Mining Wikipedia Revision Histories for Improving Sentence Compression

Elif Yamangil   **Rani Nelken**
School of Engineering and Applied Sciences
Harvard University
Cambridge, MA 02138, USA
{elif,nelken}@eecs.harvard.edu

Abstract

A well-recognized limitation of research on supervised sentence compression is the dearth of available training data. We propose a new and bountiful resource for such training data, which we obtain by mining the revision history of Wikipedia for sentence compressions and expansions. Using only a fraction of the available Wikipedia data, we have collected a training corpus of over 380,000 sentence pairs, two orders of magnitude larger than the standardly used Ziff-Davis corpus. Using this newfound data, we propose a novel lexicalized noisy channel model for sentence compression, achieving improved results in grammaticality and compression rate criteria with a slight decrease in importance.

1 Introduction

With the increasing success of machine translation (MT) in recent years, several researchers have suggested transferring similar methods for monolingual text rewriting tasks. In particular, Knight and Marcu (2000) (KM) applied a channel model to the task of *sentence compression* – dropping words from an individual sentence while retaining its important information, and without sacrificing its grammaticality. Compressed sentences can be useful either on their own, e.g., for subtitles, or as part of a larger summarization or MT system. A well-recognized problem of this approach, however, is data sparsity. While bilingual parallel corpora are abundantly available, monolingual parallel corpora, and especially collections of sentence compressions are van-

ishingly rare. Indeed, most work on sentence compression has used the Ziff-Davis corpus (Knight and Marcu, 2000), which consists of a mere 1067 sentence pairs. While data sparsity is a common problem of many NLP tasks, it is much more severe for sentence compression, leading Turner and Charniak (2005) to question the applicability of the channel model for this task altogether.

Our contribution in this paper is twofold. First, we solve the data sparsity issue by showing that abundant sentence compressions can be extracted from Wikipedia's revision history. Second, we use this data to validate the channel model approach for text compression, and improve upon it by creating a novel fully lexicalized compression model. Our model improves grammaticality and compression rate with only a slight decrease in importance.

2 Data: Wikipedia revision histories as a source of sentence compressions

Many researchers are increasingly turning to Wikipedia as a large-scale data source for training NLP systems. The vast majority of this work uses only the most recent version of the articles. In fact, Wikipedia conveniently provides not only the latest version, but the entire revision history of each of its articles, as dramatically visualized by Viégas et al. (2004). Through Wikipedia's collaborative editing process, articles are iteratively amended and refined by multiple Web users. Users can usually change any aspect of the document's structure and content, but for our purposes here, we focus only on sentence-level edits that add or drop words.

We have downloaded the July snapshot of the

English Wikipedia, consisting of 1.4 million articles, and mined a subset of them for such compressions/expansions. We make the simplifying assumption that *all* such edits also retain the core meaning of the sentence, and are therefore valid training data for our purposes. This assumption is of course patently naïve, as there are many cases in which such revisions reverse sentence meaning, add or drop essential information, are part of a flame war, etc. Classifying these edits is an interesting task which we relegate to future work.[1]

From about one-third of the snapshot, we extracted over 380,000 sentence pairs, which is 2 orders of magnitude more than the Ziff-Davis corpus.[2] Wikipedia currently has 2.3 million articles and is constantly expanding. We can therefore expect an increase of another order of magnitude. We thus can afford to be extremely selective of the sentence pairs we use. To handle a dataset of such size (hundreds of GBs), we split it into smaller chunks, and distribute all the processing.

More technically, for each article, we first extract all revisions, and split each revision into a list of its sentences. We run an edit-distance comparison between each such pair, treating each sentence as an atomic "letter". We look for all replacements of one sentence by another and check whether one sentence is a compression of the other.[3] We then run Collins' parser (1997), using just the sentence pairs where parsing succeeds with a negative log likelihood below 200.

3 Noisy channel model

We follow KM in modeling the problem using a generative noisy channel model, but use the new-found training data to lexicalize the model. Sentences start their life in short form, s, are ranked by a source language model, $p(s)$, and then probabilistically expanded to form the long sentence, $p(l|s)$. During decoding, given a long sentence, we seek the most likely short sentence that could have generated it.

[1]For instance, compressions are more likely to signal optional information than expansions; the lexical items added are likely to be indicative of the type of edit, etc.

[2]The sentence pair corpus is available by contacting the authors.

[3]We ignore word reorderings or replacements that are beyond word addition or deletion.

Using Bayes' rule, this is equivalent to seeking the short sentence s that maximizes $p(s) \cdot p(l|s)$.

3.1 Lexicalized channel model

KM's original model was purely syntax-based. Daume et al. (2002) used a lexicalized PCFG to rerank the compressions, showing that the addition of lexical information helps eliminate improbable compressions. Here, we propose to enhance lexicalization by including lexical information within the channel model, allowing us to better model which compressions are likely and which are not. A minimal example pair illustrating the utility of lexicalization is the following.

(1) Hillary *barely* won the primaries.

(2) Hillary *almost* won the primaries.

The validity of dropping the adverbial here clearly depends on the lexical value of the adverb. It is more acceptable to drop the adverb in Sentence 1, since dropping it in Sentence 2 reverses the meaning. We learn probabilities of the form:
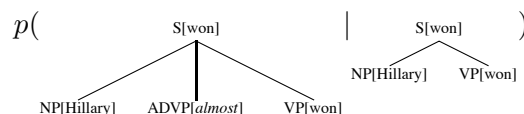
Our model has the power of making compression decisions based on lexical dependencies between the compressed and retained parts of the parse tree.

Note that Daume et al.'s reranking model cannot achieve this type of distinction, since it is based on reranking the compressed version, at which point the adverb is no longer available.

Since we are interested not only in learning how to compress, but also when to compress, we also include in this procedure unchanged CFG rule pairs that are attested in the corpus. Thus, different ways of expanding a CFG rule compete with each other as well as the possibility of not doing any expansion.

3.2 Smoothing

In order to smooth our estimates we use Witten-Bell discounting (1991) with 6 levels of back-off. This method enables us to tune the confidence parameter associated with an estimate inversely proportionally with the diversity of the context of the estimate. The different levels are illustrated in Table 1. Level 1,

the most specific level, is fully lexicalized. Transitioning to levels 2 to 4, we lose the lexical information about the subtrees that are not dropped, the head child bearing subtree, and the dropped subtrees, respectively. At level 4, we end up with the non-lexicalized estimates that are equivalent to KM's model. In subsequent back off levels, we abstract away from the CFG rules. In particular, level 5 estimates the probability of dropping subtrees in the context of a certain parent and head child, and level 6 estimates the probability of the same outcome in the coarser context of a parent only.

3.3 Source model

In addition to the lexicalized channel model, we also use a lexicalized probabilistic syntax-based source model, which we train from the parser's output on the short sentences of each pair.

3.4 Decoding

We implemented the forest-based statistical sentence generation method of Langkilde (2000). KM tailored this method to sentence compression, compactly encoding all compressions of a sentence in a forest structure. The forest ranking algorithm which extracts compressed parse trees, optimized the model scores as well as an additional bigram score. Since our model is lexicalized, the bigram scores become less relevant, which was confirmed by experimentation during development. Therefore in our implementation we exclude the bigram scores and other related aspects of the algorithm such as pruning of bigram-suboptimal phrases.

4 Evaluation

We evaluated our system using the same method as KM, using the same 32 sentences taken from the Ziff-Davis corpus. We solicited judgments of *importance* (the value of the retained information), and *grammaticality* for our compression, the KM results, and human compressions from 8 judges, on a scale of 1 (worst) to 5 (best). Mean and standard deviation are shown in Table 2. Our model improves grammaticality and compression rate criteria with only a slight decrease in importance. Here are some illustrative examples, with the deleted material shown in brackets:

(3) The chemical etching process [used for glare protection] is effective and will help if your office has the fluorescent-light overkill [that 's typical in offices].

(4) Prices range from $5,000 [for a microvax 2000] to $179,000 [for the vax 8000 or higher series].

We suspect that the decrease in importance stems from our indiscriminative usage of compressions and expansions to train our system. We hypothesize that in Wikipedia, expansions often add more useful information, as opposed to compressions which are more likely to drop superfluous or erroneous information.[4] Further work is required to classify sentence modifications.

Since one of our model's back-off levels simulates KM's model, we plan to perform an additional comparative evaluation of both models trained on the same data.

5 Discussion and future work

Turner and Charniak (2005) question the viability of a noisy channel model for the sentence compression task. Briefly put, in the typically sparse data setting, there is no way to distinguish between the probability of a sentence as a short sentence and its probability as a regular sentence of English. Furthermore, the channel model is likely to prefer to leave sentences intact, since that is the most prevalent pattern in the training data. Thus, they argue, the channel model is not really compressing, and it is only by virtue of the length penalty that anything gets shortened at all. Our hope here is that by using a far richer source of short sentences, as well as a huge source of compressions, we can overcome this problem. The noisy channel model posits a virtual competition on each word of coming either from the source model (in which case it is retained in the compression) or from the channel model (in which case it is dropped). By having access to a large data set for the first time, we hope to be able to learn which parts of the sentence are more likely to come from

---

[4]For instance, here is an expansion seen in the data, where the added information (italicized) is important: "In 1952 and 1953 he was stationed in Sendai, Japan during the Korean War *and was shot.*" It would be undesirable to drop this added phrase.

Back-off level	expanded	short
1	S[won] → NP[Hillary] ADVP[almost] VP[won]	S[won] → NP[Hillary] VP[won]
2	S[won] → NP ADVP[almost] VP[won]	S[won] → NP VP[won]
3	S → NP ADVP[almost] VP	S → NP VP
4	S → NP ADVP VP	S → NP VP
5	parent = S, head-child = VP, child = ADVP	parent = S, head-child = VP
6	parent = S, child = ADVP	parent = S

Table 1: Back off levels

	KM	Our model	Humans
Compression	72.91%	67.38%	53.33%
Grammaticality	4.02±1.03	4.31±0.78	4.78±0.17
Importance	3.86±1.09	3.65±1.07	3.90±0.58

Table 2: Evaluation results

which of the two parts of the model. Further work is required in order to clarify this point.

Naturally, discriminative models such as McDonald (2006) are also likely to improve by using the added data. We leave the exploration of this topic for future work.

Finally, we believe that the Wikipedia revision history offers a wonderful resource for many additional NLP tasks, which we have begun exploring.

Acknowledgments

References

Michael Collins. 1997. Three generative, lexicalized models for statistical parsing. In Philip R. Cohen and Wolfgang Wahlster, editors, *Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, pages 16–23, Somerset, New Jersey. Association for Computational Linguistics.

H. Daume, Kevin Knight, I Langkilde-Geary, Daniel Marcu, and K Yamada. 2002. The importance of lexicalized syntax models for natural language generation tasks. Proceedings of the Second International Conference on Natural Language Generation. Arden House, NJ, July 1-3.

Kevin Knight and Daniel Marcu. 2000. Statistics-based summarization - step one: Sentence compression. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, pages 703–710. AAAI Press / The MIT Press.

Irene Langkilde. 2000. Forest-based statistical sentence generation. In *Proceedings of the first conference on North American chapter of the Association for Computational Linguistics*, pages 170–177, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Ryan T. McDonald. 2006. Discriminative sentence compression with soft syntactic evidence. In *Proceedings of EACL 2006, 11st Conference of the European Chapter of the Association for Computational Linguistics, April 3-7, 2006, Trento, Italy*, pages 297–304.

Jenine Turner and Eugene Charniak. 2005. Supervised and unsupervised learning for sentence compression. In *ACL '05: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 290–297, Morristown, NJ, USA. Association for Computational Linguistics.

Fernanda B. Viégas, Martin Wattenberg, and Kushal Dave. 2004. Studying cooperation and conflict between authors with *istory flow* visualizations. In Elizabeth Dykstra-Erickson and Manfred Tscheligi, editors, *CHI*, pages 575–582. ACM.

I. Witten and T. Bell. 1991. The zero-frequency problem: Estimating the probabilities of novel events in adaptive text compression. *IEEE Transactions on Information Theory*, 37(4).