NPFL097 Assignment 3
Word Clustering and Component Analysis

David Mareček

December 2021

In this assignment, you will get a vocabulary of English words with corresponding 512-dimensional vector representations. These word vectors are used for predicting the next English word in a generated English sentence and were obtained form the Czech-English Neural Machine Translation model.

Your task is to analyze this highly-dimensional vector space and find the main features by which the words are distributed. The preferred programming language is Python.

Send me your final source code and commented graphs and results by email: `marecek@ufal.mff.cuni.cz`.

Data

Download the file `nmt-en-dec-512.txt` containing word vectors of the 5,000 most frequent words from English fiction. There is one word on each line followed by its vector representation (512 tab-separated real numbers).

Tasks

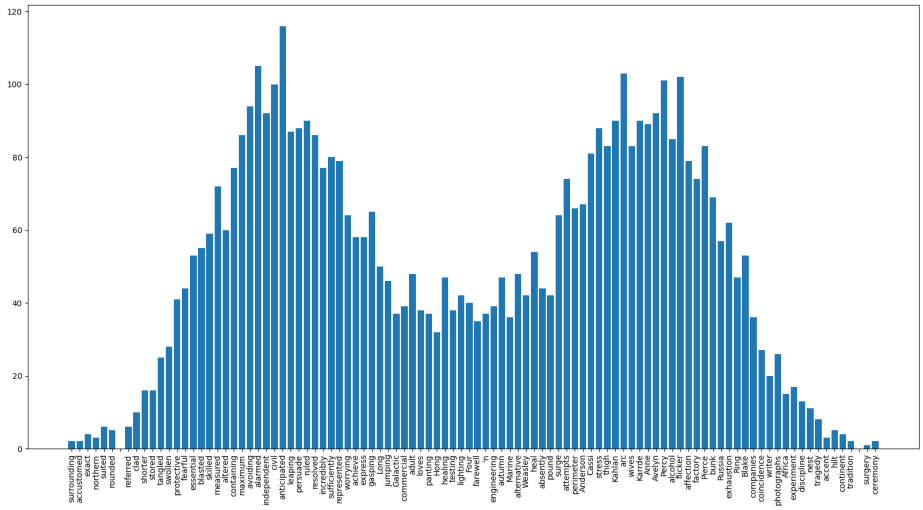
1. Visualize the word vectors in 2D using the T-SNE method. Use the `sklearn.manifold.TSNE` function. Learn how T-SNE works and what parameters it has. Draw a scatterplot with a reasonable amount of labels, so that it is readable and you can identify different clusters of words. *(1 pt)*

2. Cluster the word vectors in the original 512-dimensional vector space using the methods of K-means, GaussianMixture, and Aglomerative clustering. Use the functions implemented in the `sklearn` library. Try different number of clusters (3,5,10) and different linkage strategies (ward, single, complete). Visualize the clusters using different colors in T-SNE. Compute the Silhouette coeficients of the individual clusterings. Use

`sklearn.metrics.silhouette_score`. What method is the best? What methods failed? *(2 pts)*

3. Perform the Principal Component Analysis on the data (use the function `sklearn.decomposition.PCA`). Visualise the data transformed into the first two principal components. Try to guess what features of words are represented by the first two components. Run the K-Means, Gaussian-Mixture, and Aglomerative clustering on the data transformed into the first two PCA componens. Compute the Silhouette coeficients and find the best number of clusters. Why the silhouette coeficients are so different from those in Task 2? *(3 pts)*

4. Visualise the first 5 PCA components. Divide each component direction into 100 intervals (bins) and plot 5 bar-charts showing the distribution of words along each component. Label the bars in your bar-chart by words sampled from each bin. Try to explain what features of words are represented by individual PCA components. *(2 pts)*



5. Compute the Independent Component Analysis and obtain 50 independent components. Use "`sklearn.decomposition.FastICA`" with the parameter `n_components=50` to perform transformation into first 50 PCA components as a pre-processing step. Visualize 5 chosen components in the same way as in the Task 4. Are they non-Gaussian? What features of words they represent? Hint: Look at the long tails of the distributions. *(2 pts)*