Translation Classification

Ziwei Gu (zg48@cornell.edu)

Github repo: https://github.com/ZiweiGu/translation-classifier

Summary

In this project, I built an SVM classifier that tells whether a translation was created by a human or by a machine. I extracted a variety of linguistic features from the candidate translation, along with a similarity score between candidate and human translations, to train the SVM model. The binary classifier performed well on the testing set, achieving an average F1 score of **0.713**.

Approach

Since the specific type of machine translation system used in the given dataset was unknown, I looked for general linguistic features that could capture the **quality** of the translated sentences. The features I used are summarized in the table below.

Feature	**Explanation**
tree_depth	The depth of the parse tree, normalized by sentence length
func_density	The density of function words (including determiners, prepositions, conjunctions, and auxiliary verbs)
pron_density	The density of pronouns
similarity	The Jaccard similarity between human and candidate translation
bleu	The same bleu score in the original dataset

I used **spaCy**'s dependency parser and part-of-speech tagger to find the parse tree and token types. I chose to include density of pronouns because coreference resolution is where machine translation systems often make mistakes. **Support vector machine (SVM) with a linear kernel** performed best on a validation set (20% of training set) among all the models I tried.

Evaluation

The SVM model achieved an average F1 score of 0.7129 on the test set, with a precision score of 0.7139 and a recall score of 0.7126. As for a bit of qualitative analysis, I looked at examples that the model got wrong, which often involved a lot of paraphrases between the candidate and human translations and low Jaccard similarity values. In those cases, a different similarity metric based on word embeddings might work better.