

“Question Classifier” Project - Report

APPROACH: MLClassification

MODEL DESCRIPTION

The evaluation metric we tested was Accuracy Classification Score (intrinsic evaluation) with the help of scikit-learn: we used it to check if our set of predicted labels matched exactly with the corresponding set of labels from the training set, returning the percentage of how many did we got right.

The preprocessing starts by removing the quotes from each question in the training and development set, and all the numbers and other punctuation marks from the same sets, since they are not relevant for classifying the questions. Then we applied Tokenization to all the questions, transforming every sentence into a sequence of tokens.

The next steps were actions made on the tokens. For each question divided in tokens, we lowercased every token (i.e. “When” and “when” have the same meaning lowercase or not, so it is not relevant to have this two tokens as different ones), then performed the Porter Stemmer Algorithm to reduce derived words to their root form, and finally applied Lemmatization to group together the inflected forms of a word so they can be analyzed as a single word, reducing repetitions of words and their derived forms.

We did not remove the stop words because some of the English stop words from NLTK library were not relevant for our code, and after removing them before lowercasing we noticed that our accuracy dropped down, so we decided not to remove them.

ACCURACY

The accuracy resulting from evaluating our models in the development set is 83% in the coarse model, and 76% in the fine model.

SHORT ERROR ANALYSIS

Due to the existence of similar labels, such as *LOC:other*, *ENTY:other* and *NUM:other*, there will always be an unpredictability factor. This factor is also associated with the fact that the dataset has errors, as studied in Natural Languages classes. As a result, there is always the possibility of errors in the prediction, which contributes to never reaching 100% accuracy.