UnibucKernel: A kernel-based learning method for complex word identification

Andrei M. Butnaru, Radu Tudor Ionescu
Department of Computer Science, University of Bucharest, Romania

Paper:

Paper:

Introduction

- We present a kernel-based learning approach for the 2018 Complex Word Identification (CWI) Shared Task
- Our system is based on **(*i*)** extracting lexical, syntactic, semantic and character-level features and on **(*ii*)** training a kernel method for the classification and regression tasks
- We participated in the English monolingual track only
- Our best result during the competition was the third place on the English Wikipedia data set, but we also reported better post-competition results

Feature Extraction

- Character-level features:
  - the number of characters, vowels and constants
  - the percentage of vowels and constants from the total number of characters in the word
  - the number of consecutively repeating characters, e.g. double consonants
  - character n-grams of 1, 2, 3 and 4 characters in length
- Syntactic features:
  - the part-of-speech recorded as a one-hot vector
- Lexical and semantic features:
  - the number of senses listed in WordNet
  - the minimum, the maximum and the mean value of the cosine similarity between the target word embedding and each other word embedding from the sentence
  - the minimum, the maximum and the mean value of the cosine similarity between each sense embedding of the target word and each sense embedding computed for each other word in the sentence; to compute the sense embedding for a word sense, we first build a *disambiguation vocabulary* by including the corresponding WordNet synset words, gloss (examples included) and words found in the glosses of semantically related synsets; we then embed the collected words in an embedding space and compute the median of the resulted word vectors, as described in [Butnaru et al., 2017]
  - one-hot vectors that encode spatial bins applied over PCA-reduced word embeddings; inspired by the spatial pyramid used in computer vision [Lazebnik et al., 2006], we apply a grid to divide the 2D embedding space (obtained after applying PCA) into multiple and equal regions, named bins; we use multiple grid sizes starting from coarse divisions such as $\mathbf{2} \times \mathbf{2}$, $\mathbf{4} \times \mathbf{4}$, and $\mathbf{8} \times \mathbf{8}$, to fine divisions such as $\mathbf{16} \times \mathbf{16}$ and $\mathbf{32} \times \mathbf{32}$; we index the bins and encode the index of the bin that contains the target word as a one-hot vector (as shown in the figure below)

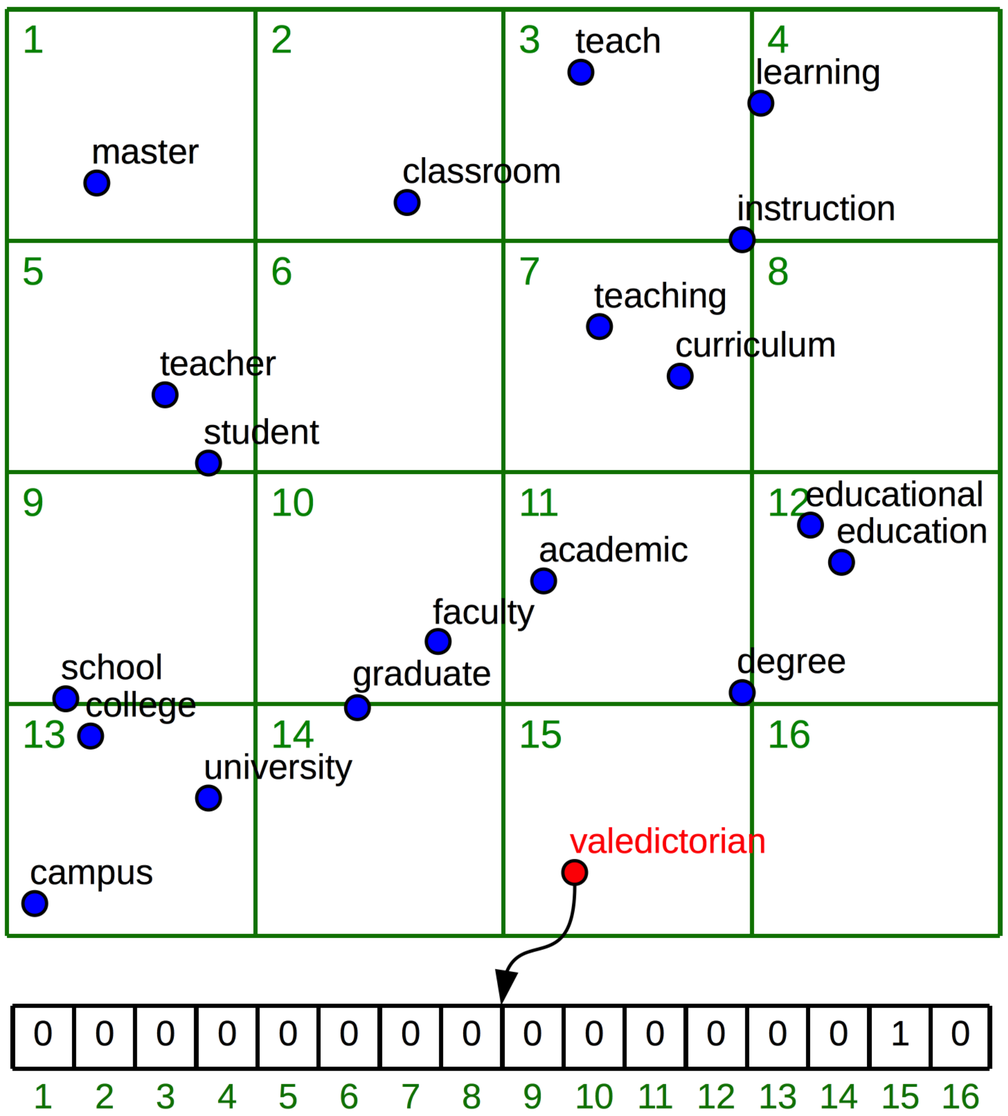

Figure: A set of word vectors represented in a 2D space generated by applying PCA on 300-dimensional word embeddings. A grid of $\mathbf{4} \times \mathbf{4}$ is applied on the 2D embedding space. For example, the word "valedictorian" is located in bin number **15**. Consequently, the corresponding one-hot vector contains a non-zero value at index **15**.

Kernel Representation

- We experiment with two commonly-used kernel functions, namely the linear kernel and the Radial Basis Function (RBF) kernel
- The *linear kernel* is easily obtained by computing the inner product of two feature vectors $\boldsymbol{x}$ and $\boldsymbol{z}$:

$$\boldsymbol{k}(\boldsymbol{x}, \boldsymbol{z}) = \langle \boldsymbol{x}, \boldsymbol{z} \rangle$$

- In a similar manner, the *RBF kernel* (also known as the Gaussian kernel) between two feature vectors $\boldsymbol{x}$ and $\boldsymbol{z}$ can be computed as follows:

$$\boldsymbol{k}(\boldsymbol{x}, \boldsymbol{z}) = \boldsymbol{exp}\left(-\frac{\mathbf{1} - \langle \boldsymbol{x}, \boldsymbol{z} \rangle}{\mathbf{2}\sigma^{\mathbf{2}}}\right)$$

- Because the range of the raw data stored in the feature vectors can have significant variation, we normalize the kernel matrices before the learning phase:

$$\hat{\boldsymbol{K}}_{\boldsymbol{ij}} = \frac{\boldsymbol{K}_{\boldsymbol{ij}}}{\sqrt{\boldsymbol{K}_{\boldsymbol{ii}} \cdot \boldsymbol{K}_{\boldsymbol{jj}}}}$$

Learning Methods

- We use the Support Vector Machines (SVM) classifier for the binary classification of words into simple versus complex classes
- We employ ν-Support Vector Regression (ν-SVR) in order to predict the complexity level of a word (a word is more complex if it is labeled as complex by more annotators)

Data Sets

Data Set	Train	Validation	Test
English News	**14002**	**1764**	**2095**
English WikiNews	**7746**	**870**	**1287**
English Wikipedia	**5551**	**694**	**870**

Table: A summary with the number of samples in each data set of the English monolingual track of the 2018 CWI Shared Task.

Classification Results

Data Set	Kernel	Accuracy	F_1-score		Rank (Post-Competition)
English News	linear	**0.8653**	**0.8547**	**0.8111***	**12** (**6**)
English News	RBF	**0.8678**	**0.8594**	**0.8178***	**10** (**5**)
English WikiNews	linear	**0.8205**	**0.8151**	**0.7786***	**10** (**5**)
English WikiNews	RBF	**0.8252**	**0.8201**	**0.8127***	**5** (**4**)
English Wikipedia	linear	**0.7874**	**0.7873**	**0.7804***	**6** (**4**)
English Wikipedia	RBF	**0.7920***	**0.7919***	**0.7919***	**3** (**3**)

Table: Classification results on the three data sets of the English monolingual track of the 2018 CWI Shared Task. The methods are evaluated in terms of the classification accuracy and the F_1-score. The results marked with an asterisk are obtained during the competition. The other results are obtained after the competition.

Regression Results

Data Set	Kernel	Mean Absolute Error	Post-Competition Rank
English News	linear	**0.0573**	**4**
English News	RBF	**0.0492**	**1**
English WikiNews	linear	**0.0724**	**4**
English WikiNews	RBF	**0.0667**	**1**
English Wikipedia	linear	**0.0846**	**4**
English Wikipedia	RBF	**0.0805**	**2**

Table: Regression results on the three data sets of the English monolingual track of the 2018 CWI Shared Task. The methods are evaluated in terms of the mean absolute error (MAE). The reported results are obtained after the competition.

Future Work

- In future work, we believe that joining the training sets provided in the English News, the English WikiNews and the English Wikipedia data sets into a single and larger training set can provide better performance
- Another direction that could be explored in future work is the addition of more features, as our current feature set is definitely far from being exhaustive

butnaruandreimadalin@gmail.com

raducu.ionescu@gmail.com