Text Clustering based on Multi-View Representations

Eya HAMMAMI[1,2,*,†], Rim FAIZ[1,3,†]

[1]LARODEC Laboratory, University of Tunis, Tunisia
[2]IRIT Laboratory, University of Toulouse III - Paul Sabatier, France
[3]IHEC, University of Carthage, Tunisia

Abstract

Multi-View learning has grown in popularity in data mining and machine learning domains. Therefore, Multi-View semi-supervised learning or unsupervised learning has gained significant interest from the research community. Recently, Multi-View Clustering (MVC) methods for textual data present an efficient solution to merge different representations called "views" by utilizing the integrality characteristics of these views. However, the existing approaches generally take into consideration only one representation process for all views that are based on term frequencies. Such representation leads to losing precious information or failing to capture the semantic aspect and the contextual meaning of the text. To cope with these issues, we propose a novel method for Multi-View text clustering that exploits different representations of text in order to improve the quality of clustering. The experimental results show that the proposed approach outperforms some baseline methods and boosts the quality of the text clustering.

Keywords

Machine Learning, Clustering Multi-View, NLP, Embedding, Transformers, BERT, SBERT, Text mining

1. INTRODUCTION

Textual clustering intends to partition a collection of documents into groups of clusters, which applies that documents in the same cluster are similar, whereas those in different clusters are dissimilar[1]. Different approaches for text clustering have been explored in the last few years. Most of the approaches belongs to partitional clustering, such as hierarchical clustering or K-means algorithms. Naturally, carrying out clustering algorithms on textual data needs a preliminary step wherein the text data is converted into a structured form using for example the Vector Space Model (VSM) [2], which is the most commonly used representation of text. Despite its popularity, this model has two major weaknesses, i.e., it loses the ordering of the words and fails to capture semantic relation between the words [3]. Thus, different sentences can have exactly the same representation, as long as the same words are used.
Benefiting from the rapid development of natural language processing (NLP), many neural network language models (NNLM) have been used to address representation problems in text clustering. Word embeddings trained by Word2vec [4] and Glove [5] are commonly applied as basic building blocks for text representation. However, these word embeddings are

CIRCLE'22: Conference of the Information Retrieval Communities in Europe, July 04–07, 2022, Samatan, Gers, France
*Corresponding author.
†These authors contributed equally.
hammami.eya@isg.u-tunis.tn (E. HAMMAMI); rim.faiz@ihec.ucar.tn (R. FAIZ)

uncontextualized and neglect the polysemy of words. Therefore, BERT (Bidirectional Encoder Representations from Transformers) model [6] comes to deal with the issues mentioned above. This language representation model makes use of a huge amount of plain text data and is trained in an unsupervised way. Most of the research community utilized BERT models in the embedding module to generate contextualized sentence embeddings for the textual clustering task [7]. Besides, taking into consideration different representations of the data can prompt enhancement in the quality of clustering [8]. Merging various representations of data is admitted as Multi-View Clustering, the dissimilar representations are referred to as "views", where each view picks up one-sided information that is not captured by other views [9, 10]. The study in [11] presents that the Multi-View versions of K-means and EM perform better than their single-view representation. In addition to that, data noise and aberrations that exist in one view may be fulfilled by other views. There are many research studies that have utilized the aspect of Multi-View data by assimilating the specific properties of each view in the learning approach to raise the performance of existing machine learning and deep learning algorithms [12, 13]. Actually, the current approaches have enhanced the clustering results. However, these methods depend only on the syntactic aspect of text i.e. term occurrences... Although views are represented with a just single model and the most commonly used is the Term Frequency weighting [14]. In addition, such representation is able to give an insight of a word's relative importance in a document, but it does not supply any idea about word semantics and contextual meaning.

In this paper, we propose a Multi-View text clustering that coverages different text representation models derived from BERT. The objectives of our proposal is to capture and assimilate information from each view and preserve the contextual and semantic facets of text.
Our main contributions regards:

- First, produce views to extract different information in order to improve the quality of clusters.
- Second, take into consideration different partitions that are derived from different views by merging them to get the final consensus clustering.

This paper is organized as follows: Section. 2 overview of some related work. The proposed Multi-View method for text clustering is described in Section. 3. The experimental results are presented in Section. 4.

2. RELATED WORK

Recently, Multi-View Clustering methods has attracted considerable attention by their abilities to provide natural way of generating clusters from Multi-View data. We have found different classes of Multi-View Clustering approaches [13, 15]. The first class consists of incorporating Multi-View integration into the clustering process through optimizing certain loss functions [11, 16]. In the second class, the majority of methods consists of projecting Multi-View data into a common lower dimensional subspace and then applying clustering algorithm to learn the partition [17, 18]. Finally, the third class is called late fusion, in which a clustering method is derived from each view and then all the partitions are merged base on consensus [19, 20]. Our

proposal belongs to the late fusion category, that merges clustering results obtained separately from each view on a consensual basis, such as, Multi-View kernel k-means clustering ensembles (MvKKMCE) and Multi-View spectral clustering ensembles (MvSpecCE) that were introduced in [21]. Also, Zhijie Xu and Shiliang Sun in [22] expanded the famous algorithm of boosting Adaboost to Multi-View learning process. Multi-View Clustering learning is also utilized to compromise with high dimensionality problems[12]. The work in [23] carries out sparse decomposition and low-rank in order to obtain the final consensus clustering. However, in the majority of the proposed methods, all the views for textual data are using only syntactic features i.e. the bag of words as views, which cannot conserve the semantic aspect of text, and as a result precious information is lost. Therefore, for the same document, bearing in mind different representations of text as views can enhance the clustering efficiency. Hence, Our proposed method consists of merging multiple clustering using models derived from BERT Encoder as text representation models and obtaining a final single consensus clustering.

3. MULTI-VIEW TEXT CLUSTERING APPROACH

For the purpose of clustering text documents, we introduce a new Multi-View method that takes as input a collection of documents, gets views using distinct representations of text derived from BERT's models and the output is clusters of documents. The proposed method is composed of three main steps:

3.1. First step: Text representation

This step consists of generating views in such a way that documents are represented with diverse representation models derived from BERT and more specifically we used pre-trained encoders from Sentence-BERT (SBERT) [24], which is a modification of the BERT network using siamese and triplet networks that is able to derive semantically meaningful sentence embeddings. This enables BERT to be used for certain new tasks. Such as large-scale semantic similarity comparison, clustering, and information retrieval via semantic search... Among the pre-trained encoders [1] of SBERT we choose the following ones as sentence embeddings:

- all-MiniLM-L6-v2
- all-distilroberta-v1
- all-MiniLM-L12-v2
- paraphrase-multilingual-mpnet-base-v2
- paraphrase-albert-small-v2
- paraphrase-MiniLM-L3-v2

At a glance, Table 1 sketches the surveyed hyper-parameters that are used in these chosen pre-trained language models in terms of Max Sequence Length (MSL), Dimensions and Normalized Embeddings (NE). Regarding the Pooling strategy, all of these models use 'Mean Pooling' strategy also they use 'cosine-similarity' as a Score functions.

[1] https://www.sbert.net/docs/pretrainedmodels.html

Table 1
Hyper-parameter of embedding models

Embedding models	MSL	Dimensions	NE
all-MiniLM-L6	256	384	True
all-distilroberta	512	768	True
all-MiniLM-L12	256	384	True
paraphrase-multilingual-mpnet-base	128	768	False
paraphrase-albert-small	256	768	False
paraphrase-MiniLM-L3	128	384	False

This step is described by the following formula:

$$V_{i,E_j}(d) = V_i'(d) \tag{1}$$

where:

- V represent the view i of the document d, $i \in 1, .., n$.
- E represent the encoder j that we mentioned above (the pre-trained language models of SBERT) for the document d.
- V' represent the view i of the document d just after applying the Encoder E_j, $i \in 1, .., n$.

3.2. Second step: Clustering Views

In this step with the aim of getting various partitions, we feed text vectors to the k-means clustering algorithm. It divides n text documents into k clusters. k is defined in advance. Firstly, k documents are randomly chosen as initial centroids. Each document is assigned to the nearest centroid with distance or similarity measure and the relevant documents belonging to the same centroid are gathered into a cluster. Then new cluster centroids are calculated, and documents are rearranged. The measure value between each text document and cluster centroids iteratively updates the cluster centroids and reorganizes the clusters until the termination condition is met or there is no change in clusters.

3.3. Third step: Aggregation Partitions

This step, includes the using of an aggregation technique to merge the distinct partitions with the aim of obtaining the final consensus clustering of documents without accessing the features utilized to get those partitions. We used The Cluster Based Similarity Partitioning (CBSP) technique [25] as an instance-based method which transforms the set of clustering into a hypergraph representation, where the number of frequency of two objects, which are accrued in the same clusters, is considered as the weight of each edge [26]. Fig. 1 describes our proposal approach.

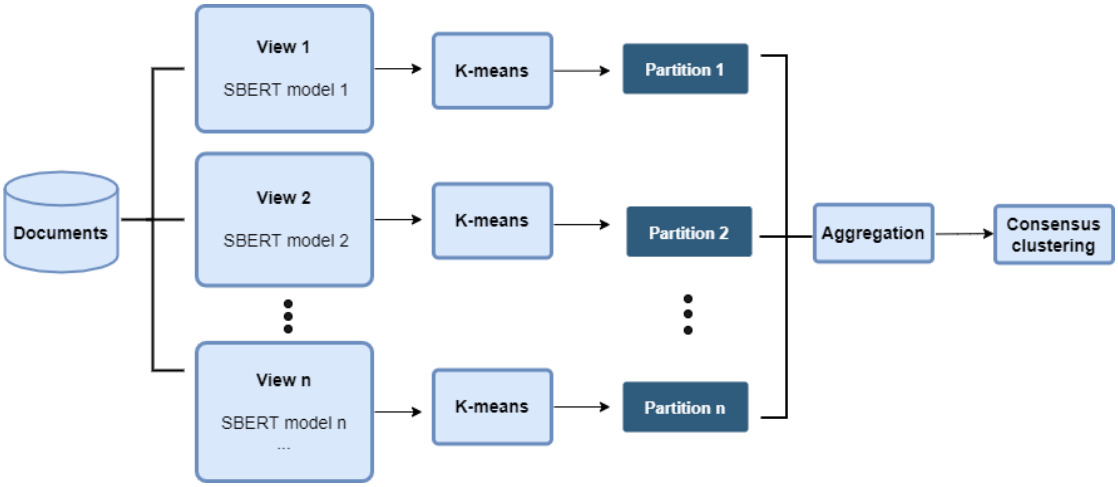

Figure 1: Multi-View approach for text clustering

4. EXPERIMENTS

For the purpose of evaluating the performance of our proposed approach, we exploit experiments on BBC News datasets referring to three evaluation metrics. Therefore, we conducted a comparison with the single view methods. Then we applied the aggregation and evaluated it according to the same evaluation measures that we used in the beginning.

4.1. Experimental Setup

Dataset. The experiments are conducted on public and open dataset from the BBC News [2] which is composed of 2225 news articles, each one annotated under one of the five following categories: business, entertainment, politics, sport or tech.

Evaluation Measures. To evaluate the quality of the clustering, three evaluation metrics are used: the Normalized Mutual Information (NMI) metric [27] that measures the quality of clustering by taking into consideration the number of clusters, the F-measure metric [28] which is a swap between Recall and Precision and the Purity metric [29] which measures the number of correctly allocated documents, where each cluster is assigned to the most common class in said cluster. For all metrics that we used in our experiments, the values range from 0 to 1, such that values closer to 1 represent good quality of clustering results however those which are closer to 0 indicate bad quality of clustering results.

[2]https://www.kaggle.com/c/learn-ai-bbc/overview

Table 2
Comparison of clustering results with single view methods according to NMI, Purity and F-measure metrics

Method	NMI	Purity
all-MiniLM-L6 + K-means	0.862	0.956
all-distilroberta + K-means	0.897	0.967
all-MiniLM-L12 + K-means	0.840	0.949
paraphrase-multilingual-mpnet-base + K-means	0.832	0.947
paraphrase-albert-small + K-means	0.783	0.926
paraphrase-MiniLM-L3 + K-means	0.779	0.923
Proposed approach	**0.899**	**0.980**

Table 3
Comparison of clustering results with single view methods according to F-measure metrics

Method	F-measure				
	Business	Entertainment	Politics	Sport	Tech
all-MiniLM-L6 + K-means	0.008	0.945	0.001	0.009	0.002
all-distilroberta + K-means	0.017	0.015	0.001	0.004	0.966
all-MiniLM-L12 + K-means	0.018	0.054	0.982	0.051	0.017
paraphrase-multilingual-mpnet-base + K-means	0.919	0.006	0.983	0.012	0.026
paraphrase-albert-small + K-means	0.002	0.906	0.014	0.038	0.925
paraphrase-MiniLM-L3 + K-means	0.894	0.907	0.002	0.907	0.010
Proposed approach	**0.901**	**0.914**	**0.988**	**0.902**	**0.972**

Hyper-parameters and settings. The hyper-parameters that are used in the first step of our proposed method are described in Table 1 in which we set the Max Sequence Length (MSL) values, the Dimensions values, the Normalized Embeddings (NE) that are used for each of our pre-trained language models. Then, in the second step which consist of applying a clustering algorithm for each view in order to get various partitions, we used the k-means clustering algorithm by fixing the k parameter to 5 which belong to the number of categories of our dataset. Finally, the third step regards the integration of all the obtained partitions together, we used The Cluster Based Similarity Partitioning (CBSP) algorithm. Here, a binary similarity matrix is constructed for each input clustering. Each column corresponds to a cluster: an entry has a value of 1 if the corresponding two points belong to the cluster, 0 otherwise. An entry-wise average of all the matrices gives an overall similarity matrix S. S is utilized to recluster the data using a graph-partitioning based approach.

4.2. Results and discussion:

Our proposed approach is evaluated by comparison with single view-based clustering methods. Each view correlates with a single text representation model derived from SBERT which are the all-MiniLM-L6, the all-distilroberta model, the all-MiniLM-L12, the paraphrase-multilingual-mpnet-base, the paraphrase-albert-small and the paraphrase-MiniLM-L3 model. We used the K-means algorithm for every single view method. As depicted in Table 3, the comparison

of clustering results with single view methods according to NMI and Purity metrics, shows that our proposed method outperforms all the single view methods by achieving 89% score in terms of NMI measure and 98 % for Purity measure. Table 3 shows also the comparison of clustering results according to the F-measure metric for each category of the dataset. Here, all-MiniLM-L6 representation model can detect more delicately the specificity of features for the category 'Entertainment' by achieving a 94% of score also all-distilroberta model detect the features of category 'Tech' with 96% of score and for paraphrase-multilingual-mpnet-base model which can captures the features of both categories 'Business' and 'Politics' by achieving a 91% and 98% scores for each one. Finally, the paraphrase-MiniLM-L3 model presents good results for Category 'Sport' with 90% of score. Based on these results we choose to combine both of these four representation models on the assumption that a smarter text Multi-View Clustering technique would be able to improve the clustering results. As a consequences the proposed method presents good results according to F-measure metric for all the clusters by achieving 90.1% of score for the category 'Business', 91.4 % of score for the category 'Entertainment', 98.8% of score for the category 'Politics', 90.2 % of score for the category 'Sport' and 97.2% of score for the category 'Tech'. The carried out experiments emphasize the relevance of our approach.

5. CONCLUSION

In this study, we proposed a new approach for Multi-View text clustering based on four different text representation models derived from SBERT which can capture The semantic and the contextual aspects of text. Then, the K-means algorithm is used to obtain distinct partitions from each view. Finally, the partitions are merged together using the Cluster Based Similarity Partitioning technique on the assumption to obtain the final consensus clustering. The experimental results in comparison with the single view based clustering shows that using multiple views models yields better results of clustering.

Acknowledgments

This work was supported by Google PhD Fellowships program.

References

[1] C. C. Aggarwal, C. Zhai, Mining text data, Springer (2012). URL: https://doi.org/10.1007/978-1-4614-3223-4. doi:NewYork.

[2] G. Salton, A. Wong, C.-S. Yang, A vector space model for automatic indexing, Communications of the ACM 18 (1975) 613–620.

[3] Q. Le, T. Mikolov, Distributed representations of sentences and documents, in: International conference on machine learning, PMLR, 2014, pp. 1188–1196.

[4] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, Proceedings of the 26th International Conference on Neural Information Processing Systems 26 (2013).

[5] J. Pennington, R. Socher, C. D. Manning, Glove: Global vectors for word representation, in: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), Association for Computational Linguistics, 2014, pp. 1532–1543.
[6] J. D. M.-W. C. Kenton, L. K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2019, pp. 4171–4186.
[7] A. Subakti, H. Murfi, N. Hariadi, The performance of bert as data representation of text clustering, Journal of big Data 9 (2022) 1–21.
[8] Y. Yang, H. Wang, Multi-view clustering: A survey, Big Data Mining and Analytics 1 (2018) 83–107.
[9] G. Chao, S. Sun, J. Bi, A survey on multiview clustering, IEEE Transactions on Artificial Intelligence 2 (2021) 146–168.
[10] S. Sun, A survey of multi-view machine learning, Neural computing and applications 23 (2013) 2031–2038.
[11] S. Bickel, T. Scheffer, Multi-view clustering, in: International Conference on Data Mining (ICDM), volume 4, IEEE, 2004, pp. 19–26.
[12] V. Kumar, S. Minz, Multi-view ensemble learning: an optimal feature set partitioning for high-dimensional data classification, Knowledge and Information Systems 49 (2016) 1–59.
[13] J. Liu, C. Wang, J. Gao, J. Han, Multi-view clustering via joint nonnegative matrix factorization, in: Proceedings of the 2013 SIAM international conference on data mining, SIAM, 2013, pp. 252–260.
[14] G. Salton, C. Buckley, Term-weighting approaches in automatic text retrieval, Information processing & management 24 (1988) 513–523.
[15] C. Yuan, Y. Zhu, Z. Zhong, W. Zheng, X. Zhu, Robust self-tuning multi-view clustering, World Wide Web 25 (2022) 489–512.
[16] A. Kumar, P. Rai, H. Daume, Co-regularized multi-view spectral clustering, Proceedings of the 24th International Conference on Neural Information Processing Systems 24 (2011).
[17] K. Chaudhuri, S. M. Kakade, K. Livescu, K. Sridharan, Multi-view clustering via canonical correlation analysis, in: Proceedings of the 26th annual international conference on machine learning, Association for Computing Machinery, 2009, pp. 129–136.
[18] M. B. Blaschko, C. H. Lampert, Correlational spectral clustering, in: 2008 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2008, pp. 1–8.
[19] E. Bruno, S. Marchand-Maillet, Multiview clustering: a late fusion approach using latent models, in: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval, Association for Computing Machinery, 2009, pp. 736–737.
[20] D. Greene, P. Cunningham, A matrix factorization approach for integrating multiple data views, in: Joint European conference on machine learning and knowledge discovery in databases, Springer, 2009, pp. 423–438.
[21] X. Xie, S. Sun, Multi-view clustering ensembles, in: 2013 international conference on machine learning and cybernetics, volume 1, IEEE, 2013, pp. 51–56.
[22] Z. Xu, S. Sun, An algorithm on multi-view adaboost, in: International conference on Neural information processing, Springer, 2010, pp. 355–362.

[23] Z. Tao, H. Liu, S. Li, Z. Ding, Y. Fu, From ensemble clustering to multi-view clustering, in: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI), 2017.

[24] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (????) 3982–3992.

[25] A. Strehl, J. Ghosh, Cluster ensembles—a knowledge reuse framework for combining multiple partitions, Journal of machine learning research 3 (2002) 583–617.

[26] E. Akbari, H. M. Dahlan, R. Ibrahim, H. Alizadeh, Hierarchical cluster ensemble selection, Engineering Applications of Artificial Intelligence 39 (2015) 146–156.

[27] F. Zhuang, G. Karypis, X. Ning, Q. He, Z. Shi, Multi-view learning via probabilistic latent semantic analysis, Information Sciences 199 (2012) 20–30.

[28] B. Larsen, C. Aone, Fast and effective text mining using linear-time document clustering, in: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 1999, pp. 16–22.

[29] F. Nie, G. Cai, X. Li, Multi-view clustering and semi-supervised classification with adaptive neighbours, in: Thirty-first AAAI conference on artificial intelligence, 2017, p. 2408–2414.