Machine Translation System Combination with Flexible Word Ordering

Kenneth Heafield, Greg Hanneman, Alon Lavie
Language Technologies Institute, Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA 15213, USA
{kheafiel,ghannema,alavie}@cs.cmu.edu

Abstract

We describe a synthetic method for combining machine translations produced by different systems given the same input. One-best outputs are explicitly aligned to remove duplicate words. Hypotheses follow system outputs in sentence order, switching between systems mid-sentence to produce a combined output. Experiments with the WMT 2009 tuning data showed improvement of 2 BLEU and 1 METEOR point over the best Hungarian-English system. Constrained to data provided by the contest, our system was submitted to the WMT 2009 shared system combination task.

1 Introduction

Many systems for machine translation, with different underlying approaches, are of competitive quality. Nonetheless these approaches and systems have different strengths and weaknesses. By offsetting weaknesses with strengths of other systems, combination can produce higher quality than does any component system.

One approach to system combination uses confusion networks (Rosti et al., 2008; Karakos et al., 2008). In the most common form, a skeleton sentence is chosen from among the one-best system outputs. This skeleton determines the ordering of the final combined sentence. The remaining outputs are aligned with the skeleton, producing a list of alternatives for each word in the skeleton, which comprises a confusion network. A decoder chooses from the original skeleton word and its alternatives to produce a final output sentence. While there are a number of variations on this theme, our approach differs fundamentally in that the effective skeleton changes on a per-phrase basis.

Our system is an enhancement of our previous work (Jayaraman and Lavie, 2005). A hypothesis uses words from systems in order, switching between systems at phrase boundaries. Alignments and a synchronization method merge meaning-equivalent output from different systems. Hypotheses are scored based on system confidence, alignment support, and a language model.

We contribute a few enhancements to this process. First, we introduce an alignment-sensitive method for synchronizing available hypothesis extensions across systems. Second, we pack similar partial hypotheses, which allows greater diversity in our beam search while maintaining the accuracy of n-best output. Finally, we describe an improved model selection process that determined our submissions to the WMT 2009 shared system combination task.

The remainder of this paper is organized as follows. Section 2 describes the system with emphasis on our modifications. Tuning, our experimental setup, and submitted systems are described in Section 3. Section 4 concludes.

2 System

The system consists of alignment (Section 2.1) and phrase detection (Section 2.2) followed by decoding. The decoder constructs hypothesis sentences one word at a time, starting from the left. A partially constructed hypothesis comprises:

Word The most recently decoded word. Initially, this is the beginning of sentence marker.

Used The set of used words from each system. Initially empty.

Phrase The current phrase constraint from Section 2.2, if any. The initial hypothesis is not in a phrase.

Features Four feature values defined in Section 2.4 and used in Section 2.5 for beam search

Proceedings of the Fourth Workshop on Statistical Machine Translation , pages 56–60,
Athens, Greece, 30 March – 31 March 2009.

and hypothesis ranking. Initially, all features are 1.

Previous A set of preceding hypothesis pointers described in Section 2.5. Initially empty.

The leftmost unused word from each system corresponds to a continuation of the partial hypothesis. Therefore, for each system, we extend a partial hypothesis by appending that system's leftmost unused word, yielding several new hypotheses. The appended word, and those aligned with it, are marked as used in the new hypothesis. Since systems do not align perfectly, too few words may be marked as used, a problem addressed in Section 2.3. As described in Section 2.4, hypotheses are scored using four features based on alignment, system confidence, and a language model. Since the search space is quite large, we use these partial scores for a beam search, where the beam contains hypotheses of equal length. This space contains hypotheses that extend in precisely the same way, which we exploit in Section 2.5 to increase diversity. Finally, a hypothesis is complete when the end of sentence marker is appended.

2.1 Alignment

Sentences from different systems are aligned in pairs using a modified version of the METEOR (Banerjee and Lavie, 2005) matcher. This identifies alignments in three phases: exact matches up to case, WordNet (Fellbaum, 1998) morphology matches, and shared WordNet synsets. These sources of alignments are quite precise and unable to pick up on looser matches such as "mentioned" and "said" that legitimately appear in output from different systems. Artificial alignments are intended to fill gaps by using surrounding alignments as clues. If a word is not aligned to any word in some other sentence, we search left and right for words that are aligned into that sentence. If these alignments are sufficiently close to each other in the other sentence, words between them are considered for artificial alignment. An artificial alignment is added if a matching part of speech is found. The algorithm is described fully by Jayaraman and Lavie (2005).

2.2 Phrases

Switching between systems is permitted outside phrases or at phrase boundaries. We find phrases in two ways. Alignment phrases are maximally long sequences of words which align, in the same order and without interruption, to a word sequence from at least one other system. Punctuation phrases place punctuation in a phrase with the preceding word, if any. When the decoder extends a hypothesis, it considers the longest phrase in which no word is used. If a punctuation phrase is partially used, the decoder marks the entire phrase as used to avoid extraneous punctuation.

2.3 Synchronization

While phrases address near-equal pieces of translation output, we must also deal with equally meaningful output that does not align. The immediate effect of this issue is that too few words are marked as used by the decoder, leading to duplication in the combined output. In addition, partially aligned system output results in lingering unused words between used words. Often these are function words that, with language model scoring, make output unnecessarily verbose. To deal with this problem, we expire lingering words by marking them as used. Specifically, we consider the frontier of each system, which is the leftmost unused word. If a frontier lags behind, words as used to advance the frontier. Our two methods for synchronization differ in how frontiers are compared across systems and the tightness of the constraint.

Previously, we measured frontiers from the beginning of sentence. Based on this measurement, the synchronization constraint requires that the frontiers of each system differ by at most s. Equivalently, a frontier is lagging if it is more than s words behind the rightmost frontier. Lagging frontiers are advanced until the synchronization constraint becomes satisfied. We found this method can cause problems in the presence of variable length output. When the variability in output length exceeds s, proper synchronization requires distances between frontiers greater than s, which this constraint disallows.

Alignments indicate where words are synchronous. Words near an alignment are also likely to be synchronous even without an explicit alignment. For example, in the fragments "even more serious, you" and "even worse, you" from WMT 2008, "serious" and "worse" do not align but do share relative position from other alignments, suggesting these are synchronous. We formalize this by measuring the relative position of frontiers from alignments on each side. For example,

if the frontier itself is aligned then relative position is zero. For each pair of systems, we check if these relative positions differ by at most s under an alignment on either side. Confidence in a system's frontier is the sum of the system's own confidence plus confidence in systems for which the pair-wise constraint is satisfied. If confidence in any frontier falls below 0.8, the least confident lagging frontier is advanced. The process repeats until the constraint becomes satisfied.

2.4 Scores

We score partial and complete hypotheses using system confidence, alignments, and a language model. Specifically, we have four features which operate at the word level:

Alignment Confidence in the system from which the word came plus confidence in systems to which the word aligns.

Language Model Score from a suffix array language model (Zhang and Vogel, 2006) trained on English from monolingual and French-English data provided by the contest.

N-Gram $\left(\frac{1}{3}\right)^{order-ngram}$ using language model $order$ and length of $ngram$ found.

Overlap $\frac{overlap}{order-1}$ where $overlap$ is the length of intersection between the preceding and current n-grams.

The N-Gram and Overlap features are intended to improve fluency across phrase boundaries. Features are combined using a log-linear model trained as discussed in Section 3. Hypotheses are scored using the geometric average score of each word in the hypothesis.

2.5 Search

Of note is that a word's score is impacted only by its alignments and the n-gram found by the language model. Therefore two partial hypotheses that differ only in words preceding the n-gram and in their average score are in some sense duplicates. With the same set of used words and same phrase constraint, they extend in precisely the same way. In particular, the highest scoring hypothesis will never use a lower scoring duplicate.

We use duplicate detecting beam search to explore our hypothesis space. A beam contains partial hypotheses of the same length. Duplicate hypotheses are detected on insertion and packed, with the combined hypothesis given the highest score of those packed. Once a beam contains the top scoring partial hypotheses of length l, these hypotheses are extended to length $l+1$ and placed in another beam. Those hypotheses reaching end of sentence are placed in a separate beam, which is equivalent to packing them into one final hypothesis. Once we remove partial hypothesis that did not extend to the final hypothesis, the hypotheses are a lattice connected by parent pointers.

While we submitted only one-best hypotheses, accurate n-best hypotheses are important for training as explained in Section 3. Unpacking the hypothesis lattice into n-best hypotheses is guided by scores stored in each hypothesis. For this task, we use an n-best beam of paths from the end of sentence hypothesis to a partial hypothesis. Paths are built by induction, starting with a zero-length path from the end of sentence hypothesis to itself. The top scoring path is removed and its terminal hypothesis is examined. If it is the beginning of sentence, the path is output as a complete hypothesis. Otherwise, we extend the path to each parent hypothesis, adjusting each path score as necessary, and insert into the beam. This process terminates with n complete hypotheses or an empty beam.

3 Tuning

Given the 502 sentences made available for tuning by WMT 2009, we selected feature weights for scoring, a set of systems to combine, confidence in each selected system, and the type and distance s of synchronization. Of these, only feature weights can be trained, for which we used minimum error rate training with version 1.04 of IBM-style BLEU (Papineni et al., 2002) in case-insensitive mode. We treated the remaining parameters as a model selection problem, using 402 randomly sampled sentences for training and 100 sentences for evaluation. This is clearly a small sample on which to evaluate, so we performed two folds of cross-validation to obtain average scores over 200 untrained sentences. We chose to do only two folds due to limited computational time and a desire to test many models.

We scored systems and our own output using case-insensitive IBM-style BLEU 1.04 (Papineni et al., 2002), METEOR 0.6 (Lavie and Agarwal, 2007) with all modules, and TER 5 (Snover et al., 2006). For each source language, we ex-

In	Sync s	BLEU	METE	TER	Systems and Confidences				
cz	length 8	.236	.507	59.1	google .46	cu-bojar .27	uedin .27		
cz	align 5	.226	.499	57.8	google .50	cu-bojar .25	uedin .25		
cz	align 7	.211	.508	65.9	cu-bojar .60	google .20	uedin .20		
cz		*.231*	*.504*	*57.8*	*google*				
de	length 7	.255	.531	54.2	google .40	uka .30	stuttgart .15	umd .15	
de	length 6	.260	.532	55.2	google .50	systran .25	umd .25		
de	align 9	.256	.533	55.5	google .40	uka .30	stuttgart .15	umd .15	
de	align 6	.200	.514	54.2	google .31	uedin .22	systran .18	umd .16	uka .14
de		*.244*	*.523*	*57.5*	*google*				
es	align 8	.297	.560	52.7	google .75	uedin .25			
es	length 5	.289	.548	52.1	google .50	talp-upc .17	uedin .17	rwth .17	
es		*.297*	*.558*	*52.7*	*google*				
fr	align 6	.329	.574	49.9	google .70	lium1 .30			
fr	align 8	.314	.596	48.6	google .50	lium1 .30	limsi1 .20		
fr	length 8	.323	.570	48.5	google .50	lium1 .25	limsi1 .25		
fr		*.324*	*.576*	*48.7*	*google*				
hu	length 5	.162	.403	69.2	umd .50	morpho .40	uedin .10		
hu	length 8	.158	.407	69.5	umd .50	morpho .40	uedin .10		
hu	align 7	.153	.392	68.0	umd .33	morpho .33	uedin .33		
hu		*.141*	*.391*	*66.1*	*umd*				
xx	length 5	.326	.584	49.6	google-fr .61	google-es .39			
xx	align 4	.328	.580	49.5	google-fr .80	google-es .20			
xx	align 5	.324	.576	48.6	google-fr .61	google-es .39			
xx	align 7	.319	.587	51.1	google-fr .50	google-es .50			
xx		*.324*	*.576*	*48.7*	*google-fr*				

Table 1: Combination models used for submission to WMT 2009. For each language, we list our primary combination, contrastive combinations, and a high-scoring system for comparison in italic. All translations are into English. The xx source language combines translations from different languages, in our case French and Spanish. Scores from BLEU, METEOR, and TER are the average of two cross-validation folds with 100 evaluation sentences each. Numbers following system names indicate contrastive systems. More evaluation, including human scores, will be published by WMT.

perimented with various sets of high-scoring systems to combine. We also tried confidence values proportional to various powers of BLEU and METEOR scores, as well as hand-picked values. Finally we tried both variants of synchronization with values of s ranging from 2 to 9. In total, 405 distinct models were evaluated. For each source source language, our primary system was chosen by performing well on all three metrics. Models that scored well on individual metrics were submitted as contrastive systems. In Table 1 we report the models underlying each submitted system.

4 Conclusion

We found our combinations are quite sensitive to presence of and confidence in the underlying systems. Further, we show the most improvement when these systems are close in quality, as is the case with our Hungarian-English system. The two methods of synchronization were surprisingly competitive, a factor we attribute to short sentence length compared with WMT 2008 Europarl sentences. Opportunities for further work include per-sentence system confidence, automatic training of more parameters, and different alignment models. We look forward to evaluation results from WMT 2009.

Acknowledgments

The authors wish to thank Jonathan Clark for training the language model and other assistance. This work was supported in part by the DARPA GALE program and by a NSF Graduate Research Fellowship.

References

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proc. ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Shyamsundar Jayaraman and Alon Lavie. 2005. Multi-engine machine translation guided by explicit word matching. In *Proc. EAMT*, pages 143–152.

Damianos Karakos, Jason Eisner, Sanjeev Khudanpur, and Markus Dreyer. 2008. Machine translation system combination using ITG-based alignments. In *Proc. ACL-08: HLT, Short Papers (Companion Volume)*, pages 81–84.

Alon Lavie and Abhaya Agarwal. 2007. METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments. In *Proc. Second Workshop on Statistical Machine Translation*, pages 228–231, Prague, Czech Republic, June.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evalution of machine translation. In *Proc. 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, PA, July.

Antti-Veikko I. Rosti, Bing Zhang, Spyros Matsoukas, and Richard Schwartz. 2008. Incremental hypothesis alignment for building confusion networks with application to machine translation system combination. In *Proc. Third Workshop on Statistical Machine Translation*, pages 183–186.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proc. Seventh Conference of the Association for Machine Translation in the Americas*, pages 223–231, Cambridge, MA, August.

Ying Zhang and Stephan Vogel. 2006. Suffix array and its applications in empirical natural language processing. Technical Report CMU-LTI-06-010, Language Technologies Institute, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, Dec.