy and standard deviation of the *mean* values.

In the original paper, there is a table with mean accuracy and its standard deviation per model but the values that were chosen were "top values" from the performance graphs that were produced for each model.

Choosing "top value" did not seem like the most optimal value to represent the average performance since the objective of the original paper was to investigate whether FV-moVMF was better in performing tasks, such as classification, clustering, and data retrieval, than FV-GMM and proving that one model performed better than the other in a certain situation does not equate that the former model has better performance than the latter.

As a result, the reproduced metrics show a similar result to the original found metrics. The ranking of the strength of each model is also in the same order as that of the original.

3.2 Clustering

The clustering was performed on the whole data set, merging the training and test data sets which are openly available. For creating the models the dimensions 20 and 100 were used (code), while in the publication 20 and 50 were stated.

For some of the models Adjusted Rand Index (ARI) and Normalised Mutual Information (NMI) were computed. With comparison to the original paper, slight differences could be observed (if the 50-dim in the original paper is a typo and ment actually 100-dim, which would have been used according to the provided code) and are listed in Table 3. If the values are from 50-dim, then the hereby calculated values are in the same range as the original source.

model	original data		reproduced data	
	50-dim	50-dim	100-dim	100-dim
	ARI	NMI	ARI	NMI
TF-IDF	0.4 +/- 0.1	5.6 +/- 0.3	0.5 +/- 0.1	3.6 +/- 0.2
LSI	0.5 +/- 0.2	5.8 +/- 0.3	0.6 +/- 0.1	3.8 +/- 0.2
LDA	20.8 +/- 1.1	40.2 +/- 0.8	-	-
cBow	31.5 +/- 0.3	51.2 +/- 0.3	24.4 +/- 0.7	43.4 +/- 0.7
PV-DBOW	53.3 +/- 1.3	66.1 +/- 0.6	45.9 +/- 1.3	63.6 +/- 0.7
PV-DM	30.1 +/- 0.9	53.3 +/- 0.5	9.4 +/- 1.5	37.9 +/- 1.1
FV-GMM	1.0 +/- 0.1	9.8 +/- 1.3	0.78 +/- 0.01	3.9 +/- 0.1
FV-moVMF	1.6 +/- 0.2	14.9 +/- 1.6	-	-

Table 3: Comparison of results of clustering experiment with 20 newsgroups dataset and 100 dimensions between this paper and Parapriello et al [3]

3.3 Data Retrieval

At the time of this report we were unfortunately unable to complete this experiment as we found that another dataset was used by Parapriello et al [3], which supplied the feature vectors of the additional Models which we were unable to get.

We compared the processing steps used in the reference paper with other papers and found that the steps were done correctly, but some further finetuning could be done using the below stated methods.

The stop word removal in the experiment was done using the Apache Lucene Standard Stop words, which includes 33 words [2]. Decreasing the stop word count might improve the performance as described by Trotman et. al. [6].

Similarly, the default Stemming algorithm is used. Here additional gains could be obtained, using different stemming and synonym algorithms, as stemming is one of the relevant factors for increasing BM25 improvements [6].

Another improvement to the retrieval results could be achieved using relevance feedback and Query Expansion as described by Trotman et. al. [6] and Robertson and Zaragosa [4]

4 Conclusion

For classification experiments, we were able to confirm the findings. The datasets were neither replaced nor updated over the past few years and therefore could be concluded that they are the same as the ones used in the original experiment. Even the codes, which have been slightly edited due to compatibility issues, would not have affected the overall outcome of the reproduced experiment. The only change that the edited codes may have caused is the difference in runtime or efficiency.

The part for the clustering models could not be fully reproduced. The loading and preprocessing of the data showed inconsistencies in the provided jupyter notebook, also the used package ‘sphere cluster’ had to be modified to start the experiments. Throughout the different models parts of the code seemed to be missing or not defined. Also result files (e.g. for the TF-IDF model) are missing and only the final summary is presented in the paper. After slight modifications some of the models could be executed, some showing slightly different values and some similar results than the original paper.

Currently we cannot conclude if the data retrieval part of the original paper can be reproduced. This part should be completed using either the missing data, or by trying to create feature vectors from the missing models.

REFERENCES

[1] ECIR_2020: *https://rsagit.researchstudio.at/lpapariello/ecir_2020/-/tree/master*. Accessed: 2021-01-24.

[2] Lucene - StopAnalyzer.java: *http://alvinalexander.com/java/jwarehouse/lucene/src/java/org/apache/lucene/analysis/StopAnalyzer.java.shtml*. Accessed: 2021-01-24.

[3] Papariello, L. et al. 2020. On the Replicability of Combining Word Embeddings and Retrieval Models. *Advances in Information Retrieval* (Cham, 2020), 50–57.

[4] Robertson, S. and Zaragoza, H. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrieval*. 3, (Jan. 2009), 333–389. DOI:https://doi.org/10.1561/1500000019.

[5] Sentiment Polarity Data: *http://www.cs.cornell.edu/people/pabo/movie-review-data/*. Accessed: 2021-01-24.

[6] Trotman, A. et al. B.: Improvements to BM25 and language models examined. *In: ADCS (2014*.

[7] Voorhees, E.M. 2005. The TREC Robust Retrieval Track. *SIGIR Forum*. 39, 1 (Jun. 2005), 11–20. DOI:https://doi.org/10.1145/1067268.1067272.

[8] sklearn.datasets.fetch_20newsgroups — scikit-learn 0.24.1 documentation: *https://scikit-learn.org/stable/modules/generated/sklearn.datasets.fetch_20newsgroups.html*.