Hierarchical Phrase-based Stream Decoding

Andrew Finch and **Xiaolin Wang** and **Masao Utiyama** and **Eiichiro Sumita**
Advanced Speech Translation Research and Development Promotion Center
Advanced Translation Technology Laboratory
National Institute of Information and Communications Technology
Kyoto, Japan
{andrew.finch,xiaolin.wang,mutiyama,eiichiro.sumita}@nict.go.jp

Abstract

This paper proposes a method for hierarchical phrase-based stream decoding. A stream decoder is able to take a continuous stream of tokens as input, and segments this stream into word sequences that are translated and output as a stream of target word sequences. Phrase-based stream decoding techniques have been shown to be effective as a means of simultaneous interpretation. In this paper we transfer the essence of this idea into the framework of hierarchical machine translation. The hierarchical decoding framework organizes the decoding process into a chart; this structure is naturally suited to the process of stream decoding, leading to an efficient stream decoding algorithm that searches a restricted subspace containing only relevant hypotheses. Furthermore, the decoder allows more explicit access to the word re-ordering process that is of critical importance in decoding while interpreting. The decoder was evaluated on TED talk data for English-Spanish and English-Chinese. Our results show that like the phrase-based stream decoder, the hierarchical is capable of approaching the performance of the underlying hierarchical phrase-based machine translation decoder, at useful levels of latency. In addition the hierarchical approach appeared to be robust to the difficulties presented by the more challenging English-Chinese task.

1 Introduction

Statistical machine translation traditionally operates on sentence segmented input. This technology has advanced to the point where it is becoming capable enough to be useful for many applications. However, this approach may be unsuitable for simultaneous interpretation where the machine translation system is required to provide translations within a reasonably short space of time after words have been spoken. Under this type of constraint, it may not be possible to wait for the end of the sentence before translating, and segmentation at the sub-sentential level may be required as a consequence. This segmentation process is difficult, even for skilled human interpreters, and presents a major challenge to a machine since in addition to the translation process, decisions need to be made about when to commit to outputting a partial translation. Such decisions are critical since once such an output is made it can be difficult and highly undesirable to correct it later if it is in error.

2 Related Work

In order to automatically perform segmentation for interpretation, two types of strategy have be proposed. In the first, which we will call pre-segmentation, the stream is segmented prior to the start of the machine translation decoding process, and the machine translation system is constrained to translate using the given segmentation. This approach has the advantage that it can be implemented without the need to modify the machine translation decoding software. In the second type of strategy, which we will call incremental decoding, the segmentation process is performed during the decoding of the input stream. In this approach the segmentation process is able to exploit segmentation cues arising from the decoding process itself. That is to say, the order in which the decoder would prefer to generate the target sequence is taken into account.

A number of diverse strategies for pre-segmentation were studied in (Sridhar et al., 2013). They studied both non-linguistic techniques, that included fixed-length segments, and a “hold-output” method which identifies contiguous blocks of text that do not contain alignments to words outside them, and linguistically-motivated segmentation techniques beased on segmenting on

Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 1089–1094,
Lisbon, Portugal, 17-21 September 2015. ©2015 Association for Computational Linguistics.

conjunctions, sentence boundaries and commas. Commas were the most effective segmentation cue in their investigation.

In (Oda et al., 2014) a strategy for segmentation prior to decoding based on searching for segmentation points while optimizing the BLEU score was presented. An attractive characteristic of this approach is that the granularity of the segmentation could be controlled by choosing the number of segmentation boundaries to be inserted, prior to the segmentation process. In (Matusov et al., 2007) it was shown that the prediction and use of soft boundaries in the source language text, when used as re-ordering constraints can improve the quality of a speech translation system.

(Siahbani et al., 2014) used a pre-segmenter in combination with a left-to-right hierarchical decoder (Watanabe et al., 2006) to achieve a considerably faster decoder in return for a small cost in terms of BLEU score.

A phrase-based incremental decoder called the stream decoder was introduced in (Kolss et al., 2008b), and further studied in (Finch et al., 2014). Their results, conducted on translation between European languages, and also on English-Chinese, showed that this approach was able to maintain a high level of translation quality for practically useful levels of latency. The hierarchical decoding strategy proposed here is based on this work.

2.1 Stream Decoding

The reader is referred to the original paper (Kolss et al., 2008a) for a complete description of the stream decoding process; in this section we provide a brief summary.

Figure 1 depicts a stream decoding process, and the figure applies to both the original phrase-based technique, and the proposed hierarchical method. The input to the stream decoder is a stream of tokens (it is also possible for the decoder to operate on tuples of confusable token sequences from a speech recognition decoder). As new tokens arrive, states in the search graph are extended with the new possible translation options arising from the new tokens. Periodically the stream decoder will commit to outputting a sequence of target tokens. At this point a state from the search graph is selected, the search graph leading from this state is kept, and the remainder discarded. The search then continues using the pruned search graph. The language model context is preserved at this state for use during the subsequent decoding. In this manner the stream decoder is able to jointly segment and translate a continuous stream of tokens that contains no segment boundary information; the segmentation occurs as a natural by-product of the decoding process. Re-ordering occurs in exactly the same manner as the sentence-by-sentence hierarchical decoder, and word re-ordering within segments is possible.

2.1.1 Latency Parameters

The stream decoding process is governed by two parameters L_{max} and L_{min}. These parameters are illustrated in Figure 1. The L_{max} parameter controls the maximum latency of the system. That is, the maximum number of tokens the system is permitted to fall behind the current position. If interpreting from speech, the parameter represents the number of words the system is allowed to fall behind the speaker, before being required to provide an output translation. This parameter is a hard constraint that guarantees the system will always be within L_{max} tokens of the current last token in the stream of input tokens. The parameter L_{min} represents the minimum number of words the system will lag behind the last word spoken. It serves as a means of preventing the decoder from committing to a translation too early.

Both the phrase-based and hierarchical phrase-based stream decoders maintain a sequence of tokens that represent the sequence of untranslated tokens from the input stream (see Figure 1). As new tokens arrive from the input stream, they are added to the end of the sequence. When the length of this sequence reaches L_{max}, the decoder is forced to provide an output.

2.1.2 Phrase-based Segmentation

When forced to commit to a translation, the phrase-based decoder rolls back the best hypothesis state by state, until the remaining state sequence translates a contiguous sequence of source words starting from beginning of the sequence of untranslated words, and the number of words that would remain in the sequence of untranslated words after the translation is made, is at least L_{min}. It is possible that no such state exists, in which case since the stream decoder is required to make an output, it must use an alternative strategy.

In this alternative strategy, the stream decoder will undertake a new decoding pass in which it is forced to make a monotonic step as the first step in the decoding process. Then, a state is selected from the best hypothesis using the roll-back strategy above. This process may also fail if the monotonic step would lead to the violation of L_{min}. In the implementation of (Finch et al., 2014), the decoder is permitted to violate L_{min} only in this case.

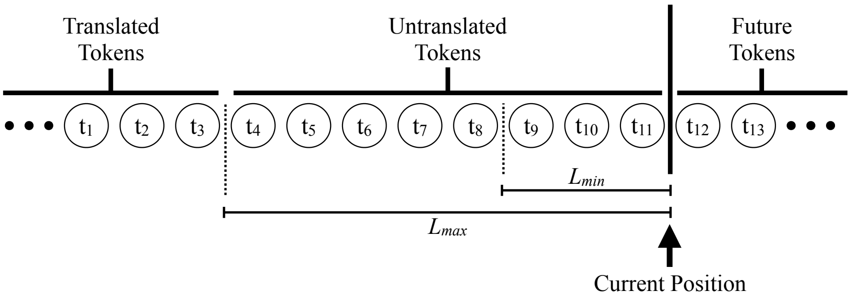


Figure 1: The stream decoding process.

2.1.3 The Proposed Method

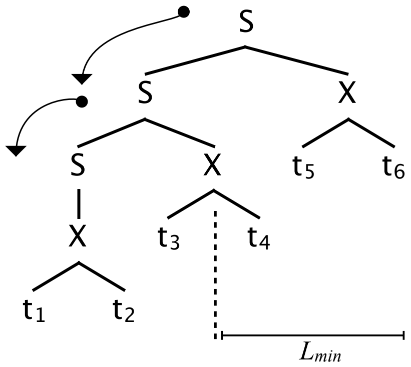


Figure 2: Selecting a segmentation point during hierarchical decoding.

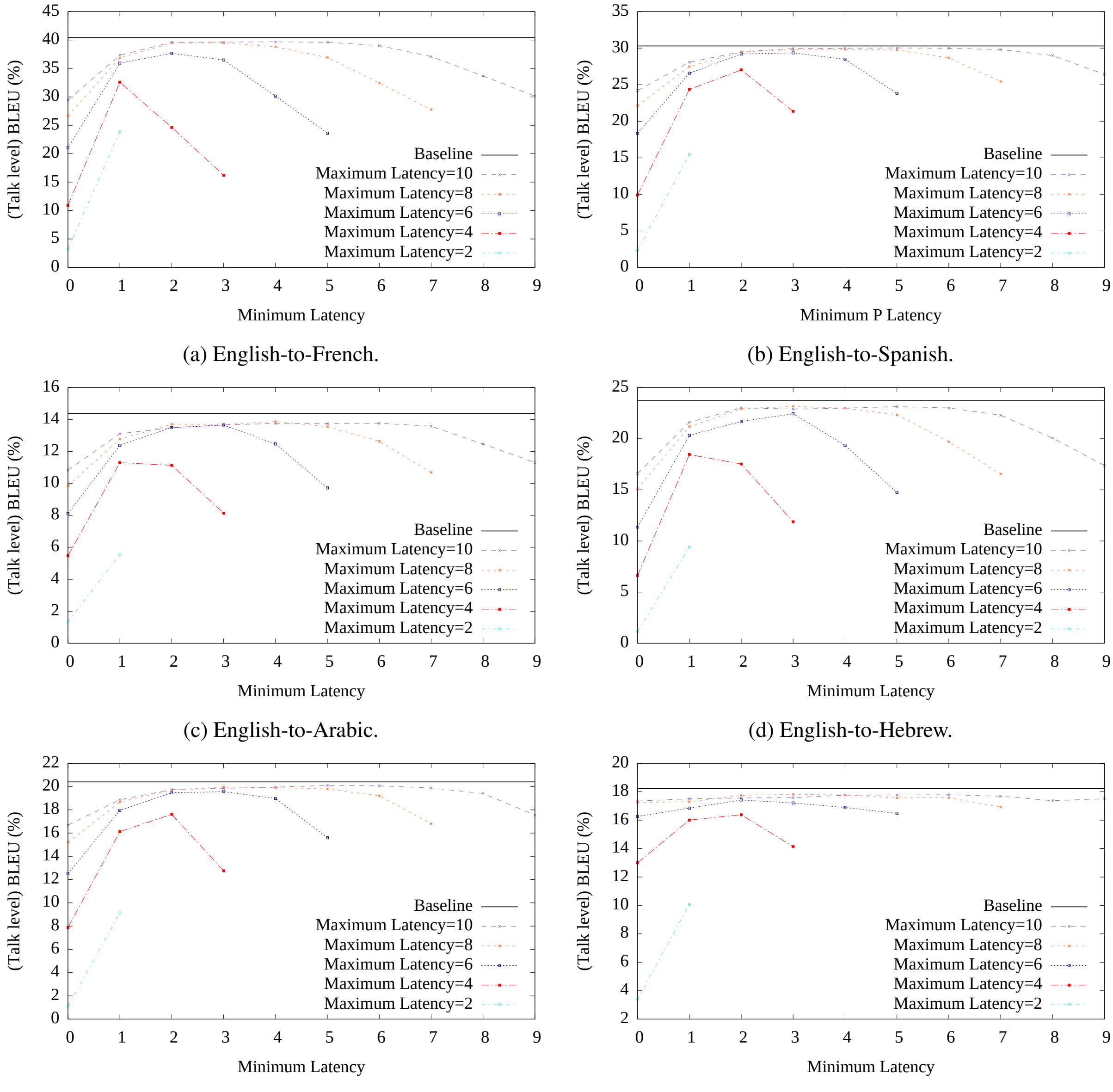
The proposed hierarchical method attempts to capture the spirit of the phrase-based method. When forced to commit to a translation of a sequence of n words, the segmentation process is simple and guided direcly by the chart.

As in the phrase-based approach, the best hypothesis at the top of the chart is used to provide the partial translation and segmentation point. This hypothesis has a span of $[1, n]$ over the source words. The left child of the rule (defined in accordance with the binarized grammar used by the decoder) that was applied to create this hypothesis is examined; let its span be $[1, k]$. If $n - k \geq L_{min}$, then this partial hypothesis represents a translation of the first k words of the sentence that leaves at least L_{min} words untranslated, and therefore the target word sequence from this partial hypothesis is output, and the associated source words are removed from the sequence of untranslated words. If this hypothesis is not able to meet the constraint, the parse tree traversal continues in the same manner: depth first along the left children until either a translation can be made, or no further traversal is possible.

Following the translation of of word sequence, similar to the phrase-based stream decoder of (Finch et al., 2014), the hierarchical stream decoder proceeds from an initial state in which the language model context is preserved. The decoding process relies on an implicit application of the glue grammar to connect the past and future nodes. An visual example of this selection process is given in Figure 2. In this example, the neither the root of the tree (spanning $t_1t_2t_3t_4t_5t_6$) nor its left child (spanning $t_1t_2t_3t_4$) are not able to generate an output since they both span sequences of words that would violate L_{min}, which is 3 in this example. The left child two levels down from the root node spans only t_1t_2 and would leave 4 words untranslated, therefore it defines an acceptable segmentation point.

Instead of forcing a monotonic decoding step in the event of a failure to find a segmentation point during the decoding, the hierarchical stream decoder directly eliminates hypotheses that would lead to such a failure. The search process is constrained such that all parse trees that cover the first word of the source sentence, must contain a subtree that can give rise to a translation that does not violate L_{min} (constituents that can produce translations cannot span more than $L_{max} - L_{min}$ words). Any search state that would violate this constraint is not allowed to enter the chart. This property is recursively propagated up the chart during the parsing process ensuring that each entry placed into the first column of the chart contains a constituent that could be used to produce a translation.

This approach is more appealing than the forced monotonic step in that it will also allow non-monotonic translations that are guaranteed to be usable. Similar to the phrase-based approach, in some circumstances it may not be possible to produce a parse that does not violate L_{min}, and only in this rare case is the decoder allowed to violate L_{min} in order to guarantee maximum latency.



(a) English-to-French.

(b) English-to-Spanish.

(c) English-to-Arabic.

(d) English-to-Hebrew.

(e) English-to-Russian.

(f) English-to-Chinese.

Figure 3: Stream decoding performance for several language pairs. The baseline was the same hierarchical phrase-based decoder, but decoded in the usual manner sentence-by-sentence without the stream decoding process. The baseline used the sentence segmentation provided by the corpus.

3 Experiments

3.1 Corpora

In all experiments, we used the TED[1] talks data sets from the IWSLT2014 campaign. We evaluated on English-to-Spanish, and English-to-Chinese translation using the same data sets that were used in (Finch et al., 2014). These pairs were chosen to include language pairs with a relatively monotonic translation process (English-Spanish) and (English-French), and also language pairs that required a greater amount of word re-ordering for example (English-Chinese). The Chinese corpus was segmented using the Stanford Chinese word segmenter (Tseng et al., 2005) according to the Chinese Penn Treebank standard.

3.2 Experimental Methodology

Our stream decoder was implemented within the framework of the AUGUSTUS decoder, a hierarchical statistical machine translation decoder (Chiang, 2007) that operates in a similar manner to the moses-chart decoder provided in the Moses machine translation toolkit (Koehn et al., 2007). The training procedure was quite typical: 5-gram language models were used, trained with modified

[1] http://www.ted.com

English input stream:

```
 ... we want to encourage a world of creators of inventors
of contributors because this world that we live in this
interactive world is ours ...
```

Sequence of translated segments:

Segment 1:	queremos	[we want to]
Segment 2:	animar a un mundo de	[encourage a world of]
Segment 3:	creadores de inventores	[creators of inventors]
Segment 4:	de colaboradores	[of collaborators]
Segment 5:	porque este mundo	[because this world]
Segment 6:	en el que vivimos	[in which we live]
Segment 7:	este interactiva mundo	[this interactive world]
Segment 8:	es la nuestra	[is ours]

Figure 4: Example translation segmentation from the English-Spanish task ($L_{max} = 8$ and $L_{min} = 4$).

Kneser-Ney smoothing; MERT (Och, 2003) was used to train the log-linear weights of the models; the decoding was performed with a distortion limit of 20 words.

To allow the results to be directly comparable to those in (Finch et al., 2014), the talk level BLEU score (Papineni et al., 2001) was used to evaluate the machine translation quality in all experiments.

3.3 Results

The results for decoding with various values of the latency parameters are shown in Figure 3 for English-French, English-Spanish, English-Arabic, English-Hebrew, English-Russian and English-Chinese. Overall the behavior of the system was quite similar in character to the published results for phrase-based stream decoding for English-Spanish (Kolss et al., 2008b; Finch et al., 2014). The hierarchical system seemed to be more sensitive to small values of minimum latency, and less sensitive to larger values. The results for the more challenging English-Chinese pair were more surprising. In (Finch et al., 2014), the performance of the phrase-based decoder suffered as expected in comparison to pairs of European languages. This was in line with the increase in difficulty of the task due to word order differences. However, in comparison to prior results published on the phrase-based stream decoder, the hierarchical stream decoder seems less affected by the differences between these languages; the curves are higher at the optimal values of minimum latency, and seem less sensitive to its value. The character of the results appears to be very similar to those from English-Spanish. This result is encouraging and suggests that the hierarchical method may be better suited to interpreting between the more difficult language pairs. Figure 4 shows the segmentation given by the system with $L_{max} = 8$ and $L_{min} = 4$, on a sequence of English words which is a subsequence of an unseen test stream of words being decoded.

4 Conclusion

In this paper we propose and evaluate the first hierarchical phrase-based steam decoder. The standard hierarchical phrase-based decoding process generates from the source in left-to-right order, making it naturally suited for incremental decoding. The hierarchical decoder organizes the search process in a chart which can be directly exploited to perform stream decoding. The proposed hierarchical stream decoding process only searches a subset of the search space that is capable of generating useful partial translation hypothesis. This eliminates the necessity for the forced monotonic step necessary in the phrase-based counterpart. Hypotheses that are not useful are discarded, and are therefore not able to compete with useful hypotheses in the search. Additionally, a beneficial side-effect of the pruning of the search space is that decoding speed increased by a factor of approximately 8 over the baseline sentence-by-sentence decoder. Looking to the future, one important benefit of taking a hierarchical approach is that the re-ordering process is made explicit, and in further research we wish to explore the possibility of introducing of new interpretation-oriented rules into the stream decoding process.

Acknowledgements

We would like to thank the reviewers for their valuable comments.

References

David Chiang. 2007. Hierarchical phrase-based translation. *Computational Linguistics*, 33(2):201–228.

Andrew Finch, Xiaolin Wang, and Eiichiro Sumita. 2014. An Exploration of Segmentation Strategies in Stream Decoding. In *Proceedings of the International Workshop on Spoken Language Translation (IWSLT)*, pages 139–142, South Lake Tahoe, USA.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowa, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL 2007): demo and poster sessions*, pages 177–180, Prague, Czeck Republic, June.

Muntsin Kolss, Stephan Vogel, and Alex Waibel. 2008a. Stream decoding for simultaneous spoken language translation. In *Proceedings of Interspeech*, pages 2735–2738, Brisbane, Australia.

Muntsin Kolss, Matthias Wölfel, Florian Kraft, Jan Niehues, Matthias Paulik, and Alex Waibel. 2008b. Simultaneous German-English lecture translation. In *Proceedings of the International Workshop on Spoken Language Translation (IWSLT)*, pages 174–181, Waikiki , Hawai'i, USA.

Evgeny Matusov, Dustin Hillard, Mathew Magimai-Doss, Dilek Hakkani-Tur, Mari Ostendorf, and Hermann Ney. 2007. Improving speech translation with automatic boundary prediction. In *Proceedings of Interspeech*, pages 2449–2452, Antwerp.

Franz J. Och. 2003. Minimum error rate training for statistical machine translation. In *Proceedings of the 41st Meeting of the Association for Computational Linguistics (ACL 2003)*, volume 1, pages 160–167, Sapporo, Japan.

Yusuke Oda, Graham Neubig, Sakriani Sakti Tomoki Toda, and Satoshi Nakamura. 2014. Optimizing segmentation strategies for simultaneous speech translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, Baltimore, USA, June. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2001. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318, Morristown, NJ, USA. Association for Computational Linguistics.

Maryam Siahbani, Ramtin Mehdizadeh Seraj, Baskaran Sankaran, and Anoop Sarkar. 2014. Incremental translation using hierarchichal phrase-based translation system. In *Spoken Language Technology Workshop (SLT), 2014 IEEE*, pages 71–76. IEEE.

Vivek Kumar Rangarajan Sridhar, John Chen, Srinivas Bangalore, Andrej Ljolje, and Rathinavelu Chengalvarayan. 2013. Segmentation strategies for streaming speech translation. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics – Human Language Technologies (HLT-NAACL)*, pages 230–238, Atlanta, USA.

Huihsin Tseng, Pichuan Chang, Galen Andrew, Daniel Jurafsky, and Christopher Manning. 2005. A conditional random field word segmenter for sighan bakeoff 2005. In *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing*, volume 171. Jeju Island, Korea.

Taro Watanabe, Hajime Tsukada, and Hideki Isozaki. 2006. Left-to-right target generation for hierarchical phrase-based translation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, ACL-44, pages 777–784, Stroudsburg, PA, USA. Association for Computational Linguistics.