Data Collection & Management Lab

Homework 1

Snir Lugassy (206312506)

Abstract

Given an extracted text about a company, we need to select up to 10 words which captures the most **information** about the company's industry. In order to select words from a given text, we calculated a score for each word and chose the words with the highest score. We trained a MLP to predict the score given a word embedding vector.

Data Processing and Cleaning

The following were applied on each text:

- Remove special characters and punctuation characters
- Remove "Stop words" – according to *NLTK*'s list of general and common words
- Remove words with length $n \leq 2$
- Convert text to lowercase
- Tokenize – convert the text to a list of tokens (words)
- Word embedding – using pre-trained *word2vec*, convert each token to a vector in $\mathbb{R}^{300}$

Initially we filtered out HTML tags using "*Beautiful Soup*" library, but it had great impact on performance, therefore it was removed from the pipeline. In the word embedding process most HTML tags will be ignored (most tags are OOV - Out of Vocabulary).

Domain Specific Word Score

We assigned a score to each word, which aims to capture how much a word is domain (industry) specific (high score), or general and commonly used (low score).
Using the distribution of the word appearance in each domain:

$$d_w = (f_1, \dots, f_n)$$

Where f_i is the frequency of the word w in the domain i.
The score:

$$score(w) = \max_i d_{w,i}$$

Practically, we used a threshold over the frequency in order to moderate long-tail appearance of a word in different industries.

Score Prediction and Inference

We trained a Multi-Layer Perceptron (MLP) to predict the score of a word (word vector).

$$MLP(v_w) = score(w)$$

$$\mathbb{R}^{300} \to [0,1]$$

Given unseen text, we normalized it using the previously mentioned pipeline, predicted the score of each word and chose the 10 highest words considering the predicted score.