OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG FACULTY OF COMPUTER SCIENCE

Constrained Clustering of Text based on TREC 4 5 with textual explanations

Adrija Bhavana Felix Nikhil Sourima

Otto von Guericke University Magdeburg

Motivation

In recent years, there has been an explosion of text data, and it's crucial to mine actionable insights from these texts.In text clustering the texts are grouped together in a way that makes them more similar to each other than to those in other clusters. To further incorporate domain knowledge in the form of constraints, we use the constraint clustering approach which is a class of semi-supervised learning algorithms.

Problem Statement: Our task here consists of text clustering using constrained clustering methods and textual explanations of the clusters and the questions we try to answer are-

Why are a bunch of data points within a cluster together?

Why are two data points in two different clusters?

What is the global explanation that explains a selected cluster?

Dataset

The dataset we have used is the TREC 4-5. This document-set includes material from the Financial Times Limited, the Congressional Record of the 103rd Congress, the Federal Register, the Foreign Broadcast Information Service, and the Los Angeles Times.

Concept of Constrained Clustering and Explanations

- The algorithm we have chosen for our dataset is **PCKmeans** as it allows violation of constraints to some extent at the cost of some penalty.For our text data, it could happen that we introduce must-link constraints between two documents on different topics,simply because they share a lot of ngrams. So in this way it's better to keep the clustering flexible.
- **Explainability:** 1) For a given document we are checking its similarity with the other documents in the same cluster using Doc2Vec and getting the key-phrases out of it which leads us to understand the aspect of the document. 2)As an additional method to explain the cluster itself, we have trained a decision tree which gives us the cluster labels using the features and we are extracting the top representative words of each cluster which led to the decision.
- **Constraints generation:** The constraints are created by checking the similarity between all the documents using Doc2Vec method. If any two documents have similarity greater than a certain threshold(0.8 as per consideration) we say they can be connected with Must-link constraint, otherwise cannot-link.
- **Pros & Cons:** Doc2Vec computes a feature vector for every document in the corpus unlike Word2Vec which is computationally efficient. Also it generalizes to longer documents, and can learn from unlabelled data. Disadvantage-If the input corpus is one with lots of misspellings like tweets, this algorithm does not perform well.

Implementation

- During constrained generation we made two assumptions-
  - 1) We set the number of constraints less to avoid overfitting.
  - 2)The similarity threshold for being a must-link constraint we set as (>0 .8).
- The optimum k value is determined by the ELBO loss method performed on the K-means clustering, and in turn applied to the PCKMeans(Figure 1).
- Complete implementation steps are given in Figure 2.

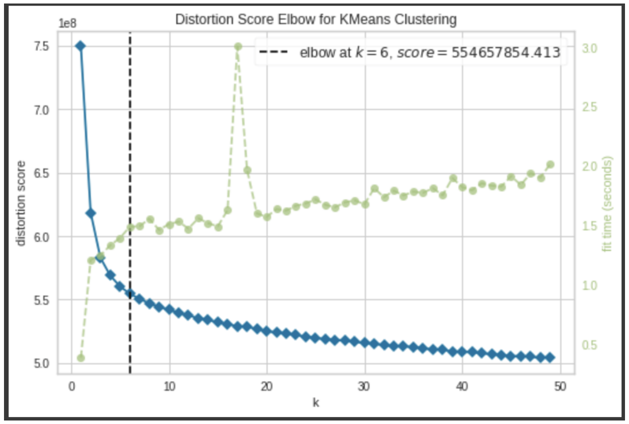
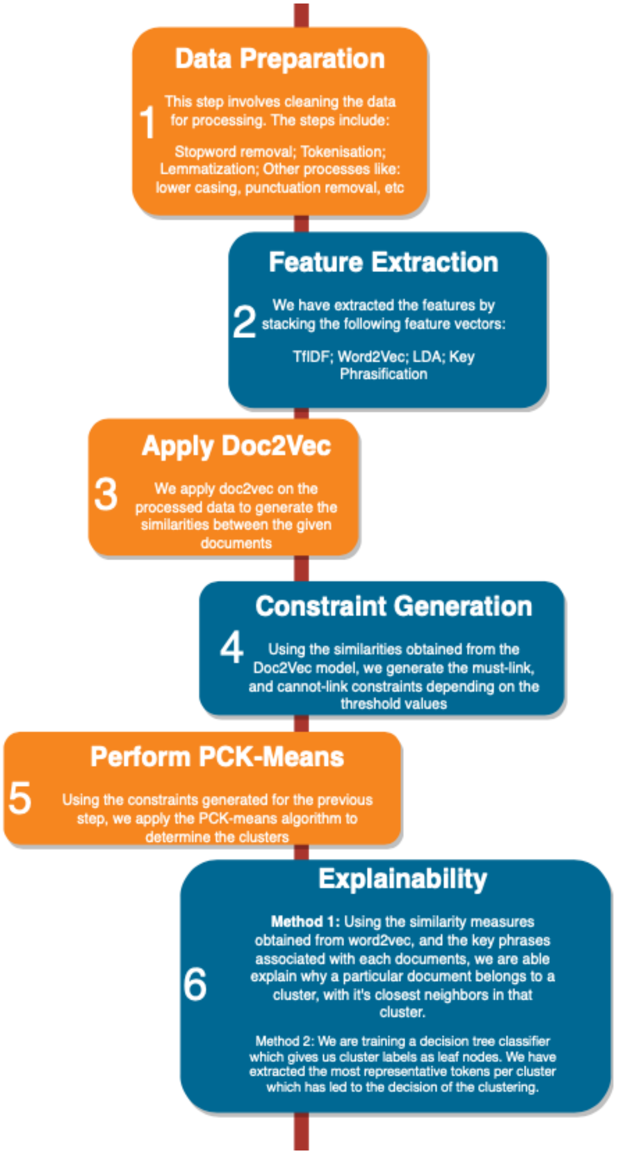


Figure1: ELBO Loss



Data Preparation

1 This step involves cleaning the data for processing. The steps include:

Stopword removal; Tokenisation; Lemmatization; Other processes like: lower casing, punctuation removal, etc

Feature Extraction

2 We have extracted the features by stacking the following feature vectors:

TfIDF; Word2Vec; LDA; Key Phrasification

Apply Doc2Vec

3 We apply doc2vec on the processed data to generate the similarities between the given documents

Constraint Generation

4 Using the similarities obtained from the Doc2Vec model, we generate the must-link, and cannot-link constraints depending on the threshold values

Perform PCK-Means

5 Using the constraints generated for the previous step, we apply the PCK-means algorithm to determine the clusters

Explainability

6 **Method 1:** Using the similarity measures obtained from word2vec, and the key phrases associated with each documents, we are able explain why a particular document belongs to a cluster, with it's closest neighbors in that cluster.

Method 2: We are training a decision tree classifier which gives us cluster labels as leaf nodes. We have extracted the most representative tokens per cluster which has led to the decision of the clustering.

Figure2: Implementation steps

OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG FACULTY OF COMPUTER SCIENCE

Constrained Clustering of Text based on TREC 4 5 with textual explanations

Adrija Bhavana Felix Nikhil Sourima

Otto von Guericke University Magdeburg

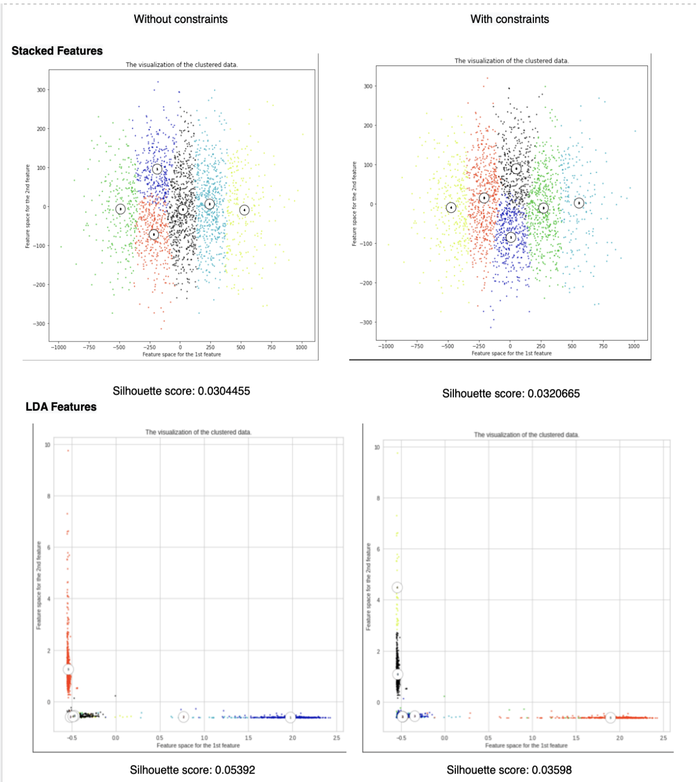
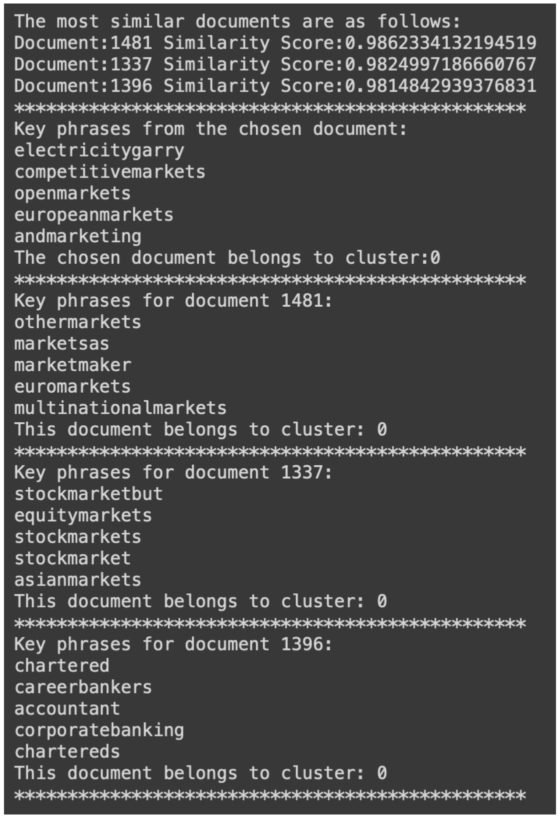


Figure3: Clustering with features obtained from stacked features(above), and LDA(below)

Results and Evaluation

- The goodness of the cluster is evaluated by the Silhouette score which tells us how well the clusters are well apart from each other.
- We can see from the resulting graphs(Figure 3) and evaluation techniques that there are improvements in the Silhouette scores after applying the constraints compared to no constraints at all.
- In the case of feature extraction, we have performed the stacking operation where we combine feature vectors obtained from different feature extraction techniques such as Tf-Idf, Word2Vec, and LDA. We can see that the clustering performance improves in the case of the stacked features than the standalone features with respect to the Silhouette score.
- In terms of explainability, given a document, we provide the most similar documents along with the similarity scores to that document. We also portray the key phrases and the cluster labels of the query document and the obtained similar documents which acts as a justification as to why a document is clustered the way it is(Figure 4)



```
The most similar documents are as follows:
Document:1481 Similarity Score:0.9862334132194519
Document:1337 Similarity Score:0.9824997186660767
Document:1396 Similarity Score:0.9814842939376831
************************************************
Key phrases from the chosen document:
electricitygarry
competitivemarkets
openmarkets
europeanmarkets
andmarketing
The chosen document belongs to cluster:0
************************************************
Key phrases for document 1481:
othermarkets
marketsas
marketmaker
euromarkets
multinationalmarkets
This document belongs to cluster: 0
************************************************
Key phrases for document 1337:
stockmarketbut
equitymarkets
stockmarkets
stockmarket
asianmarkets
This document belongs to cluster: 0
************************************************
Key phrases for document 1396:
chartered
careerbankers
accountant
corporatebanking
chartereds
This document belongs to cluster: 0
************************************************
```

Figure4: Explainability using Doc2Vec, and keyphrasification

Conclusion

In this project, we have performed clustering of textual data by introducing constraints to it. Since we are working with textual data instead of hard constraints like COPKmeans we are allowing some constraint violation among the clusters using PCKmeans.In order to implement cluster explainability, we are comparing documents of a cluster based on the similarity using Doc2Vec.Also we extracted key-phrases of each document of a cluster to get the aspect of that cluster.

Learning and challenges:-We got a good experience working with an unsupervised learning task with a large amount of text data.We also learnt about different models and different algorithms. The overall learning experience is pretty good though we faced few major issues while working-

- Crashing of runtime due to huge feature size
- Selecting the feature size

Future Improvements:- The work can be further improved by handling the large number of features in a more efficient manner especially by solving the runtime/memory issue that we faced.

References

1. https://github.com/datamole-ai/active-semi-supervised-clustering
2. https://towardsdatascience.com/unsupervised-nlp-topic-models-as-a-supervised-learning-input-cf8ee9e5cf28
3. [1405.4053] Distributed Representations of Sentences and Documents (arxiv.org)
4. https://keyphrasification.github.io/slides/part2.1-dl-methods-for-keyphrase-extraction.pdf