Time-Efficient Creation of an Accurate Sentence Fusion Corpus

Kathleen McKeown, Sara Rosenthal, Kapil Thadani and **Coleman Moore**
Columbia University
New York, NY 10027, USA
{kathy,sara,kapil}@cs.columbia.edu, cjm2140@columbia.edu

Abstract

Sentence fusion enables summarization and question-answering systems to produce output by combining fully formed phrases from different sentences. Yet there is little data that can be used to develop and evaluate fusion techniques. In this paper, we present a methodology for collecting fusions of similar sentence pairs using Amazon's Mechanical Turk, selecting the input pairs in a semi-automated fashion. We evaluate the results using a novel technique for automatically selecting a representative sentence from multiple responses. Our approach allows for rapid construction of a high accuracy fusion corpus.

1 Introduction

Summarization and question-answering systems must transform input text to produce useful output text, condensing an input document or document set in the case of summarization and selecting text that meets the question constraints in the case of question answering. While many systems use sentence extraction to facilitate the task, this approach risks including additional, irrelevant or non-salient information in the output, and the original sentence wording may be inappropriate for the new context in which it appears. Instead, recent research has investigated methods for generating new sentences using a technique called *sentence fusion* (Barzilay and McKeown, 2005; Marsi and Krahmer, 2005; Filippova and Strube, 2008) where output sentences are generated by fusing together portions of related sentences.

While algorithms for automated fusion have been developed, there is no corpus of human-generated fused sentences available to train and evaluate such systems. The creation of such a dataset could provide insight into the kinds of fusions that people produce. Furthermore, since research in the related task of sentence compression has benefited from the availability of training data (Jing, 2000; Knight and Marcu, 2002; McDonald, 2006; Cohn and Lapata, 2008), we expect that the creation of this corpus might encourage the development of supervised learning techniques for automated sentence fusion.

In this work, we present a methodology for creating such a corpus using Amazon's Mechanical Turk[1], a widely used online marketplace for crowdsourced task completion. Our goal is the generation of accurate fusions between pairs of sentences that have some information in common. To ensure that the task is performed consistently, we abide by the distinction proposed by Marsi and Krahmer (2005) between *intersection* fusion and *union* fusion. Intersection fusion results in a sentence that contains only the information that the sentences had in common and is usually shorter than either of the original sentences. Union fusion, on the other hand, results in a sentence that contains all information content from the original two sentences. An example of intersection and union fusion is shown in Figure 1.

We solicit multiple annotations for both union and intersection tasks separately and leverage the different responses to automatically choose a representative response. Analysis of the responses shows that our approach yields 95% accuracy on the task of union fusion. This is a promising first step and indicates that our methodology can be applied towards efficiently building a highly accurate corpus for sentence fusion.

[1] https://www.mturk.com

1. Palin actually turned against the bridge project only after it became a national symbol of wasteful spending.
2. Ms. Palin supported the bridge project while running for governor, and abandoned it after it became a national scandal.
Intersection: Palin turned against the bridge project after it became a national scandal.
Union: Ms. Palin supported the bridge project while running for governor, but turned against it when it became a national scandal and a symbol of wasteful spending.

Figure 1: Examples of intersection and union

2 Related Work

The combination of fragments of sentences on a common topic has been studied in the domain of single document summarization (Jing, 2000; Daumé III and Marcu, 2002; Xie et al., 2008). In contrast to these approaches, sentence fusion was introduced to combine fragments of sentences with common information for multi-document summarization (Barzilay and McKeown, 2005). Automated fusion of sentence pairs has since received attention as an independent task (Marsi and Krahmer, 2005; Filippova and Strube, 2008). Although generic fusion of sentence pairs based on importance does not yield high agreement when performed by humans (Daumé III and Marcu, 2004), fusion in the context of a query has been shown to produce better agreement (Krahmer et al., 2008). We examine similar fusion annotation tasks in this paper, but we asked workers to provide two specific types of fusion, intersection and union, thus avoiding the less specific definition based on importance. Furthermore, as our goal is the generation of corpora, our target for evaluation is *accuracy* rather than agreement.

This work studies an approach to the automatic construction of large fusion corpora using workers through Amazon's Mechanical Turk service. Previous studies using this online task marketplace have shown that the collective judgments of many workers are comparable to those of trained annotators on labeling tasks (Snow et al., 2008) although these judgments can be obtained at a fraction of the cost and effort. However, our task presents an additional challenge: building a corpus for sentence fusion requires workers to enter free text rather than simply choose between predefined options; the results are prone to variation and this makes comparing and aggregating multiple responses problematic.

A. After a decade on the job, Gordon had become an experienced cop.
B. Gordon has a lot of experience in the police force.

Figure 2: An example of sentences that were judged to be too similar for inclusion in the dataset

3 Collection Methodology

Data collection involved the identification of the types of sentence pairs that would make suitable candidates for fusion, the development of a system to automatically identify good pairs and manual filtering of the sentence pairs to remove erroneous choices. The selected sentence pairs were then presented to workers on Mechanical Turk in an interface that required them to manually type in a fused sentence (intersection or union) for each case.

Not all pairs of related sentences are useful for the fusion task. When sentences are too similar, the result of fusion is simply one of the input sentences. For example (Fig. 2), if sentence A contains all the information in sentence B but not vice versa, then B is also their intersection while A is their union and no sentence generation is required. On the other hand, if the two sentences are too dissimilar, then no intersection is possible and the union is just the conjunction of the sentences.

We experimented with different similarity metrics aimed at identifying pairs of sentences that were inappropriate for fusion. The sentences in this study were drawn from clusters of news articles on the same event from the Newsblaster summarization system (McKeown et al., 2002). While these clusters are likely to contain similar sentences, they will contain many more dissimilar than similar pairs and thus a metric that emphasizes precision over recall is important. We computed pairwise similarity between sentences within each cluster using three standard metrics: word overlap, n-gram overlap and cosine similarity. Bigram overlap yielded the best precision in our experiments. We empirically arrived at a lower threshold of .35 to remove dissimilar sentences and an upper threshold of .65 to avoid near-identical sentences, yielding a false-positive rate of 44.4%. The remaining inappropriate pairs were then manually filtered. This semi-automated procedure enabled fast selection of suitable sentence pairs: one person was able to select 30 pairs an hour yielding the 300 pairs for the full experiment in ten hours.

Responses	Intersection	Union
All (1500)	0.49	0.88
Representatives (300)	0.54	0.95

Table 1: Union and intersection accuracy

3.1 Using Amazon's Mechanical Turk

Based on a pilot study with 20 sentence pairs, we designed an interface for the full study. For intersection tasks, the interface posed the question "*How would you combine the following two sentences into a single sentence conveying only the information they have in common?*". For union tasks, the question was "*How would you combine the following two sentences into a single sentence that contains ALL of the information in each?*".

We used all 300 pairs of similar sentences for both union and intersection and chose to collect five worker responses per pair, given the diversity of responses that we found in the pilot study. This yielded a total of 3000 fused sentences with 1500 intersections and 1500 unions.

3.2 Representative Responses

Using multiple workers provides little benefit unless we are able to harness the collective judgments of their responses. To this end, we experiment with a simple technique to select one representative response from all responses for a case, hypothesizing that such a response would have a lower error rate. We test the hypothesis by comparing the accuracy of representative responses with the average accuracy over all responses.

Our strategy for selecting representatives draws on the common assumption used in human computation that human agreement in independently-generated labels implies accuracy (von Ahn and Dabbish, 2004). We approximate agreement between responses using a simple and transparent measure for overlap: cosine similarity over stems weighted by *tf-idf* where *idf* values are learned over the Gigawords corpus[2]. After comparing all responses in a pairwise fashion, we need to choose a representative response. As using the centroid directly might not be robust to the presence of erroneous responses, we first select the pair of responses with the greatest overlap as *candidates* and then choose the candidate which has the greatest total overlap with all other responses.

[2]LDC Catalog No. LDC2003T05

Errors	Intersection	Union
Missing clause	2	7
Union/Intersection	46	6
S1/S2	21	8
Additional clause	10	1
Lexical	3	1

Table 2: Errors seen in 30 random cases (150 responses)

4 Results and Error Analysis

For evaluating accuracy, fused sentences were manually compared to the original sentence pairs. Due to the time-consuming nature of the evaluation, 50% of the 300 cases were randomly selected for analysis. 10% were initially analyzed by two of the authors; if a disagreement occurred, the authors discussed their differences and came to a unified decision. The remaining 40% were then analyzed by one author. In addition to this high-level analysis, we further analyzed 10% of the cases to identify the the types of errors made in fusion as well as the techniques used and the effect of task difficulty on performance.

The accuracy for intersection and union tasks is shown in Table 1. For both tasks, accuracy of the selected representatives significantly exceeded the average response accuracy. In our error analysis, we found that workers often answered the intersection task by providing a union, possibly due to a misinterpretation of the question. This caused intersection accuracy to be significantly worse than union. We analyzed the impact of this error by computing accuracy on the first 30 cases (10%) without this error and the accuracy for intersection increased 22%.

Error types were categorized as "missing clause", "using union for intersection and vice versa", "choosing an input sentence (S1/S2)", "additional clause" and "lexical error". Table 2 shows the number of occurrences of each in 10% of the cases.

We binned the sentence pairs according to the difficulty of the fusion task for each pair (easy/medium/hard) and found that performance was not dependent on difficulty level; accuracy was relatively similar across bins. We also observed that workers typically performed fusion by selecting one sentence as a base and removing clauses or merging in additional clauses from the other sentence.

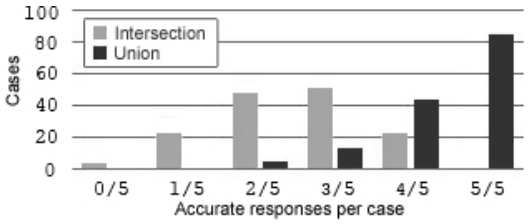

Figure 3: Number of cases in which $x/5$ workers provided accurate responses for fusion

In order to determine the benefit of using many workers, we studied the number of workers who answered correctly for each case. Figure 3 reveals that 2/5 or more workers (summing across columns) responded accurately in 99% of union cases and 82% of intersection cases. The intersection results are skewed due to the question misinterpretation issue which, though it was the most common error, was made by 3/5 workers only 17% of the time. Thus, in the majority of the cases, accurate fusions can still be found using the representative method.

5 Conclusion

We presented a methodology to build a fusion corpus which uses semi-automated techniques to select similar sentence pairs for annotation on Mechanical Turk[3]. Additionally, we showed how multiple responses for each fusion task can be leveraged by automatically selecting a representative response. Our approach yielded 95% accuracy for union tasks, and while intersection fusion accuracy was much lower, our analysis showed that workers sometimes provided unions instead of intersections and we suspect that an improved formulation of the question could lead to better results. Construction of the fusion dataset was relatively fast; it required only ten hours of labor on the part of a trained undergraduate and seven days of active time on Mechanical Turk.

Acknowledgements

This material is based on research supported in part by the U.S. National Science Foundation (NSF) under IIS-05-34871 Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

[3]The corpus described in this work is available at `http://www.cs.columbia.edu/~kathy/fusioncorpus`

References

Regina Barzilay and Kathleen R. McKeown. 2005. Sentence fusion for multidocument news summarization. *Computational Linguistics*, 31(3):297–328.

Trevor Cohn and Mirella Lapata. 2008. Sentence compression beyond word deletion. In *Proceedings of COLING*, pages 137–144.

Hal Daumé III and Daniel Marcu. 2002. A noisy-channel model for document compression. In *Proceedings of ACL*, pages 449–456.

Hal Daumé III and Daniel Marcu. 2004. Generic sentence fusion is an ill-defined summarization task. In *Proceedings of the ACL Text Summarization Branches Out Workshop*, pages 96–103.

Katja Filippova and Michael Strube. 2008. Sentence fusion via dependency graph compression. In *Proceedings of EMNLP*, pages 177–185.

Hongyan Jing. 2000. Sentence reduction for automatic text summarization. In *Proceedings of Applied Natural Language Processing*, pages 310–315.

Kevin Knight and Daniel Marcu. 2002. Summarization beyond sentence extraction: a probabilistic approach to sentence compression. *Artificial Intelligence*, 139(1):91–107.

Emiel Krahmer, Erwin Marsi, and Paul van Pelt. 2008. Query-based sentence fusion is better defined and leads to more preferred results than generic sentence fusion. In *Proceedings of ACL*, pages 193–196.

Erwin Marsi and Emiel Krahmer. 2005. Explorations in sentence fusion. In *Proceedings of the European Workshop on Natural Language Generation*, pages 109–117.

Ryan McDonald. 2006. Discriminative sentence compression with soft syntactic evidence. In *Proceedings of EACL*, pages 297–304.

Kathleen R. McKeown, Regina Barzilay, David Evans, Vasileios Hatzivassiloglou, Judith L. Klavans, Ani Nenkova, Carl Sable, Barry Schiffman, and Sergey Sigelman. 2002. Tracking and summarizing news on a daily basis with Columbia's Newsblaster. In *Proceedings of HLT*, pages 280–285.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. 2008. Cheap and fast—but is it good?: Evaluating non-expert annotations for natural language tasks. In *Proceedings of EMNLP*, pages 254–263.

Luis von Ahn and Laura Dabbish. 2004. Labeling images with a computer game. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 319–326.

Zhuli Xie, Barbara Di Eugenio, and Peter C. Nelson. 2008. From extracting to abstracting: Generating quasi-abstractive summaries. In *Proceedings of LREC*, May.