Homework 1 - Report

Quanze Chen

February 4, 2015

In this project I implemented IBM models 1 and 2 with some extra normalization steps and naive intersection agreement re-iteration. In the paragraphs below I will address performance and motivation for the models chosen.

IBM Model 1: IBM Model 1 is a step up from the default Dice coefficient. As such it was able to provide relatively good results. My implementation of Model 1 executes a constant 5 rounds of EM. The vanilla implementation produced a > 30 AER, which while being greatly better than the baseline, was still making a large amount of errors. From the inspiration of the referenced papers, my first attempts to improve Model 1 were the following:

- Properly handle NULL.
- Normalize the English text

My initial implementation supported the NULL generator token but since it was placed in the beginning of each French sentence, the indexes had to be subtracted by 1. Thus 0 values would become -1. Upon fixing the problem, I noticed a slight increase in AER. Normalizing the English proved to be a lot more helpful. As stated in papers, non normalized text may be bad for the EM algorithm to converge, and the loss of normalizing caps (e.g. in proper nouns) does not affect alignment much.

After implementing normalization only on the English pairs, I produced the following results:

```
Model-1 /w case normalization of English, 5 Rounds of EM:
  AER = 29.86

  ./grade
  Precision = 0.627837
  Recall = 0.784024
  AER = 0.319319
```

And with treatment of null:

```
Model-1 /w case normalization of English, 5 Rounds of EM (fix NULL):
  AER = 29.91

  ./grade
  Precision = 0.626324
  Recall = 0.784024
  AER = 0.320320
```

It turns out that through later experiments, normalizing the case for English and French together can give around $1\% - 2\%$ decrease in AER rather consistently.

IBM Model 1 with Two Way Agreement The second enhancement to Model 1 I was able to experiment with was to take English-French aligments and French-English alignments and find the common alignments true to both runs. This has the added benefit of being lesser prone to errors, since EM must converge to the same set of alignments on both directions.

The downside to this approach is that we will be producing fewer aligned pairs than we would had we just did a single iteration. In fact we see nearly a 45% decrease in output.

```
9325559 Feb  1 01:51 hw1.txt.1 - Second run, Norm and null
5263817 Feb  3 01:45 hw1.txt.2 - Dual intersect
```

With this in mind, Model 1 aided with 2 Way agreements yielded much better AER results:

```
Model-1 /w Agreement intersection:
  AER = 23.19

  ./grade
  Precision = 0.860870
  Recall = 0.662722
  AER = 0.237189
```

Notice the shortcomings of this approach: Although we were able to increase the precision and enhance AER, we did so at the cost of Recall, which went down from 0.78 to 0.66. The algorithm is more conservative when picking alignments.

IBM Model 1 - More Iterations of EM I also ran more iterations of EM on Model 1 to see its effects on accuracy. There were some limited effects. I did achieve a slightly better AER with the local grading script, uploads show that after 10 rounds of EM on Model 1 with normalization, I got AER = 22.66, a slight improvement from just Agreement. This round runs with agreement enabled so we run double the number of iterations since we do it for both languages.

IBM Model 2 It seems that I was unable to enhance IBM Model 1 any further, so I went to try a model with more data assumptions. Model 2 also takes into account the position of a word in the sentence. A naive approach yielded an AER of 22.55, which is marginally better than the best performing Model 1 I could get.

```
Model-2 /w normalization, no duality, 5 Rounds of EM:
  AER = 22.55

  ./grade
  Precision = 0.730711
  Recall = 0.831361
  AER = 0.235235
```

Note that since we are not using any intersection agreement, we are again getting high recall while still maintaining a good AER. In this sense Model 2 offers us another baseline to improve on.

IBM Model 2 + IBM Model 1 Since the models take long to run after iterations, in between I decided to see if the models can be combined to enhance results. I combined the output of running model 1 (EM=5, With intersecting) and the output of a base Model 2 (EM = 5, No intersect) by taking the intersection of both.

```
Model-1+2 1/w duality 2/wo duality
  AER = 21.70

  ./grade
  Precision = 0.927152
  Recall = 0.639053
  AER = 0.225000
```

Again we see a slight increase in AER, followed by a large increase in precision but with a decrease in Recall as we would expect when taking the intersection. The interesting thing is that by intersecting the results of the two models, we got an overall better result than either of the individual runs. (Precision 0.93 up from 0.73 and 0.86, while Recall was lower than the lower of the two.)

IBM Model 2 - Agreement Union For this part I decided to attempt using the union of results from English-French and French-English alignments. This will give more data to work with than the intersections but introduces a lot of noise.

```
Precision = 0.628024
Recall = 0.899408
AER = 0.303008
```

Now this itself seems to be a backwards step since we are increasing AER greatly and precision is low. However, the Recall is highest among models (We produced 155% size of a single run of Model 2.) Naturally with such a high recall, maybe we can use intersection again to get a better AER.

So I attempted to intersect this data with our Model 1 (EM=5, With intersecting) model. This produces in general a lower AER, higher precision and recall does not drop too much from the lowest of the two:

```
Model-1+2 Union
  AER = 20.87

  ./grade
  Precision = 0.911315
  Recall = 0.659763
  AER = 0.216541
```

At this point we have enhanced AER from Model 2 and the best possible Model 1 by around 2%. Also in this round we attempted to normalize both English and French text to lowercase.

IBM Model 2 With Agreement Since using intersection yielded so much improvement in AER for Model 1, I naturally attempted it for Model 2. We trained the model, again using enhancements from before like normalizing text, at EM=5 rounds. We saw a consistent improvement in AER:

```
Model-2 /w case normalization, duality, 5 Rounds of EM:
  AER = 16.59

  ./grade
  Precision = 0.888041
  Recall = 0.745562
  AER = 0.177839
```

This yielded much better AER and surprisingly with not much decrease in Recall.

Further attempts to intersect this result and Model 1 returned mixed results, getting a tiny increase when using the grade script provided, but a worse AER for the online testing. Further experimentation such as more iterations of EM for Model 2 were not attempted due to lack of time.

Conclusion I saw that in general there are two ways to improve on the vanilla IBM models. One is to normalize the text case, which consistently yields around a $1 - 2\%$ enhancement of AER.

Another more effective way was to run the reverse model and make an agreement on alignment. This agreement consistently produces a decrease in AER of around 5%. We also see that IBM Model 1 with agreement is more susceptible to decreases in Recall while Model 2 did not exhibit as much change. This may be because inherently Model 2 is performing better and is spreading out the alignments less.

Having a more conservative algorithm does increase accuracy, though accuracy alone may not be what we really want.