Some Experiments with a Convex IBM Model 2

Andrei Simion
Columbia University
IEOR Department
New York, NY, 10027
aas2148@columbia.edu

Michael Collins
Columbia University
Computer Science
New York, NY, 10027
mc3354@columbia.edu

Clifford Stein
Columbia University
IEOR Department
New York, NY, 10027
cs2035@columbia.edu

Abstract

Using a recent convex formulation of IBM Model 2, we propose a new initialization scheme which has some favorable comparisons to the standard method of initializing IBM Model 2 with IBM Model 1. Additionally, we derive the Viterbi alignment for the convex relaxation of IBM Model 2 and show that it leads to better F-Measure scores than those of IBM Model 2.

1 Introduction

The IBM translation models are widely used in modern statistical translation systems. Unfortunately, apart from Model 1, the IBM models lead to non-convex objective functions, leading to methods (such as EM) which are not guaranteed to reach the global maximum of the log-likelihood function. In a recent paper, Simion et al. introduced a convex relaxation of IBM Model 2, I2CR-2, and showed that it has performance on par with the standard IBM Model 2 (Simion et al., 2013).

In this paper we make the following contributions:

- We explore some applications of I2CR-2. In particular, we show how this model can be used to seed IBM Model 2 and compare the speed/performance gains of our initialization under various settings. We show that initializing IBM Model 2 with a version of I2CR-2 that uses large batch size yields a method that has similar run time to IBM Model 1 initialization and at times has better performance.

- We derive the Viterbi alignment for I2CR-2 and compare it directly with that of IBM Model 2. Previously, Simion et al. (2013) had compared IBM Model 2 and I2CR-2 by using IBM Model 2's Viterbi alignment rule, which is not necessarily the optimal alignment for I2CR-2. We show that by comparing I2CR-2 with IBM Model 2 by using each model's optimal Viterbi alignment the convex model consistently has a higher F-Measure. F-Measure is an important metric because it has been shown to be correlated with BLEU scores (Marcu et al., 2006).

Notation. We adopt the notation introduced in (Och and Ney, 2003) of having 1^m2^n denote the training scheme of m IBM Model 1 EM iterations followed by initializing Model 2 with these parameters and running n IBM Model 2 EM iterations. The notation $\text{EG}_B^m2^n$ means that we run m iterations of I2CR-2's EG algorithm (Simion et al., 2013) with batch size of B, initialize IBM Model 2 with I2CR-2's parameters, and then run n iterations of Model 2's EM.

2 The IBM Model 1 and 2 Optimization Problems

In this section we give a brief review of IBM Models 1 and 2 and the convex relaxation of Model 2, I2CR-2 (Simion et al., 2013). The standard approach in training parameters for Models 1 and 2 is EM, whereas for I2CR-2 an exponentiated-gradient (EG) algorithm was developed (Simion et al., 2013).

We assume that our set of training examples is $(e^{(k)}, f^{(k)})$ for $k = 1 \ldots n$, where $e^{(k)}$ is the k'th English sentence and $f^{(k)}$ is the k'th French sentence. The k'th English sentence is a sequence of words $e_1^{(k)} \ldots e_{l_k}^{(k)}$ where l_k is the length of the k'th English sentence, and each $e_i^{(k)} \in E$; similarly the k'th French sentence is a sequence $f_1^{(k)} \ldots f_{m_k}^{(k)}$ where each $f_j^{(k)} \in F$. We define $e_0^{(k)}$ for $k = 1 \ldots n$ to be a special NULL word (note that E contains the NULL word). IBM Model 2 is detailed in several sources such as (Simion et al., 2013) and (Koehn, 2004).

The convex and non-convex objectives of respectively IBM Model 1 and 2 can be found in (Simion

Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, pages 180–184,
Gothenburg, Sweden, April 26-30 2014.

et al., 2013). For I2CR-2, the convex relaxation of IBM Model 2, the objective is given by

$$\frac{1}{2n}\sum_{k=1}^{n}\sum_{j=1}^{m_k}\log'\sum_{i=0}^{l_k}\frac{t(f_j^{(k)}|e_i^{(k)})}{(L+1)} + \frac{1}{2n}\sum_{k=1}^{n}\sum_{j=1}^{m_k}\log'\sum_{i=0}^{l_k}\min\{t(f_j^{(k)}|e_i^{(k)}), d(i|j)\}\ .$$

For smoothness reasons, Simion et al. (2013) defined $\log'(z) = \log(z+\lambda)$ where $\lambda = .001$ is a small positive constant. The I2CR-2 objective is a convex combination of the convex IBM Model 1 objective and a direct (convex) relaxation of the IBM2 Model 2 objective, and hence is itself convex.

3 The Viterbi Alignment for I2CR-2

Alignment models have been compared using methods other than Viterbi comparisons; for example, Simion et al. (2013) use IBM Model 2's optimal rule given by (see below) Eq. 2 to compare models while Liang et al. (2006) use posterior decoding. Here, we derive and use I2CR-2's Viterbi alignment. To get the Viterbi alignment of a pair $(e^{(k)}, f^{(k)})$ using I2CR-2 we need to find $a^{(k)} = (a_1^{(k)}, \ldots, a_{m_k}^{(k)})$ which yields the highest probability $p(f^{(k)}, a^{(k)}|e^{(k)})$. Referring to the I2CR-2 objective, this corresponds to finding $a^{(k)}$ that maximizes

$$\frac{\log\prod_{j=1}^{m_k} t(f_j^{(k)}|e^{(k)}_{a_j^{(k)}})}{2} + \frac{\log\prod_{j=1}^{m_k}\min\{t(f_j^{(k)}|e^{(k)}_{a_j^{(k)}}), d(a_j^{(k)}|j)\}}{2}\ .$$

Putting the above terms together and using the monotonicity of the logarithm, the above reduces to finding the vector $a^{(k)}$ which maximizes

$$\prod_{j=1}^{m_k} t(f_j^{(k)}|e^{(k)}_{a_j^{(k)}})\min\{t(f_j^{(k)}|e^{(k)}_{a_j^{(k)}}), d(a_j^{(k)}|j)\}.$$

As with IBM Models 1 and 2, we can find the vector $a^{(k)}$ by splitting the maximization over the components of $a^{(k)}$ and focusing on finding $a_j^{(k)}$ given by

$$\text{argmax}_a(t(f_j^{(k)}|e_a^{(k)})\min\{t(f_j^{(k)}|e_a^{(k)}), d(a|j)\})\ . \quad (1)$$

In previous experiments, Simion et al. (Simion et al., 2013) were comparing I2CR-2 and IBM Model 2 using the standard alignment formula derived in a similar fashion from IBM Model 2:

$$a_j^{(k)} = \text{argmax}_a(t(f_j^{(k)}|e_a^{(k)})d(a|j))\ . \quad (2)$$

4 Experiments

In this section we describe experiments using the I2CR-2 optimization problem combined with the stochastic EG algorithm (Simion et al., 2013) for parameter estimation. The experiments conducted here use a similar setup to those in (Simion et al., 2013). We first describe the data we use, and then describe the experiments we ran.

4.1 Data Sets

We use data from the bilingual word alignment workshop held at HLT-NAACL 2003 (Michalcea and Pederson, 2003). We use the Canadian Hansards bilingual corpus, with 247,878 English-French sentence pairs as training data, 37 sentences of development data, and 447 sentences of test data (note that we use a randomly chosen subset of the original training set of 1.1 million sentences, similar to the setting used in (Moore, 2004)). The development and test data have been manually aligned at the word level, annotating alignments between source and target words in the corpus as either "sure" (S) or "possible" (P) alignments, as described in (Och and Ney, 2003).

As a second data set, we used the Romanian-English data from the HLT-NAACL 2003 workshop consisting of a training set of 48,706 Romanian-English sentence-pairs, a development set of 17 sentence pairs, and a test set of 248 sentence pairs.

We carried out our analysis on this data set as well, but because of space we only report the details on the Hansards data set. The results on the Romanian data were similar, but the magnitude of improvement was smaller.

4.2 Methodology

Our experiments make use of either standard training or intersection training (Och and Ney, 2003). For standard training, we run a model in the source-target direction and then derive the alignments on the test or development data. For each of the

Training	2^{10}	$1^5 2^{10}$	$\text{EG}^1_{125} 2^{10}$	$\text{EG}^1_{1250} 2^{10}$
Iteration		Objective		
0	-224.0919	-144.2978	-91.2418	-101.2250
1	-110.6285	-85.6757	-83.3255	-85.5847
2	-91.7091	-82.5312	-81.3845	-82.1499
3	-84.8166	-81.3380	-80.6120	-80.9610
4	-82.0957	-80.7305	-80.2319	-80.4041
5	-80.9103	-80.3798	-80.0173	-80-1009
6	-80.3620	-80.1585	-79.8830	-79.9196
7	-80.0858	-80.0080	-79.7911	-79.8048
8	-79.9294	-79.9015	-79.7247	-79.7284
9	-79.8319	-79.8240	-79.6764	-79.6751
10	-79.7670	-79.7659	-79.6403	-79.6354

Table 1: Objective results for the English → French IBM Model 2 seeded with either uniform parameters, IBM Model 1 ran for 5 EM iterations, or I2CR-2 ran for 1 iteration with either B = 125 or 1250. Iteration 0 denotes the starting IBM 2 objective depending on the initialization.

models—IBM Model 1, IBM Model 2, and I2CR-2— we apply the conventional methodology to intersect alignments: first, we estimate the t and d parameters using models in both source-target and target-source directions; second, we find the most likely alignment for each development or test data sentence in each direction; third, we take the intersection of the two alignments as the final output from the model. For the I2CR-2 EG (Simion et al., 2013) training, we use batch sizes of either $B = 125$ or $B = 1250$ and a step size of $\gamma = 0.5$ throughout.

We measure the performance of the models in terms of *Precision*, *Recall*, *F-Measure*, and *AER* using only sure alignments in the definitions of the first three metrics and sure and possible alignments in the definition of AER, as in (Simion et al., 2013) and (Marcu et al., 2006). For our experiments, we report results in both AER (lower is better) and F-Measure (higher is better).

4.3 Initialization and Timing Experiments

We first report the summary statistics on the test set using a model trained only in the English-French direction. In these experiments we seeded IBM Model 2's parameters either with those of IBM Model 1 run for 5, 10 or 15 EM iterations or I2CR-2 run for 1 iteration of EG with a batch size of either $B = 125$ or 1250. For uniform comparison, all of our implementations were written in C++ using STL/Boost containers.

There are several takeaways from our experiments, which are presented in Table 2. We first note that with $B = 1250$ we get higher F-Measure and lower AER even though we use less training time: 5 iterations of IBM Model 1 EM training takes about 3.3 minutes, which is about the time it takes for 1 iteration of EG with a batch size of 125 (4.1 minutes); on the other hand, using $B = 1250$ takes EG 1.7 minutes and produces the best results across almost all iterations. Additionally, we note that the initial solution given to IBM Model 2 by running I2CR-2 for 1 iteration with $B = 1250$ is fairly strong and allows for further progress: IBM2 EM training improves upon this solution during the first few iterations. We also note that this behavior is global: no IBM 1 initialization scheme produced subsequent solutions for IBM 2 with as low in AER or high in F-Measure. Finally, comparing Table 1 which lists objective values with Table 2 which lists alignment statistics, we see that although the objective progression is similar throughout, the alignment quality is different.

To complement the above, we also ran intersection experiments. Seeding IBM Model 2 by Model 1 and intersecting the alignments produced by the English-French and French-English models gave both AER and F-Measure which were better than those that we obtained by any seeding of IBM Model 2 with I2CR-2. However, there are still reasons why I2CR-2 would be useful in this context. In particular, we note that I2CR-2 takes roughly half the time to progress to a better solution than IBM Model 1 run for 5 EM iterations. Second, a possible remedy to the above loss in marginal improvement when taking intersections would be to use a more refined method for obtaining the joint alignment of the English-French and French-English models, such as "grow-diagonal" (Och and Ney, 2003).

4.4 Viterbi Comparisons

For the decoding experiments, we used IBM Model 1 as a seed to Model 2. To train IBM Model 1, we follow (Moore, 2004) and (Och and Ney, 2003) in running EM for 5, 10 or 15 iterations. For the EG algorithm, we initialize all parameters uniformly and use 10 iterations of EG with a batch size of 125. Given the lack of development data for the alignment data sets, for both IBM Model 2 and the I2CR-2 method, we report test set F-Measure and AER results for each of the 10 iterations, rather than picking the results from a single iteration.

Training	2^{10}	$1^5 2^{10}$	$1^{10} 2^{10}$	$1^{15} 2^{10}$	$\text{EG}^1_{125} 2^{10}$	$\text{EG}^1_{1250} 2^{10}$
Iteration			AER			
0	0.8713	0.3175	0.3177	0.3160	**0.2329**	0.2662
1	0.4491	0.2547	0.2507	0.2475	0.2351	**0.2259**
2	0.2938	0.2428	0.2399	0.2378	0.2321	**0.2180**
3	0.2593	0.2351	0.2338	0.2341	0.2309	**0.2176**
4	0.2464	0.2298	0.2305	0.2310	0.2283	**0.2168**
5	0.2383	0.2293	0.2299	0.2290	0.2268	**0.2188**
6	0.2350	0.2273	0.2285	0.2289	0.2274	**0.2205**
7	0.2320	0.2271	0.2265	0.2286	0.2274	**0.2213**
8	0.2393	0.2261	0.2251	0.2276	0.2278	**0.2223**
9	0.2293	0.2253	0.2246	0.2258	0.2284	**0.2217**
10	0.2288	0.2248	0.2249	0.2246	0.2275	**0.2223**
Iteration			F-Measure			
0	0.0427	0.5500	0.5468	0.5471	**0.6072**	0.5977
1	0.4088	0.5846	0.5876	0.5914	0.6005	**0.6220**
2	0.5480	0.5892	0.5916	0.5938	0.5981	**0.6215**
3	0.5750	0.5920	0.5938	0.5947	0.5960	**0.6165**
4	0.5814	0.5934	0.5839	0.5952	0.5955	**0.6129**
5	0.5860	0.5930	0.5933	0.5947	0.5945	**0.6080**
6	0.5873	0.5939	0.5936	0.5940	0.5924	**0.6051**
7	0.5884	0.5931	0.5955	0.5941	0.5913	**0.6024**
8	0.5899	0.5932	0.5961	0.5942	0.5906	**0.6000**
9	0.5899	0.5933	0.5961	0.5958	0.5906	**0.5996**
10	0.5897	0.5936	0.5954	0.5966	0.5910	**0.5986**

Table 2: Results on the Hansards data for English → French IBM Model 2 seeded using different methods. The first three columns are for a model seeded with IBM Model 1 ran for 5, 10 or 15 EM iterations. The fourth and fifth columns show results when we seed with I2CR-2 ran for 1 iteration either with $B = 125$ or 1250. Iteration 0 denotes the starting statistics.

Training	$1^5 2^{10}$	$1^{10} 2^{10}$	$1^{15} 2^{10}$	EG^{10}_{125}	EG^{10}_{125}
Viterbi Rule	$t \times d$	$t \times d$	$t \times d$	$t \times d$	$t \times \min\{t \times d\}$
Iteration			AER		
0	**0.2141**	0.2159	0.2146	0.9273	0.9273
1	0.1609	0.1566	**0.1513**	0.1530	0.1551
2	0.1531	0.1507	0.1493	0.1479	**0.1463**
3	0.1477	0.1471	0.1470	0.1473	**0.1465**
4	0.1458	**0.1444**	0.1449	0.1510	0.1482
5	0.1455	0.1438	**0.1435**	0.1501	0.1482
6	0.1436	0.1444	**0.1429**	0.1495	0.1481
7	0.1436	**0.1426**	0.1435	0.1494	0.1468
8	0.1449	**0.1427**	0.1437	0.1508	0.1489
9	0.1454	**0.1426**	0.1430	0.1509	0.1481
10	0.1451	0.1430	**0.1423**	0.1530	0.1484
Iteration			F-Measure		
0	**0.7043**	0.7012	0.7021	0.0482	0.0482
1	0.7424	0.7477	0.7534	0.7395	**0.7507**
2	0.7468	0.7499	0.7514	0.7448	**0.7583**
3	0.7489	0.7514	0.7520	0.7455	**0.7585**
4	0.7501	0.7520	0.7516	0.7418	**0.7560**
5	0.7495	0.7513	0.7522	0.7444	**0.7567**
6	0.7501	0.7501	0.7517	0.7452	**0.7574**
7	0.7493	0.7517	0.7507	0.7452	**0.7580**
8	0.7480	0.7520	0.7504	0.7452	**0.7563**
9	0.7473	0.7511	0.7513	0.7450	**0.7590**
10	0.7474	0.7505	0.7520	0.7430	**0.7568**

Table 3: Intersected results on the English-French data for IBM Model 2 and I2CR-2 using either IBM Model 1 trained to 5, 10, or 15 EM iterations to seed IBM2 and using either the IBM2 or I2CR-2 Viterbi formula for I2CR-2.

In Table 3 we report F-Measure and AER results for each of the iterations under IBM Model 2 and I2CR-2 models using either the Model 2 Viterbi rule of Eq. 2 or I2CR-2's Viterbi rule in Eq. 1. We note that unlike in the previous experiments presented in (Simion et al., 2013), we are directly testing the quality of the alignments produced by I2CR-2 and IBM Model 2 since we are getting the Viterbi alignment for each model (for completeness, we also have included in the fourth column the Viterbi alignments we get by using the IBM Model 2 Viterbi formula with the I2CR-2 parameters as Simion et al. (2013) had done previously). For these experiments we report intersection statistics. Under its proper decoding formula, I2CR-2 model yields a higher F-Measure than any setting of IBM Model 2. Since AER and BLEU correlation is arguably known to be weak while F-Measure is at times strongly related with BLEU (Marcu et al., 2006), the above results favor the convex model.

We close this section by pointing out that the main difference between the IBM Model 2 Viterbi rule of Eq. 2 and the I2CR-2 Viterbi rule in Eq. 1 is that the Eq. 1 yield fewer alignments when doing intersection training. Even though there are fewer alignments produced, the quality in terms of F-Measure is better.

5 Conclusions and Future Work

In this paper we have explored some of the details of a convex formulation of IBM Model 2 and showed it may have an application either as a new initialization technique for IBM Model 2 or as a model in its own right, especially if the F-Measure is the target metric. Other possible topics of interest include performing efficient sensitivity analysis on the I2CR-2 model, analyzing the balance between the IBM Model 1 and I2CR-1 (Simion et al., 2013) components of the I2CR-2 objective, studying I2CR-2's intersection training performance using methods such as "grow diagonal" or "agreement" (Liang et al., 2006), and integrating it into the GIZA++ open source library so we can see how much it affects the downstream system.

Acknowledgments

Michael Collins and Andrei Simion are partly supported by NSF grant IIS-1161814. Cliff Stein is

partly supported by NSF grants CCF-0915681 and CCF-1349602. We thank Professor Paul Blaer and Systems Engineer Radu Sadeanu for their help setting up some of the hardware used for these experiments. We also thank the anonymous reviewers for many useful comments; we hope to pursue the comments we were not able to address in a followup paper.

References

Peter L. Bartlett, Ben Taskar, Michael Collins and David Mcallester. 2004. Exponentiated Gradient Algorithms for Large-Margin Structured Classification. *In Proceedings of NIPS*.

Steven Boyd and Lieven Vandenberghe. 2004. Convex Optimization. Cambridge University Press.

Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert. L. Mercer. 1993. The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics*, 19:263-311.

Michael Collins, Amir Globerson, Terry Koo, Xavier Carreras and Peter L. Bartlett. 2008. Exponentiated Gradient Algorithms for Conditional Random Fields and Max-Margin Markov Networks. *Journal Machine Learning,* 9(Aug): 1775-1822.

A. P. Dempster, N. M. Laird, and D. B. Rubin. 1977. Maximum Likelihood From Incomplete Data via the EM Algorithm. *Journal of the royal statistical society, series B*, 39(1):1-38.

Alexander Fraser and Daniel Marcu. 2007. Measuring Word Alignment Quality for Statistical Machine Translation. *Journal Computational Linguistics*, 33(3): 293-303.

Joao V. Graca, Kuzman Ganchev and Ben Taskar. 2007. Expectation Maximization and Posterior Constraints. *In Proceedings of NIPS.*

Yuhong Guo and Dale Schuurmans. 2007. Convex Relaxations of Latent Variable Training. *In Proceedings of NIPS.*

Simon Lacoste-Julien, Ben Taskar, Dan Klein, and Michael Jordan. 2008. Word Alignment via Quadratic Assignment. *In Proceedings of the HLT-NAACL.*

Phillip Koehn. 2004. Statistical Significance Tests for Machine Translation Evaluation. *In Proceedings of the EMNLP*.

Phillip Koehn. 2008. Statistical Machine Translation. Cambridge University Press.

Kivinen, J., Warmuth, M. 1997. Exponentiated Gradient Versus Gradient Descent for Linear Predictors. *Information and Computation*, 132, 1-63.

Percy Liang, Ben Taskar and Dan Klein. 2006. Alignment by Agreement. *In Proceedings of NAACL.*

Daniel Marcu, Wei Wang, Abdessamad Echihabi, and Kevin Knight. 2006. SPMT: Statistical Machine Translation with Syntactified Target Language Phrases. *In Proceedings of the EMNLP*.

Rada Michalcea and Ted Pederson. 2003. An Evaluation Exercise in Word Alignment. *HLT-NAACL 2003: Workshop in building and using Parallel Texts: Data Driven Machine Translation and Beyond.*

Robert C. Moore. 2004. Improving IBM Word-Alignment Model 1. *In Proceedings of the ACL.*

Stephan Vogel, Hermann Ney and Christoph Tillman. 1996. HMM-Based Word Alignment in Statistical Translation. *In Proceedings of COLING*.

Franz Och and Hermann Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. *Computational-Linguistics*, 29(1): 19-52.

Andrei Simion, Michael Collins and Cliff Stein. 2013. A Convex Alternative to IBM Model 2. *In Proceedings of the EMNLP*.

Kristina Toutanova and Michel Galley. 2011. Why Initialization Matters for IBM Model 1: Multiple Optima and Non-Strict Convexity. *In Proceedings of the ACL.*

Ashish Vaswani, Liang Huang and David Chiang. 2012. Smaller Alignment Models for Better Translations: Unsupervised Word Alignment with the $L0$-norm. *In Proceedings of the ACL.*