Lab 8: Machine Translation

Julie Weeds

April 18, 2018

1 Getting Started

In today's lab you will be looking at ways of evaluating machine translation.

There are two plain text files provided. One `HarryPotter-en` provides the first ten sentences of the first Harry Potter novel (one sentence per line). The second `HarryPotter-fr` provides the same ten sentences translated into French by a human - i.e., this is a *gold standard* translation.

2 Tasks

1. Use Google Translate (or some other translation system) to automatically translate the English into French. Copy the output into a plain text file where there is one sentence per line. Make a second file which is an automatic translation of the French into English.

2. Write a function to compute the unigram precision for a sentence i.e., the proportion of unigrams in the automatic translation which are also found in the gold standard translation. Use your function to compute the unigram precision for each sentence in a file and return the average over the whole file. Test out your code on both the English-French translation and the French-English translation and comment on the difference.

3. One easy way to cheat on simple precision scores by just providing very common words in a translation. What is the most frequently occurring English word in the sample? What is the most frequently occurring French word in the sample? Compute average unigram precision for some nonsense translations made up of very common English / French words.

4. Modified unigram precision (Papineni et al., 2002) is not fooled by very short translations or translations which simply repeat frequent words. Implement this measure and run it on both the automatic translations provided by Google Translate and your nonsense translations.

5. Unigram precision (modified or otherwise) does not take into account the ordering of the translated words. Extend your code to also compute the average bigram precision and average trigram precision. Combine these scores by averaging. Do you think it is necessary to compute modified precision in the case of bigrams and trigrams?

3 Extensions

1. Can you modify / extend your code to implement the BLEU score as described by Papineni et al. (2002)?

2. An alternative way of evaluating automatic translations is by considering how much post-editing is required to *correct* the automatic translation into something which is both faithful and fluent. One way of assessing post-editing effort is by computing the HTER score (see `http://languagelog.ldc.upenn.edu/nll/?p=193`) - count the number of substitutions, insertions, deletions and shifts required and divide by the number of words in the gold standard translation. Can you compute the HTER scores for the French-English translation? Do you think you could automate this?

References

Kishore Papineni, Salim Roukos, Todd ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of ACL*.