Datamining project

Louna Ansari & Eero Asikainen
377652 595654

December 2021

1 Methods

In this project we conduct 9 fold of experiment to apply three different clustering methods along to three different represenatations of the data set. We compare these 6 folds of experiment and consider the word frequencies in the results section. We compare the accuracy of the clusters based on the nmi score. In the appendix the plot for all the 9 fold of experiments are presented.

1.1 Preprocessing

Each row in the data was first transformed into a list of token words. After that, the following steps were performed:

- **Stopword removal:** Stopwords were removed using the nltk `stopwords` english wordlist into which the punctuation symbols from `string.punctuation` were added. In addition all words that are or contain numbers were removed.
- **Stemming:** To stem the tokens, the `SnowballStemmer` from nltk was used with the english language parameter.
- **Tf-idf transformation:** Transforming the data into the *term-frequency times inverse document-frequency* representation was done using sklearn's `TfidfVectorizer` with default options. The sklearn's implementation with default options is

$$idf(t) = log\frac{1+n}{1+df(t)} + 1$$

where n is the number of documents and $df(t)$ the number of documents in the set that contain the term t. These idf vectors are then normalized by dividing them with their euclidean norm.

1.2 Description of clustering methods

In this section we briefly report what parameters were used for each of the 3 clustering functions. All functions were from the `sklearn` library.

As our first method we used K-means clustering. Since we have been given the true classes of the documents, we know to use the value 5 as the number of clusters. Other than that

we used default options. Whereas K-means devides the objects into K clusters such that metrics relative to the centroids of the clsuters in minmized, in spectral clustering the data points as nodes of connected graph and clusters are found by partitioning graph and based on their spectral decomposition into subgraphs.

For our second method we used spectral clustering. The sklearn implementation uses K-means as the method to cluster the affinity graph that is first constructed, so here again the number of clusters was set to be 5. The affinity method was set to nearest neighbors, and the number of neighbors was 200. This means that when constructing the affinity graph, each node will be connected to 200 of its nearest neighbor nodes using the euclidean distance as a similarity measure.

The third clustering method we used was agglomerative hierarchical clustering. Again, the cluster number was set to 5, and as a linkage metric we used the ward linkage. Hierarchical clustering is a set of nested clusters which are arranged as a tree.

1.3 Feature selection and dimensionality reduction

One approach was applied to assess the usefulness of each input feature individually, selecting a subset of most useful features. This method applied is a generic variable ranking procedure where a single score is calculated for every input variable by applying a scoring criterion where the variables are sorted in descending order of these scores. Feature importance provides a score for each feature of the data, the higher the score the more important or relevant is the feature towards the output variable.

Here a subset of the most highly ranking variables can be chosen for example by considering the top k variables or all variables exceeding a score threshold (k or generally decided by the user). This method is also referred to as the "filtering" approach, due to the non-selected variables begin "filtered out". More specifically we applied selectKBest from sklearn.feature.selection and applied the mutual information regression to extract the most important features, we used k= 1000 in this case to choose the most important 1000 features out of the 9076 features.

The other feature selection approach we used was principal component analysis, in order to the dimensions of the data by transforming it into a space where variance is maximized. It was found that using 10 principal components seemed to provide slightly better or similar results as the whole data.

2 Results and Conclusions

As illustrated in the previous section, we represented the data in 3 different ways: original plane tf-idf vectros as they had been extracted, the K best features chosen using the sklearn library and the PCA representation of the data. To validate and compare results from different approaches of clustering and preprocessing we used the "Normalized Mutual Information" score from sklearn's `normalized_mutual_info_score` with the average method set to geometric. This equals the following equation:

$$NMI(C, D) = \frac{I(C, D)}{\sqrt{H(C)H(D)}}$$

where given true class C and predicted class D, the nominator $I(C, D)$ is the mutual information and the denominator is the square of the entropy H for both classes multiplied.

This validation score is an external score, since we are using the true class labels in it. Below are the results for each of the three different clustering methods with three variations of the data. Plain refers to just the tf-idf vectors of the data. K best refers to the feature selection done with a regression classifier taking the 1000 highest scoring features and PCA is the data after dimension reduction to 10 components with pca. The Normalized Mutual Information metric is presented in the below table as:

	K-means	Spectral	Hierarchical
Plain data	0.739	0.781	0.443
K best selected features	0.251	0.228	0.257
PCA based features	0.657	0.794	0.553

In addition a visual presentation of the clusterings is shown in figure 1 in the appendix, where the data has been scaled down to 2 pca components for visualisation purposes.

We can see that the best performing cluster method was spectral clustering after the data was reduced with pca, although the difference to the unreduced score is small. Here are the most common 5 words and their frequencies for all of the 5 clusters that were found using that method:

Class	Words and frequency
1	databas(677) data(581) relat(358) system(305) queri(272)
2	compil(582) program(369) use(336) comput(268) code(252)
3	robot(774) control(328) use(306) system(294) model(172)
4	use(704) imag(641) method(560) detect(483) propos(465)
5	secur(692) use(410) propos(384) scheme(340) data(328)

These words already give quite a good idea of the general topics of the clusters. We looked also at the 20 most common words, which are not all listed here, but after that here is our best guess about the topics of the clusters:

- Databases
- Algorithms and optimization
- Robotics and systems
- Computer vision
- Security engineering

We found differences between clustering methods and different feature representations. The best performance comes from combination of spectral clustering with PCA based features with NMI of 0.794.

Appendix

Libraries used and their versions:

pandas	1.1.5
numpy	1.19.5
nltk	3.2.5
scipy	1.4.1
scikit-learn	1.0.1
sns	0.11.2

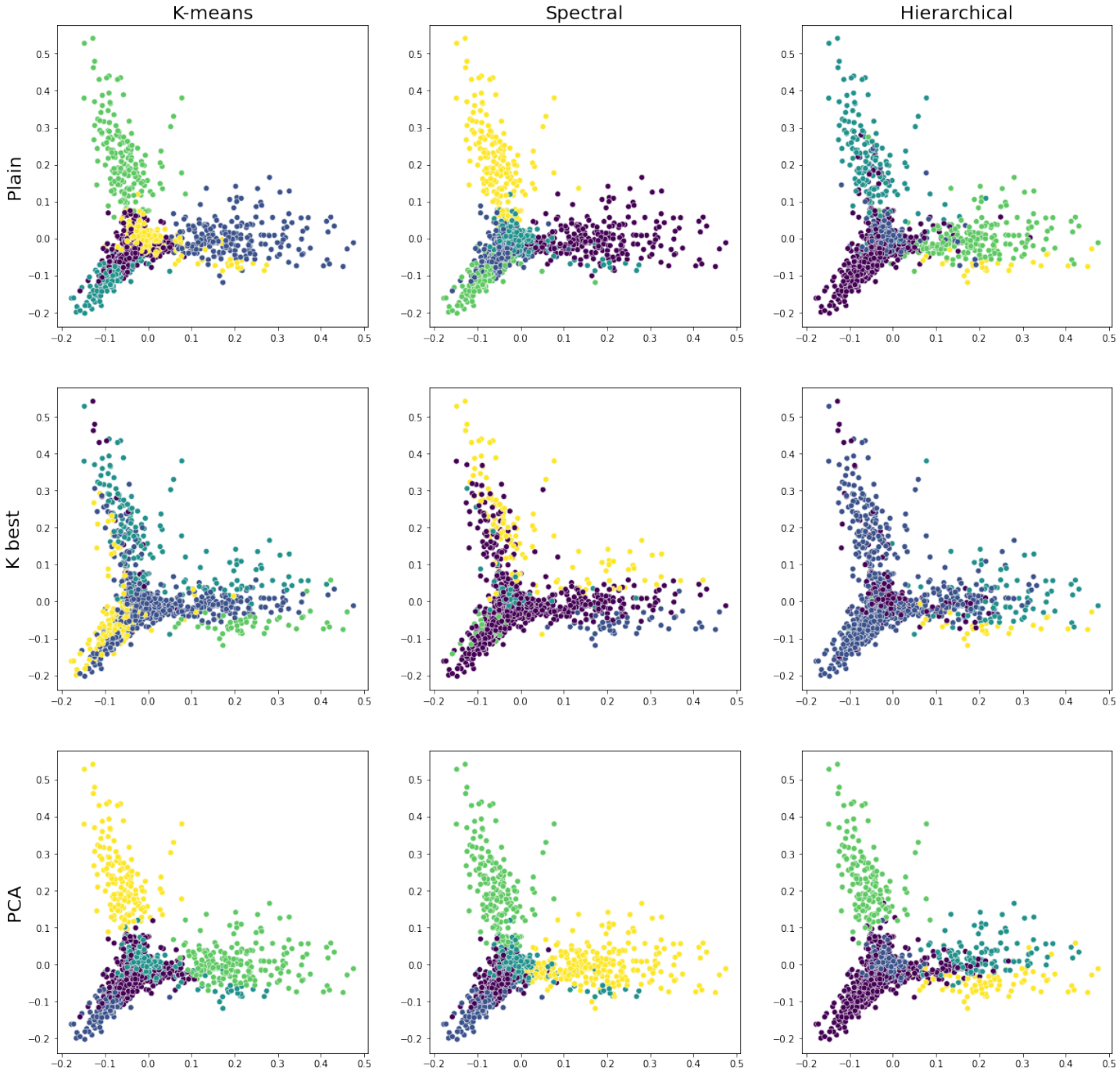

Figure 1: First 2 pca components plotted for each clustering