Simplex Algorithms for Reranking

Akshay Srivatsan, Ian Mukherjee, Nathan Smith

March 31, 2016

1 Introduction

Our group incrementally implemented various iterations of the simplex algorithm and new features, beginning with a simple three-dimensional simplex algorithm implementation. We then added features to count the number of words in a sentence as well as the difference in length between the source sentence and translation. With these two additional features we now needed a simplex algorithm with six points for the $n + 1$ parameters. We experimented with various convergence thresholds, where when the absolute difference between the "worst" and "best" point bleu scores was below this given threshold a new iteration of the simplex algorithm took place. This final implementation yielded an optimal bleu score of 0.29274, with slight variations in the optimal score each runtime, but almost always exceeding 0.291. This can be run with the command `./simplex__final`

2 First Simplex Model:

The rerank method was first tested, yielding the following results:

Bleu Test for Rerank Model:

Bleu Score	0.27351

Firstly, following the outline in the lecture slides, we implemented the simplex method with four points to handle the three initially provided weights. In the simplex method, four initial points are randomly generated, and reflect, refine, and shrink steps are used to converge towards an optimal bleu score point. When the best and worst points of the simplex fall below our provided threshold of 0.001, we cease the program and return the final optimal bleu score found. For this initial implementation, results were as follows:

Bleu Test for 3-Weight Simplex Model:

Bleu Score	0.282

While this yielded significant improvements over the initial rerank algorithm, the addition of new features and simplex points was a clear way to further improve the optimal bleu score.

3 Finalized Sentence Model:

After completing a basic simplex implementation, we considered additional features and algorithm improvements. Two features were added: the length of the translation sentence, and the difference in length between the translation and source sentences. These new features were computed and appended to the dev+test.100best file in identical format to the previous features. These two features were then added to the weights table. Next, we considered further improvements to the simplex algorithm to improve bleu scores and runtime. An iteration feature was implemented, where when the difference between scores for the best and worst points went below the given threshold, the best point was added with a random sample of five other points to create a new simplex, which again continued to reflect, refine, and shrink until convergence. For slight speedups, we initially randomly sample 20 points and compute their bleu scores, and choose the best 6 to form the initialization of the simplex. A number of iteration and threshold combinations were experimented with to optimize scores and runtime, and 5 iterations with a threshold of 0.001 was found to be optimal. Results were as follows:

Bleu Test for Optimized Simplex Model:

Bleu Score	0.292786

4 Bibliography:

References

[1] Mark Hopkins, Jonathan May. Tuning as ranking.

[2] Nadir Durrani, Helmut Schmid, Alexander Fraser, Philipp Koehn, Hinrich Schutze The operation Sequence Model

[3] Daniel Ortiz Martinez, Ismael Garcia Varea, Francisco Casacuberta Nolla Generalized Stack Decoding Algorithms for Statistics Machine Translation