HMM Training and Phrase-Extraction Strategies for PORTAGE

George Foster
National Research Council Canada
first.last@nrc.gc.ca

14 Dec 09

1 Introduction

This report describes the results of experiments to determine the optimal training and phrase-extraction settings for HMM-based phrase tables. These were motivated by the implementation of new options for HMM training (bi-lexical conditioning, corrected end-distribution semantics, etc.) as well as the elimination of the "insect", which had adversely affected phrase extraction by adding phrase pairs consisting entirely of unaligned words.

2 Setting

Experiments were performed on two systems:

- A **French-English** system trained on the Hansard (approx 5M sentence pairs); and
- A **Chinese-English** system trained on a subset of the NIST09 data (approx 3M sentence pairs). Training data excluded the large UN and Hong Kong Hansard corpora, which typically contribute little to performance on the standard newswire and webtext test corpora.

Both systems were based on a single phrase table extracted from the training corpus using HMM alignments only (no IBM2 alignments). Phrase-table features (in both directions) were unsmoothed relative-frequency estimates, and Zens-Ney estimates derived from on the HMM ttable. Other features were word-count, displacement distortion (non-lexicalized), and a

4-gram language model trained on the parallel corpus. The NIST09 system additionally used a Gigaword 5-gram LM. Other parameters were standard. The complete configurations, including results, are available on balzac in:

```
/home/fosterg/experiments/phrase-extraction/nist09
/home/fosterg/experiments/phrase-extraction/hans.fren
```

To reduce MERT variation, parameter tuning was carried out only once. Four separate random MERT runs were performed for each system using a baseline configuration, and the weights that gave the best BLEU scores on the test corpora were fixed for all remaining experiments, which did not vary in the number of features used. This strategy has the additional advantage that the lengthy MERT procedure does not have to be re-run for each experiment.

Results were measured using BLEU on two randomly-chosen test sets for the Hansard, and on the NIST06/08 eval sets for the NIST09 system.

description	parameter	value
— HMM params —		
"new" HMM implementation	`-newhmm`	always set
transition probability to a null alignment	`-p0`	0.0
length-dependent null-alignment probability increment	`-up0`	1.0
maximum parameterized jump (higher jumps are binned)	`-max-jump`	0 (none)
use a distinct distr for jumps from the start position	`-start-dist`	no
use a distinct distr for jumps to the final state	`-final-dist`	no
anchor alignments to end of sequence	`-anchor`	no
condition jumps on given word-classes	`-word-classes-l1`	no
condition jumps on given type of lexical/class-based events	`-condition-on`	no
interpolation coeff for uniform smoothing of jumps	`-alpha`	0.01
additive value for count-based smoothing of jumps	`-lambda`	0.0
weight of MAP prior for lexicalized jumps	`-map-tau`	0 (none)
perform symmetrized training using specified method	`-symmetrized`	no
— phrase-extraction params —		
symmetrized alignment algorithm	`-a`	Och 3
number of best IBM src/tgt translations to add to phrasetable	`-w`	1
maximum phrase length	`-m`	7
max difference in src/tgt phrase lengths	`-d`	4
the HMM `-p0` parameter above, but re-set for alignment	`-p0`	0.0

Table 1: Baseline Parameter Settings

parameters	hans			nist09		
baseline	54.56	55.53	55.05	32.10	26.00	29.05
`-start-dist`	54.58	55.56	55.09	32.33	26.08	29.21
`-start-dist -anchor`	54.66	55.59	55.13	32.21	26.17	29.19
`-start-dist -final-dist`	54.63	55.60	55.11	32.18	26.23	29.21
`-start-dist -final-dist -anchor`	54.66	55.61	**55.13**	32.38	26.36	**29.37**

Table 2: Start and Final Distributions

3 Results—HMM Training

The general strategy in the experiments was to try to work through the training sequence, beginning with HMM training and ending with phrase extraction, greedily choosing and fixing the best parameter settings at each step. Table 1 shows the baseline setting, and includes all significant parameters that were changed in these experiments.

In all the results tables that follow, unless otherwise specified, the results for the Hansard system are given in three columns, pertaining to the two test sets and their average. Results for NIST09 are also given in three columns, corresponding to the NIST06 and NIST08 evaluation sets and their average. All scores are percent BLEU.

3.1 Start and Final Distributions

Table 2 shows the results for various settings of the start/final distributions options, which use special parameterizations for the beginning and ends of sentences. For both systems, it appears to be advantageous to use all available parameters, which give gains over the baseline of approximately 0.1 and 0.3 BLEU for the Hansard and NIST09 systems.

3.2 Lexical and Class-Based Conditioning

Table 3 shows the results for various methods of conditioning HMM jumps on words or word-classes. The starting configuration for all experiments is the best one from table 2 (start + final + anchor).

- The 2nd line shows the results for conditioning the current jump probability on the current hidden-sequence class, one of 100 classes derived from Och's mkcls program. This does not improve over the baseline.
- The next 5 lines show results for conditioning the jump probability on the previous hidden-sequence word, using the `-map-tau` value as the

parameters	hans			nist09		
table 2 baseline	54.66	55.61	**55.13**	32.38	26.36	29.37
`-word-classes-l1 100`	54.37	55.58	54.98	32.10	26.04	29.07
`-map-tau 10`	54.10	55.36	54.73	32.51	26.20	29.35
`-map-tau 100`	54.04	55.42	54.73	32.68	26.37	29.52
`-map-tau 500`	—	—	—	32.51	26.62	**29.56**
`-map-tau 1000`	54.09	55.40	54.75	32.49	26.60	29.55
`-map-tau 10000`	54.37	55.52	54.94	—	—	—
`-map-tau 10k -condition-on n-o`	54.37	55.35	54.86	—	—	—
`-map-tau 500 -condition-on n-o`	—	—	—	32.14	26.19	29.16
`-map-tau 10k -condition-on p-o`	54.24	55.26	54.75	—	—	—
`-map-tau 500 -condition-on p-o`	—	—	—	32.15	25.97	29.06
`-map-tau 1000 -condition-on p-h:n-o`	54.56	55.35	54.96	—	—	—
`-map-tau 500 -condition-on p-h:p-o`	—	—	—	32.36	26.03	29.20

Table 3: Lexical and Class-Based Conditioning

weight on the prior distribution, which is not conditioned on specific words. This improves over the baseline for the NIST09 system, at an optimal `-map-tau` value of 500, but does not give improvements for the Hansard system.

- The next 4 lines condition jump probability on next-observed (n-o) and prev-observed (p-o) words, using `-map-tau` settings optimal for each system. Both strategies are inferior to prev-hidden conditioning.

- The final 2 lines condition jump probability on the previous hidden-sequence word and the next observed word (p-h:n-o), and on the previous hidden-sequence and the previous observed word (p-h:n-o). These do not improve over the prev-hidden strategy either. Because these runs have large memory requirements (32 * 4G is an upper bound), only the configurations shown were tested for each system, and the option of conditioning on both the observed sequence words was not tried.

The conclusion from these conditioning tests is that the standard prev-hidden conditioning gives a gain of about 0.2 BLEU for the NIST09 system with `-map-tau 500`. No conditioning method improved over the baseline for the Hansard system.

parameters	hans			nist09		
table 3 baselines	54.66	55.61	55.13	32.51	26.62	29.56
`-max-jump 10`	54.61	55.59	55.10	32.11	26.19	29.15
`-max-jump 20`	54.67	55.62	55.15	32.28	26.27	29.27
`-lambda 1.0`	54.75	55.64	**55.19**	32.44	26.70	**29.57**
`-lambda 2.0`	—	—	—	32.44	26.70	29.57
`-lambda 10.0`	54.75	55.64	55.19	—-	—-	—-

Table 4: Smoothing Jumps

3.3 Smoothing Jumps

Table 4 shows results for various ways of smoothing the HMM jump parameters. The baseline configurations are the best from table 3 (different for each system).

- The first two lines after the baseline show results for binning jumps longer than a given value. The baseline uses no limit. There is a very slight improvement for the Hansard system at 20, and none for NIST09. (To corroborate the lexical results from the previous section, the prev-hidden conditioning model was run for Hansard with these two jump settings. Performance remained well below the non-conditioned model used here.)

- The final group of lines shows performance using additive smoothing for jump distributions instead of the default interpolated smoothing. There is a very slight gain over previous-best configurations (Hansard used `-max-jump 20` for these tests; NIST09 used `-max-jump 0`) for adding 1.0, and no change for larger values. Since the baseline interpolated smoothing has a much larger effect for most jumps than adding 1, this result may indicate that less jump smoothing is better.

3.4 Jump-to-Null

The probabilities of jumping to a null word are given by $p0 + up0/(I + 1)$, where $p0$ and $up0$ are values set by the corresponding switches, and I is the length of the hidden (conditioning) sequence. Table 5 shows the results of varying these parameters from the default values of 0 and 1 respectively.

- The 2nd group of lines tunes `-p0` with `-up0` set to 0. There is a relatively large effect on performance, with maximum average BLEU

parameters	hans				nist09			
table 4 baselines	54.75	55.64	55.19	110	32.44	26.70	29.57	20
`-p0 0.2 -up0 0`	55.00	55.63	55.31	162	32.69	26.57	29.63	32
`-p0 0.3 -up0 0`	—	—	—	—	32.94	26.54	29.74	39
`-p0 0.4 -up0 0`	55.38	55.76	55.57	222	33.06	26.85	**29.95**	50
`-p0 0.5 -up0 0`	55.52	55.96	55.74	253	32.91	26.78	29.84	61
`-p0 0.6 -up0 0`	55.66	55.86	55.76	290	32.69	26.74	29.72	76
`-p0 0.7 -up0 0`	55.61	55.95	55.78	338	32.57	26.57	29.57	99
`-p0 0.8 -up0 0`	55.33	55.67	55.50	409	32.29	26.23	29.26	140
`-p0 0.0 -up0 1.5`	54.66	55.41	55.03	117	32.62	26.56	29.59	21
`-p0 0.0 -up0 1.8`	54.68	55.48	55.08	122	32.54	26.60	29.57	21
`-p0 0.3 -up0 0.1`	—	—	—	—	32.98	26.53	29.76	40
`-p0 0.3 -up0 1.0`	—	—	—	—	32.98	26.62	29.80	42
`-p0 0.4 -up0 0.1`	—	—	—	—	33.06	26.80	29.93	50
`-p0 0.4 -up0 0.2`	—	—	—	—	32.97	26.81	29.89	50
`-p0 0.5 -up0 0.1`	55.55	55.91	55.73	255	32.88	26.81	29.84	63
`-p0 0.5 -up0 0.5`	55.68	55.94	55.81	260	32.88	26.76	29.82	63
`-p0 0.6 -up0 0.5`	55.74	55.98	**55.86**	298	—	—	—	–
`-p0 0.7 -up0 0.1`	55.64	55.89	55.77	339	—	—	—	–
`-p0 0.7 -up0 0.5`	55.54	55.84	55.69	350	—	—	—	–

Table 5: Jump-to-NULL. The 4th column for each system gives phrase-table size in millions of pairs.

increases of around 0.6 for Hansard and 0.4 for NIST. However, this is achieved at the cost of a large increase in phrase-table size due to the presence of more null-aligned words that are free to join neighbouring phrase pairs. Max-BLEU phrase-table size triples compared to the baseline for the Hansard system, and more than doubles for NIST09.

- The 3rd group of lines tunes `-up0` with `-p0` set to 0. In the legal range $[0, 2]$ for `-up0`, there is very little increase in BLEU, and in fact only a small effect on the phrase tables, judging from the small increase in number of phrase pairs. It would be interesting to change the implementation of `-up0` to allow for larger values equivalent to the larger values of `-p0`.

- The 4th group of lines tunes `-up0` for `-p0` values close to the optimum. This yields no gains over the best `-p0` value of 0.4 for NIST09, but a slight gain using `-p0 0.6 -up0 0.5` for the Hansard system. Hap-

pily, the latter produces a somewhat smaller phrase table than does the pure-p0 optimum. (To again corroborate the negative results for Hansard lexical conditioning from section 3.1, this setting was also run with prev-hidden conditioning. This resulted in a BLEU drop of approximately 0.2.)

3.5 Symmetrized Training

parameters	hans				nist09			
table 2 baseline	54.66	55.61	**55.13**	109	32.38	26.36	**29.37**	21
`-symmetrized liang`	54.21	55.37	54.79	176	30.30	24.20	27.25	38
`-symmetrized liang-variant`	54.14	55.39	54.76	177	30.35	24.18	27.26	38

Table 6: Symmetrized Training. The 4th column for each system gives phrase-table size in millions of pairs.

Table 6 shows the results of symmetrizing the HMM training process using the *Alignment by Agreement* algorithm proposed by Liang et al, as well as a variant developed by Eric Joanis. Both techniques were applied to the best configuration from table 2. Both resulted in a fairly large drop with respect to the baseline (very large for NIST09), so no further experiments were performed with symmetrized models.

4 Results—Phrase Extraction

Phrase-extraction experiments involved fixed HMMs with "best" parametrizations identified in the previous section:

- Hansard: `-newhmm -start-dist -final-dist -anchor -max-jump 20 -alpha 0.0 -lambda 1.0 -p0 0.6 -up0 0.5`
- NIST09: `-newhmm -start-dist -final-dist -anchor -map-tau 500 -alpha 0.0 -lambda 1.0 -p0 0.4 -up0 0.0`

4.1 Minor Parameters

Table 7 shows results for the `-w` parameter (to `gen-jpt-parallel.sh`) that controls the number of ttable translations added to the phrase-table for source/target words that don't have phrase translations (default 1); and for the `-d` parameter that sets the maximum permissible difference in phrase

parameters	hans				nist09			
best HMM	55.74	55.98	55.86	298	33.06	26.85	**29.95**	50
`-d 6`	55.79	55.96	**55.88**	301	33.02	26.88	29.95	50
`-d 6 -w 2`	55.75	55.97	55.86	301	33.06	26.81	29.93	51

Table 7: Minor Phrase-Extraction Parameters. The 4th column for each system gives phrase-table size in millions of pairs.

length for all phrase pairs (default 4, maximum length is 7). The `-d 6` setting gave a very small gain for the Hansard system.

4.2 Tuning `-p0` for Alignment

parameters	hans				nist09			
table 7 baselines	55.79	55.96	**55.88**	298	33.06	26.85	**29.95**	50
`-p0 0.3`	—	—	—	—	32.93	26.61	29.77	37
`-p0 0.4`	—	—	—	—	32.90	26.75	29.83	45
`-p0 0.5`	55.22	55.67	55.44	222	32.82	26.84	29.83	53
`-p0 0.6`	55.22	55.73	55.48	245	32.86	26.82	29.84	62
`-p0 0.7`	55.58	55.83	55.70	273	32.51	26.72	29.62	74
`-p0 0.8`	55.54	55.86	55.70	315	32.38	26.50	29.44	91

Table 8: Tuning `-p0` for Alignment. The 4th column for each system gives phrase-table size in millions of pairs.

Table 8 shows results for various values of the HMM jump-to-null parameter `-p0` for word alignment. In these experiments, `-p0` is set to a low value (0.2) during HMM training, then re-set to the value shown during word alignment. This strategy has been successful in the past, but did not give any gains for either system tested here. A further enhancement would be to use *separate* values of `-p0` for each alignment direction (source-to-target and target-to-source). This was not tested.

4.3 Symmetrized Word-Alignment Algorithms

Table 9 shows results for various alternatives to the default `-IBMOchAligner 3` symmetrized word-alignment algorithm (see `gen_phrase_tables -H` for a description). In general, these algorithms have little effect on BLEU, although in some cases they greatly expand the size of the phrase tables. The

parameters	hans				nist09			
table 7 baselines	55.79	55.96	55.88	298	33.06	26.85	29.95	50
`-a IBMOchAligner 2`	56.33	55.93	56.13	583	31.90	26.06	28.98	362
`-a IBMDiagAligner 2`	56.33	55.88	56.12	616	31.81	26.14	28.97	400
`-a IBMDiagAligner 3`	55.85	55.94	**55.90**	303	33.16	26.77	**29.96**	51
`-a HybridPostAligner 0.70`	—	—	—	—	32.52	26.15	29.33	202
`-a HybridPostAligner 0.80`	—	—	—	—	32.61	26.30	29.46	145
`-a HybridPostAligner 0.85`	55.75	55.56	55.66	297	—	—	—	—
`-a HybridPostAligner 0.90`	55.63	55.70	55.67	255	32.92	26.17	29.54	110
`-a HybridPostAligner 0.95`	—	—	—	—	32.89	26.06	29.48	98

Table 9: Symmetrized Word-Alignment Algorithms. The 4th column for each system gives phrase-table size in millions of pairs.

`IBMDiagAligner 3` option gives tiny gains over the baseline algorithm for both Hansard and NIST09.

5 Conclusion

The main conclusions from these experiments are: 1) lexical conditioning gives surprisingly small gains; 2) using high jump-to-null probabilities improves BLEU score but results in large phrase tables; and 3) the current default symmetrized alignment and phrase-extraction strategies are difficult to improve upon. Table 10 shows the best configurations for each system tested.

parameter	Hansard setting	NIST09 setting
	— HMM params —	
`-newhmm`	yes (not changed)	yes (not changed)
`-p0`	0.6	0.4
`-up0`	0.5	0.0
`-max-jump`	20	0
`-start-dist`	yes	yes
`-final-dist`	yes	yes
`-anchor`	yes	yes
`-word-classes-l1`	no	no
`-condition-on`	no	prev-hidden
`-alpha`	0.0	0.0
`-lambda`	1.0	1.0
`-map-tau`	0	500
`-symmetrized`	no	no
	— phrase-extraction params —	
`-a`	IMB(Och/Diag)Aligner 3	IBM(Och/Diag)Aligner 3
`-w`	1	1
`-m`	7 (not changed)	7 (not changed)
`-d`	6	4
`-p0`	no	no

Table 10: Optimum Parameter Settings