Project 1: Lexical Alignment

March 18, 2019

This project will help you familiarise yourself with word-based models. Word-based models remain at the core of today's SMT systems in the form of alignment models. You will implement the simplest (though still widely used) word-based models, namely, IBM model 1, a lexical translation model, and IBM model 2, which models an impoverished form of word alignments.

In summary, your task is to

- Implement IBM model 1;
- Implement IBM model 2 using a jump distribution as in Vogel et al. (1996);
- Experiment with maximum likelihood estimation;
- Write a technical report where you present the models and an empirical comparison. Your report should also present learning curves where applicable along with a discussion explaining aspects such as non-convexity, stability and convergence.

1 IBM model 1

1. a) Implement EM training (Brown et al., 1993) for IBM model 1;
   b) All of the tasks below should be performed for both models.
2. Plot the evolution of **training** log likelihood as a function of the iteration.
3. Plot the evolution of alignment error rate (AER) on **validation** data as a function of the iteration;
4. Experiment with two criteria for model selection (i.e. deciding on number of training iterations): 1) convergence in terms of **training log likelihood**; 2) best **AER on validation** data;
5. For the selected models, obtain Viterbi alignments for every sentence pair in a test corpus and compute AER using a gold-standard provided by the assistant;

2 IBM model 2

1. Extend your previous model by implementing a full IBM model 2 (Brown et al., 1993), however using absolute positions, and the cheaper parameterisation in terms of jumps;
2. IBM 2 is non-convex, thus you will see that optimising the log-likelihood function is not as trivial as in the case of IBM model 1, particularly, convergence will depend on how you initialise the model parameters, you will try
   - uniform initialisation
   - random initialisation (try 3 different starting points)
   - initialise the lexical parameters using the output of a complete run of model 1
3. Plot **training** log-likelihood as a function of the iteration for all these cases
4. Plot **validation** AER as a function of the iteration for all these cases
5. Select two models: 1) one in terms of **training log likelihood**, 2) another in terms of **validation AER**;
6. Compare the selected models to IBM model 1 in terms of AER in the test set.

3 Data

All relevant data (including details about file formats) are available from https://uva-slpl.github.io/nlp2/projects.html.

In this project, you will work with a parallel corpus taken from the Canadian Hansards (parliament proceedings). The data consists of preprocessed sentence pairs (please do not further pre-process the data). There are two files, one for the English and one for the French sentences. Sentences with the same line number are translations of each other.

We are making available *training* data (which you can use to perform parameter estimation), *validation* data (which you can use to debug your implementation as well as to perform model selection), and finally in due time *test* data (which you will use to conduct your final empirical comparison).

You can use the results in Table 1 to sanity check your own implementation.

4 Report

You should use latex for your report, and you should use the ACL template available from http://acl2017.org/downloads/acl17-latex.zip (unlike the template suggests, your submission should not be anonymous).

Model	AER	Training regime
IBM 1	0.3378	10 iterations
IBM 2	0.2428	10 iterations (lexical component), then 5 additional iterations (lexical and jump components)

Table 1: Validation results for IBM model 1 and 2 trained for maximum likelihood via EM.

We expect short reports (5 pages plus references) written in English. The typical submission is organised as follows:

- abstract: conveys scope and contributions;
- introduction: present the problem and relevant background;
- model: technical description of models;
- experiments: details about the data, experimental setup and findings;
- conclusion: a critical take on contributions and limitations.

5 Submission

You should submit a tgz file containing a folder (folder name `lastname1.lastname2`) with the following content:

- Test predictions (in naacl format) using your best run for each of the following models
  - IBM1 MLE (filename: `ibm1.mle.naacl`)
  - IBM2 MLE (filename: `ibm2.mle.naacl`)
- Report as a single pdf file (filename: `report.pdf`)

Your report may contain a link to an open-source repository (such as github), but please do not attach code or additional data to your tgz submission.

You can complete your project submission on Canvas.

6 Assessment

Your report will be assessed according to the following evaluation criteria:

1. Scope (max 2 points): Is the problem well presented? Do students understand the challenges/contributions?
2. Theoretical description (max 3 points): Are the models presented clearly and correctly?

3. Empirical evaluation (max 3 points): Is the experimental setup sound/convincing? Are experimental findings presented in an organised and effective manner?

4. Writing style (max 2 points): use of latex, structure of report, use of tables/figures/plots, command of English.

5. Extra (max 1 point).

References

Brown, P. F., Pietra, V. J. D., Pietra, S. A. D., and Mercer, R. L. (1993). The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, 19(2):263–311.

Vogel, S., Ney, H., and Tillmann, C. (1996). HMM-based word alignment in statistical translation. In *Proceedings of the 16th Conference on Computational Linguistics - Volume 2*, COLING '96, pages 836–841, Stroudsburg, PA, USA. Association for Computational Linguistics.