Project proposals

Sentence compressor

Given the dataset published in Sentence Compression by Deletion with LSTMs develop a model for compressing sentences.
I can provide some Python starter code to read the dataset and train a baseline.

UIMA for Python

The project goal is to implement:

- a simple server that accepts a document, runs an annotation pipeline and returns JSON output;
- a Python class that parses the JSON output and implement the main functionalities of the UIMA CAS.

Bonus points: implementing an high throughput server.

Tree structures for the classification of questions

The goal of the project is to reimplement the tree structures described in this paper and evaluate them in a question classification setting.

Question Classification using unsupervised data

The project goal is to use the 30 M Factoid Question dataset to produce training data for different question categories: ABBR, DESC, ENTY, HUM, LOCATION, NUM.

Simple rules and patterns can be used to filter the questions given the Freebase categories. The noisy labelled data will be used to train a simple ngram classifier.

Assigning word embeddings to OOV words

Deep learning systems use word embeddings learned on huge corpora as input. If a word is unseen you do not have a vector for it, or you can map the word to a special symbol for rare word. Nevertheless, embeddings trained by others on more comprehensive corpora (e.g. Google News pretrained word embeddings) may contain the vector for your word. The project will consist in mapping a word A (for which you do not have a vector) from a word embedding set X, to another word B from another word embedding set Y, such that A is in Y, B is similar to A and B is in X. In this way we can assign the vector of B from X to A and evaluate the effect of the process on a classification task.

Evalita 2016 tasks

Select one of the task of Evalita 2016 and build a model for it.

DKPRO in Jython

The project will consist in setting up and analyzing a simple dataset using the Python bindings of DKPro. See: https://dkpro.github.io/dkpro-core/pages/jython-intro/

UIMA annotators for the OpeNER Project

Info: http://www.opener-project.eu/
The OpeNER Project contains many natural language processors for different languages. The goal of this project is to learn how to configure and launch these processors, and wrap them into UIMA annotators. The UIMA annotators (one annotator for each task) should cover at least the following tasks:

- language detection
- tokenization
- part of speech tagging OR named entity recognition

In addition, the annotators will have a configuration parameter for specifying the language of the analyzed text, in order to use the appropriate OpeNER models.
The DKPro type system must be used for building the annotators.