OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG FACULTY OF COMPUTER SCIENCE

Semester Project for Advanced Topics in Machine Learning

Topic:

Constrained Clustering ofTextBased on Newsgroups with textual (or visual) explanations

Strategy/ Plans of Execution:-

Please find the detailed instructions of the project below

1) Dataset: TREC disks 4 and 5 dataset containing news articles

2) Document Pre-processing: For preprocessing our data we plan to proceed with the following steps-

   - Lower casing to provide uniformity in the text.
     - lower () of python
   - Convert sentences into tokens by sentence tokenization or work tokenization.
     - NLTK library and word_tokenize()
   - Remove the punctuation marks and other non-letter characters from our list of words.
     - regex
   - Eliminate a proportion of stop words from our vocabulary
     - stopwords module from NLTK
   - Lemmatization
     - WordNet Lemmatizer from NLTK

3) Feature Extraction: We will implement three mandatory and one additional feature extraction algorithm for our dataset as mentioned below-

Mandatory features:-

1) TF-IDF:- We will use the TfidfVectorizer from sklearn to get the frequency of words and their relevance in the corpus. This will generate a feature matrix.

2) Word embedding- We will map our words into vectors of numbers and for that we will use Word2vec and the gensim package.

3) Key-Phrases :- We will use keyphrasification that summarizes texts with phrases. We will use PKE, an open source python-based keyphrase extraction toolkit to solve the purpose.

Additional Feature:-

1. We will use LDA (Latent Dirichlet Allocation) for topic modeling by using gensim and sklearn.This is a good choice because maximizing the distance between the means of each class when projecting the data in a lower-dimensional space can lead to better classification results (thanks to the reduced overlap between the different classes).

4) Feature Scaling and Selection: After extracting all the features, we will next find out the relevant features for our dataset. For feature selection-

- We will implement a chi-square test using selectkbest of sklearn.
- Another alternative can be Mutual Information using mutual_info_classif from sklearn.
- As a further advancement we can implement an iterative feature selection process. The idea behind the iterative method is- we will remove the terms whose document frequency is lower than 3 before clustering. Then at each iteration, about 10% terms (or 3% terms when less than 10% terms left) with lowest ranking score outputted by IG or CHI are removed.[6]

5) Methodology for Clustering: Methods, Hyperparameters, Constraints :

- **Generation of Constraints-** As discussed in the Model Training section below, we will use PCK-Means for clustering, a constrained clustering method. For this we will need to find must-link- or cannot-link-constraints. We will find them using the following procedure, which is taken from [3]:

  Choosing constraints:
  - Extract the n-grams of the documents to cluster
  - Discard the n-grams containing at least one stopword
  - Create a must-link constraint between all pairs of documents that share at least t n-grams

  This method relies on two hyperparameters: n - the length of the n-grams, and t - the minimum number of shared n-grams to introduce a constraint. If we choose n and t to be too small, then too many must-link-constraints will be introduced and we will likely only get one cluster (or just a few). If we choose n and t to be too large, then barely any constraints will be introduced, which will defeat the purpose of constraint clustering. Therefore, our method of finding suitable hyperparameters n,t will be the following:
  Step 1: Pick values for n,t
  Step 2: Produce a clustering for the given hyperparameters
  Step 3: Use the metrics discussed in Section V to evaluate the degree of separation of the different clusters and intra-cluster similarity. If intra-cluster similarity is too low, we increase the values of n and t to reduce the number of constraints, if the degree of separation of the different clusters is too low, we reduce the values of n and t to increase the number of constraints.

- **Model Training-** The constrained clustering algorithm we have chosen is PCKMeans. We do not choose COP-k-Means, since its clustering is dependent on the ordering of the data.
  We also don't choose MPCK-Means, since our data is very high-dimensional and computing/training the matrix (and its determinant) would therefore be very expensive.

As discussed in [1], hierarchical clustering techniques become computationally infeasible for large collections in high-dimensional space, and therefore they are not a viable technique for our task.

PCK-Means - The algorithm allows constraint violations to some extent.Since we are using n-grams technique to identify constraints, it could happen that we introduce a must-link constraint between two documents on different topics, simply because they share a lot of n-grams.The algorithm is as follows [7]-

1. Initialize cluster centers **using constraints**
2. Repeat until convergence
   1. Given k cluster centers:
      Assign each instance to the optimal cluster according to the **new objective function**
   2. Given the instances of k clusters:
      Compute the clusters centers

The cluster centroids are initialized based on neighborhood sets and also in the objective function apart from the classic k means objective part we add a penalty terms for constraint violation for both mustlink and cannot link constraints as described by the below figure-

$$c^* = \arg\min_c \left(\frac{1}{2}\|x_i - \mu_c\|^2 + \sum_{(x_i,x_j)\in con_=} w_{ij}\delta(c \neq c_j) + \sum_{(x_i,x_j)\in con_\neq} \overline{w}_{ij}\delta(c = c_j) \right)$$

- **Hyperparameter-**PCK-Means mainly relies on the hyperparameters k (number of clusters) and the w_ij (penalty weights). We select the number of k using the Elbow Method. We will set the values w_ij equal to each other, as we **assume** (for simplicity) that two documents sharing n-grams is as expressive as any other two documents sharing any other n-grams.

6) Explainability of Clustering: For explaining a text clustering we intend to compute the word frequency distribution for each cluster and use the most frequent words in a cluster to represent the cluster's semantics as proposed by [4]. However this comes with the problem that high-frequency words are often common among several clusters. To overcome this problem we will adjust every word's weight for each cluster adaptively as described in [4]. The algorithm is as follows [4]-

Input: $Corpus$ is the corpus, and $clusterResult$ is the pseudolabel list.
Output: $indWordsList$ contains the list of indication words for every cluster.

Let $n = size(Corpus)$.
Let $featList = []$.
{Map the text into 0-1 bag-of-words features.}
for $i \in [1, n]$ **do**
    Split the text $Corpus[i]$ into tokens.
    Filter out stop words and low-frequency words.
    Map the remaining tokens into 0-1 feature vector $feat$.
    Append the feature vector $feat$ into $featList$.
end for
{Obtain Indication Words for Every Cluster.}
Train the logistic regression classifier $lgClassifier$ on training data $(featList, clusterResult)$.
{$weightList$ is the weight list whose length is the number of labels.}
Let $weightList = lgClassifier.weights$
Let $indWordsList = []$
for $weight \in weightList$ **do**
    Let $tmpWeight = []$
    for $w \in weight$ **do**
        Append $abs(w)$ into $tmpWeight$
    end for
    Let $tmpIdx = argsort(tmpWeight)[: n]$
    Map $tmpIdx$ into indication words $indWords$
    Append $indWords$ into $indWordsList$
end for
return $indWordsList$

Additionally, we build off of the ideas in [2]:

- train a decision tree: We use the IMM-method (see [5]) to try to find a simple decision tree to explain the cluster assignment.
- design a 2-layer neural network that produces the clustering

7) Model Evaluation: We aim for a clustering for which (a) intra-cluster-similarity is high and (b) different clusters are well-separated. For this we will use the following metrics, as discussed in [1]:

- *Silhouette index:* The Silhouette index measures how much structure is present in a given clustering. A Silhouette index close to 1 will indicate that intra-cluster-similarity is high.
- *Davies-Bouldin index:* The Davies-Bouldin index is indirectly proportional to the separation of the clusters. If different clusters tend to be further apart, we will get a lower DB index.

8) Reference:

[1] Meng Yuan, Justin Zobel, Pauline Lin: Measurement of clustering effectiveness for document collections

[2] Grégoire Montavon, Jacob Kauffmann , Wojciech Samek, Klaus-Robert Müller: Explaining the Predictions of Unsupervised Learning Models

[3] M. Eduardo Ares and Alvaro Barreiro: Constrained Text Clustering Using Word Trigrams

[4] Renchu Guan, Member, IEEE, Hao Zhang, Yanchun Liang, Fausto Giunchiglia, Lan Huang, and Xiaoyue Feng: Deep Feature-Based Text Clustering and Its Explanation

[5] https://ucsdml.github.io/jekyll/update/2020/10/16/explain_k_means.html

[6] Liu, Tao, Shengping Liu, Zheng Chen, and Wei-Ying Ma. "An evaluation on feature selection for text clustering." In Proceedings of the 20th international conference on machine learning (ICML-03), pp. 488-495. 2003.
[7]https://elearning.ovgu.de/pluginfile.php/669410/mod_resource/content/2/english_ATiML05_ConstrainedClustering.pdf