

The quality of machine translation systems is heavily dependent on the parallel corpora on which it is trained. For many language pairs, the amount of parallel sentences readily available is low. Wikipedia is a potential source for these sentence pairs since many of its articles are written in multiple languages. The goal of the project is, given a set of paired Wikipedia articles where each pair has one english article and one foreign language article, to create a set of parallel sentences in that language pair. This would be simple if the article pairs were exact translations of each other, but this is almost never the case (i.e., the two articles are usually written independently by speakers of each language). However, because the articles discuss the same topics, they should have some sentences which express the same meaning. Given two articles that are not direct translations of each other, we would like to grab as many parallel sentences from the articles in order to create a new parallel corpora.

Getting Started

The quality of the system that you develop will be based on the following metrics:

Let A be the set of parallel sentences produced by the system and B be a human-produced set of parallel sentences from the set of articles on which the system was run.

$$\text{Precision: } \frac{|A \cap B|}{|A|}$$

$$\text{Recall: } \frac{|A \cap B|}{|B|}$$

The final score you see is an F_1 score, which is defined as the following:

$$F_1 = 2 * \frac{\text{Precision} + \text{Recall}}{\text{Precision} * \text{Recall}}$$

As your system improves, your F_1 score should decrease. The default system is implemented by pairing up sentences in the articles in order. This is probably not a good system because it basically relies on one page being a direct translation of the other - something that does occur on Wikipedia, but definitely not an assumption we should make in general. To score the default, run the following command:

```
$ python default_pair.py | python grade
```

The Challenge

For the baseline, create features for pairs of sentences based on the provided one-gram and two-gram translation models. The features used in our baseline include:

- Absolute value of the difference in sentence length
- The sum of the translation probabilities of spanish one-grams matched to english one-grams all over the length of the spanish sentence
- The sum of the translation probabilities of spanish one-grams matched to english one-grams all over the length of the english sentence
- The sum of the translation probabilities of spanish two-grams matched to english one-grams all over the length of the spanish sentence
- The sum of the translation probabilities of spanish two-grams matched to english one-grams all over the length of the english sentence

The data is not formatted with these features. You are only given the text of the spanish and english articles, so you will have to implement the feature creation yourself. When that is done, train a binary classifier to classify a pair of sentences as parallel or not parallel, given the pair's features. You can either implement your own binary classifier or have a look at some of the classifiers provided by [mlpy](#).

This should significantly increase your score, but there is much more that can be done. For instance:

- Try implementing the other features mentioned in [this paper](#) especially those pertaining directly to Wikipedia. Use [this](#) library for parsing the Wikipedia articles themselves - the names of the articles used in the grading set can be found in `data/titles_esn` and `data/titles_enu`.
 - Helpful hint: the grader does not take image captions into account, so don't bother implementing that feature. Stick to features that are found in the main body of text.
- Read [this](#) paper regarding use of comparable corpora and incorporate the ideas into your system
- Read [this](#) paper regarding use of large scale parallel document mining
- As always, do your own research and find something we didn't mention!