Supervised Sentence Fusion with Single-Stage Inference

Kapil Thadani and **Kathleen McKeown**
Department of Computer Science
Columbia University
New York, NY 10025, USA
{kapil,kathy}@cs.columbia.edu

Abstract

Sentence fusion—the merging of sentences containing similar information—has been shown to be useful in an abstractive summarization context. We present a new dataset of sentence fusion instances obtained from evaluation datasets in summarization shared tasks and use this dataset to explore supervised approaches to sentence fusion. Our proposed inference approach recovers the highest scoring output fusion under an n-gram factorization using a compact integer linear programming formulation that avoids cycles and disconnected structures. In addition, we introduce simple fusion-specific features and constraints that outperform a compression-inspired baseline as well as a variant that relies on human-identified concept spans for perfect content selection.

1 Introduction

Abstractive text summarization has long been a high-level goal of natural language processing. Although progress in text-to-text (T2T) generation tasks such as sentence compression and paraphrase generation has been steady, the fusion of multiple sentences offers a particularly formidable challenge. Sentence fusion refers to the task of combining two or more sentences which overlap in information content, avoiding extraneous details and preserving common information. This procedure has been observed in human summarization (Jing and McKeown, 2000) and has been shown to be a valuable component of automated summarization systems (Barzilay and McKeown, 2005). However, research in sentence fusion has long been hampered by the absence of datasets for the task, and the difficulty of generating one has cast doubt on the viability of automated fusion (Daumé III and Marcu, 2004).

This paper presents a new fusion dataset generated from existing human annotations and also introduces a discriminative T2T system that generalizes the single sentence compression approach of Thadani and McKeown (2013) to n-way sentence fusion. Our fusion dataset is constructed from evaluation data for summarization shared tasks in the Document Understanding Conference (DUC)[1] and the Text Analysis Conference (TAC).[2] Specifically, we use human-generated annotations produced for the pyramid method (Nenkova et al., 2007) for summarization evaluation to produce a dataset of natural human fusions with quantifiable agreement. This offers advantages over previous datasets used for standalone English sentence fusion which contain annotator-induced noise (McKeown et al., 2010) or cannot be distributed (Elsner and Santhanam, 2011). In addition, both these datasets contain approximately 300 instances of fusion while the new dataset presented here contains 1858 instances.

Crucially, this larger corpus encourages supervised approaches to sentence fusion and we leverage this to explore new strategies for the task. Previous approaches to fusion have generally relied on variations of dependency graph combination (Barzilay and McKeown, 2005; Filippova and Strube, 2008b; Elsner and Santhanam, 2011) for content selection with a separate step for linearization that is usually based on a language model (LM). In contrast, we experiment with combin-

[1] http://duc.nist.gov
[2] http://www.nist.gov/tac

1	In 1991, the independents claimed nearly a third of adult book purchases **but six years later their market share** was nearly cut in half, **down to 17%**.
2	**By 1999, independent booksellers held only a 17 percent market share**.
SCU	Six years later independent booksellers' market share was down to 17%
1	**The heavy-metal group Metallica filed a federal lawsuit in 2000 against Napster for copyright infringement**, charging that Napster encouraged users to trade copyrighted material without the band's permission.
2	The heavy metal rock band **Metallica**, rap artist Dr. Dre and the RIAA **have sued Napster**, developer of Internet sharing software, **alleging the software enables the acquisition of copyrighted music without permission**.
3	**The heavy-metal band Metallica sued Napster** and three universities **for copyright infringement** and racketeering, seeking $10 million in damages.
SCU	Metallica sued Napster for copyright infringement
1	The government was to pardon 23 FARC members as the two sides **negotiate prisoner exchanges**.
2	The Columbian government plans to pardon more than 30 members of FARC as **they negotiate a prisoner swap**.
3	**The government and FARC continued to argue over details of a prisoner swap**.
SCU	The government and FARC negotiate prisoner exchanges

Table 1: SCU annotations drawn from DUC 2005–2007 and TAC 2008–2011. Human-annotated contributors to the SCU are indicated as boldfaced spans within the respective source sentences.

ing linearization with content selection to produce a single-stage joint approach to fusion. For this, we adapt the sequential *structured transduction* approach described in Thadani and McKeown (2013) for sentence compression and extend it to process multiple input sentences for fusion tasks. This discriminative approach to sentence generation permits rich features that estimate the informativeness of specific tokens chosen from the input sentences as well as the fluency of the n-grams used to assemble them for the output sentence. Furthermore, our inference formulation allows all potential orderings of input tokens to be considered in the output and prevents degenerate cylic or disjoint orderings via *commodity flow* constraints (Magnanti and Wolsey, 1994).

The primary contributions of this work are:

- A novel dataset of natural sentence fusions drawn from a corpus of pyramid evaluations for summarization shared tasks which is available to the NLP community.
- A supervised approach to sentence fusion that jointly addresses non-redundant content selection and linearization.

We evaluated the proposed fusion system against a basic compression baseline that does not include fusion-specific features as well as a proposed strong baseline that directly leverages human-annotated concept boundaries in the original dataset, thereby avoiding the issue of content selection. An evaluation under a variety of automated metrics indicates that our proposed approach strongly outperforms the former and appears competitive with the latter.

2 Pyramid fusion corpus

The pyramid method is a technique for summarization evaluation that aims to quantify the semantic content of summaries and compare automated summaries to human summaries on the basis of this semantic content (Nenkova et al., 2007). For each summarization topic to be evaluated, a number of human-authored summaries are first produced. In the DUC and TAC evaluations, the number of summaries is usually fixed at 7 per topic. A collection of *summarization content units* or SCUs—intended to correspond to atomic units of information—are then generated by annotators reading these summaries. Each SCU comprises a *label* which is a concise English sentence that states the meaning of the SCU[3] and a list of *contributors* which are discontinuous character spans from the summary sentences—herafter referred to as *source* sentences—in which that SCU is realized. Table 1 contains examples of SCUs drawn from DUC 2005–2007 and TAC 2008–2011 data.

Our fusion corpus is constructed by taking the source sentences of an SCU as input and the SCU labels as the gold fusion output. The fusion task posed by this corpus is similar to sentence *intersection* as defined by Marsi and Krahmer (2005) although it does not fit the criteria for *strict* intersection as addressed in Thadani and McKeown (2011) since source sentences do not always expressly mention all the information in an SCU label due to unresolved anaphora and entail-

[3] An SCU annotation guide from DUC 2005 is available at `http://www1.cs.columbia.edu/~ani/DUC2005/AnnotationGuide.htm`.

ment. The following procedure was used to extract meaningful fusion instances from the SCUs.

1. SCUs that have no more than one contributor which covers a single summary sentence are dropped. In addition, we chose to restrict the number of input sentences to at most four[4] since larger SCUs are very infrequent.
2. Although SCU descriptions are required to be full sentences, we found that this was not upheld in practice. We therefore removed SCUs whose labels contain fewer than 5 words and did not have an identifiable verb beyond the first token. As a practical consideration, SCUs with source sentences which have more than 100 tokens were also dropped.
3. Annotated concepts in this dataset often only cover a small fraction of source sentences and may not represent the full overlap between them. To account for this, we ignored SCUs without contributors that are at least half the length of their source sentences as well as SCUs whose labels are less than half the length of the smallest contributor.
4. Finally, we chose to retain only SCUs whose labels contain terms present in at least one source sentence, thus ensuring that gold fusions are reachable without paraphrasing.

This yields 1858 fusion instances of which 873 have two inputs, 569 have three and 416 have four.

3 Single-stage Fusion

Previous approaches to fusion have often relied on dependency graph combination (Barzilay and McKeown, 2005; Filippova and Strube, 2008b; Elsner and Santhanam, 2011) to produce an intermediate syntactic representation of the information in the sentence. Linearization of output fusions is usually performed by ranking hypotheses with a language model (LM), sometimes with language-specific heuristics to filter out ill-formed sentences. This approach is also known as *overgenerate-and-rank* and is often found to be a source of errors in T2T problems (Barzilay and McKeown, 2005).

Although syntactic representations are natural for assembling text across sentences, recent work in unsupervised multi-sentence fusion has shown that well-formed output can often be constructed purely on the basis of adjacency relationships in a word graph (Filippova, 2010). Similarly, systems for related T2T tasks such as sentence compression (McDonald, 2006; Clarke and Lapata, 2008) and strict sentence intersection (Thadani and McKeown, 2011) have also seen promising results by linearizing n-grams without explicitly relying on syntactic representations.

Our framework takes a similar perspective and assembles output text directly from n-grams over input tokens, but we employ a discriminative structured prediction approach in which likelihood under an LM is one of many features of output quality and parameters for all features are learned from a training corpus. Moreover, rather than rely on pipelined stages to first select the output content and then linearize an intermediate representation, we jointly address token selection alongside phrase-based ordering thereby yielding a single-stage approach to fusion.

3.1 ILP formulation

The starting point for this work is the sequential structured transduction[5] model of Thadani and McKeown (2013), originally devised for single sentence compression. This approach relies on integer linear programming (ILP) to find a globally optimal solution to generation problems involving heterogenous substructures. ILP has been used frequently in recent T2T generation systems including many for sentence fusion (Filippova and Strube, 2008b; Elsner and Santhanam, 2011), intersection (Thadani and McKeown, 2011) and compression (Clarke and Lapata, 2008; Filippova and Strube, 2008a; Berg-Kirkpatrick et al., 2011), as well as other natural language processing tasks. Although LPs with integer constraints are NP-hard in the general case, the availability of optimized general-purpose ILP solvers and the natural limits on English sentence length make ILP inference attractive for sentence-level optimization problems.

Consider a single fusion instance involving k source sentences $\mathcal{S} \triangleq \{S_1, \ldots, S_k\}$. The notation $F_\mathcal{S}$ is used to denote a fusion of the sentences in $\mathcal{S}$. The inference step aims to retrieve the output sentence $F^*_\mathcal{S}$ that is the most likely fusion of $\mathcal{S}$, i.e., the sentence that maximizes $p(F_\mathcal{S}|S)$ or equivalently maximizes some scoring function $score(F_\mathcal{S})$. In

[4] This is accomplished by removing additional contributors that share the fewest words with the SCU label.

[5] The full joint model presented in Thadani and McKeown (2013) also explictly infers tree-structured dependencies, but we found in preliminary experiments that this did not perform well with multiple sentence inputs. See discussion in §5.

our feature-based discriminative setting, we define $score(F_{\mathcal{S}})$ as a dot product of weights $\mathbf{w}$ and a feature map $\mathbf{\Phi}(S, F_{\mathcal{S}})$ defined over the fusion and its input; in other words

$$F^*_{\mathcal{S}} \triangleq \arg\max_{F_{\mathcal{S}}} \ \mathbf{w}^\top \mathbf{\Phi}(S, F_{\mathcal{S}}) \tag{1}$$

The feature map $\mathbf{\Phi}$ for an arbitrary fusion sentence is defined to factor over the words and potential n-grams from the input text. Let $T \triangleq \{t_i : 1 \leq i \leq N_j, 1 \leq j \leq |\mathcal{S}|\}$ represent the set of tokens (including duplicates) in $\mathcal{S}$ and let $x_i \in \{0, 1\}$ represent a token indicator variable whose value corresponds to whether token t_i is present in the output sentence $F_{\mathcal{S}}$. We also consider n-gram phrases defined over the tokens in T and assume the use of bigrams without loss of generality.[6] Let U represent the set of all possible bigrams that can be constructed from the tokens in T; in other words $U \triangleq \{\langle t_i, t_j\rangle : t_i \in T \cup \{\text{START}\}, t_j \in T \cup \{\text{END}\}, i \neq j\}$. Following the notation for token indicators, let $y_{ij} \in \{0, 1\}$ represent a bigram indicator variable for whether the contiguous pair of tokens $\langle t_i, t_j\rangle$ is in the output sentence. We represent entire token and bigram configurations with incidence vectors $\mathbf{x} \triangleq \langle x_i\rangle_{t_i \in T}$ and $\mathbf{y} \triangleq \langle y_{ij}\rangle_{\langle t_i, t_j\rangle \in U}$ which are equivalent to some subset of T and U respectively. With this notation, (1) can be rewritten as

$$\begin{aligned} F^*_{\mathcal{S}} = \ & \arg\max_{\mathbf{x},\mathbf{y}} \ \sum_{t_i \in T} x_i \cdot \boldsymbol{w}^\top_{\text{tok}} \boldsymbol{\phi}_{\text{tok}}(t_i) \\ & + \sum_{\langle t_i, t_j\rangle \in U} y_{ij} \cdot \boldsymbol{w}^\top_{\text{ngr}} \boldsymbol{\phi}_{\text{ngr}}(\langle t_i, t_j\rangle) \\ = \ & \arg\max_{\mathbf{x},\mathbf{y}} \ \mathbf{x}^\top \boldsymbol{\theta}_{\text{tok}} + \mathbf{y}^\top \boldsymbol{\theta}_{\text{ngr}} \end{aligned} \tag{2}$$

where $\boldsymbol{\phi}$ is a feature vector for tokens or bigrams and $\boldsymbol{w}$ is a corresponding vector of weight parameters. Each $\boldsymbol{\theta} \triangleq \langle \boldsymbol{w}^\top \boldsymbol{\phi}(s)\rangle$ is therefore a vector of feature-based scores for either tokens or bigrams.

The joint objective in (2) conveniently permits content-based features in $\boldsymbol{\phi}_{\text{tok}}$ for content selection and fluency features such as LM log-likelihoods in $\boldsymbol{\phi}_{\text{ngr}}$ for linearization. However, decoding a valid sentence with this objective is non-trivial. Merely selecting the tokens and bigrams that maximize (2) is liable to produce degenerate structures, i.e., cycles, disconnected components, branches and inconsistency between the token and bigram configurations in $\mathbf{x}$ and $\mathbf{y}$. Most prior T2T linearization approaches such as the Viterbi-based approach of McDonald (2006) and the ILP of Clarke and Lapata (2008) cannot be applied when the tokens in the input do not have a total ordering, as is the case when the input consists of more than one sentence.

[6]This approach permits n-grams of any order (Thadani and McKeown, 2013) but we use bigrams here to produce ILPs that scale quadratically with the number of input tokens.

3.2 Structural Constraints

We now briefly describe the structural constraints proposed by Thadani and McKeown (2013) to address the problem of degeneracy in sentential structure. First, we consider the problem of output *consistency*—more formally, bigram variables y_{ij} that are non-zero must activate their token variables x_i and x_j while token variables can only activate a single bigram variable in the first and second position each.

$$x_i - \sum_j y_{ij} = 0, \quad \forall t_j \in T \tag{3}$$

$$x_j - \sum_i y_{ij} = 0, \quad \forall t_i \in T \tag{4}$$

The second requirement for non-degenerate output is that non-zero y_{ij} must form a sentence-like *linear ordering* of tokens, avoiding cycles and branching. For this purpose, auxiliary variables are introduced to establish *single-commodity flow* (Magnanti and Wolsey, 1994) between all pairs of tokens that may appear adjacent in the output. Linear token ordering is maintained by defining real-valued commodity flow variables γ_{ij} which are non-negative.

$$\gamma_{ij} \geq 0, \quad \forall \langle t_i, t_j\rangle \in U \tag{5}$$

Each active token in the solution must have some positive incoming commodity and *consumes* one unit of this commodity, transmitting the remaining value to outgoing flow variables. This ensures that cycles cannot be present in the flow structure.

$$\sum_i \gamma_{ij} - \sum_k \gamma_{jk} = x_j, \quad \forall t_j \in T \tag{6}$$

The acyclic flow structure can be imparted to $\mathbf{y}$ by constraining bigram indicators to be active only if their corresponding tokens have positive commodity flow between them.

$$\gamma_{ij} - C_{\text{max}} y_{ij} \leq 0, \quad \forall \langle t_i, t_j\rangle \in U \tag{7}$$

where C_{max} is the maximum amount of commodity that the γ_{ij} variables may carry and serves as an upper bound on the number of output tokens.

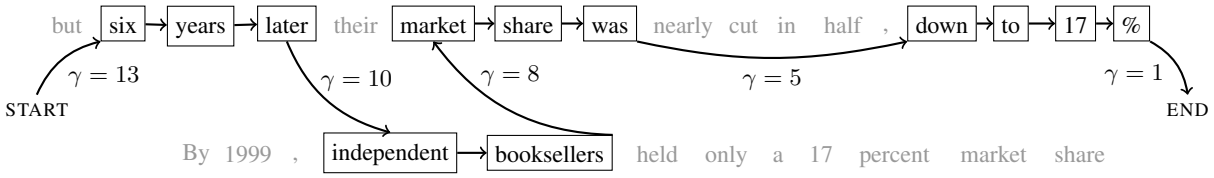

Figure 1: An illustration of commodity values for a valid solution of the ILP.

Finally, in order to establish *connectivity* in the output, we also introduce indicator variables y_{*j} and y_{i*} to denote the sentence-starting and terminating bigrams $\langle \text{START}, t_j \rangle$ and $\langle t_i, \text{END} \rangle$ respectively. A valid output sentence must be started and terminated by exactly one bigram.

$$\sum_j y_{*j} = 1 \tag{8}$$

$$\sum_i y_{i*} = 1 \tag{9}$$

Flow variables γ_{*j} and γ_{i*}, are also defined for START and END respectively. Since START has no incoming flow variables, the amount of commodity in γ_{*j} are unconstrained. This provides the only point of origin for the commodity and, in conjunction with (7), induces connectivity in $\mathbf{y}$.

3.3 Further Extensions for Fusion

The constraints specified above are adequate to enforce structural soundness in an output sentence and are applicable to a range of T2T linearization problems. We now address the issue of *redundancy*, which is unique to sentence fusion. The input sentences are expected to contain overlapping information which is useful to identify because: (a) it is a signal of salience, and (b) it is reasonable to expect that this repeated information should not appear redundantly in the output.

3.3.1 Supported content words

To address the first point above, we iterate through each sentence and generate groups G of similar or identical tokens across sentences, which we refer to as *supported* tokens. The selection of tokens is limited to open-class words such as nouns, verbs, adjectives and adverbs. Matching is accomplished via stemming, lemmatization, Wordnet synonymy and abbreviation expansion, and each group G_k is closed under transitivity. We expect that tokens from large groups, i.e., occurences in multiple sentences or repeated occurences in a single sentence, will be more likely to appear in the output. In the following section, we design features over supporting tokens so that the learning algorithm can encourage or discourage their occurrence following the patterns seen in the training corpus.

3.3.2 Redundancy constraints

While we expect largely positive weights on features for supporting tokens, this will also have the effect of encouraging of more than one token from the same group to occur in the output. In order to avoid this problem, we add a constraint for each group $G_k \in \mathcal{G}$ that prevents tokens within a group from appearing more than once.

$$\sum_{i:t_i \in G_k} x_i \leq 1, \quad \forall G_k \in \mathcal{G} \tag{10}$$

3.4 Features

We now describe the features ϕ over tokens and bigrams that guide inference for fusion instances.

- **Salience**: Fluent output fusions might require specific words to be preserved, highlighted or perhaps rejected. This can be expressed through features on token variables that indicate *a priori* salience, for which we consider patterns of part-of-speech (POS) tags and dependency arc labels obtained from input parses. Specifically, we define indicator features for POS sequences of length up to 2 that surround the token and the POS tag of the token's syntactic governor conjoined with the label. We also maintain features for whether tokens appear within parentheses and if they are part of a capitalized sequence of tokens (an approximation of named entity markup).
- **Fluency**: These features are intended to capture how the presence of a given bigram contributes to the overall fluency of a sentence. The bigram variables are scored with a feature expressing their log-likelihood under an LM. We also include features that indicate the sequence of POS tags and dependency labels corresponding to the tokens an bigram variable covers.

- **Fidelity**: One might reasonably expect that many bigrams in the input sentences will appear unchanged in the output fusion. We therefore propose boolean features that indicate whether an bigram was seen in the input.
- **Pseudo-normalization**: A major drawback of using linear models for generation problems is an inability to employ output sentence length normalization when scoring structures. Word penalty features are used for this purpose following their use in machine translation (MT) systems. These features are simply set to 1 for every token and bigram and their parameters are intended to balance out biases in output length that are induced by other features.
- **Support**: We note the amount of support—repetitions across input sentences—for nouns, verbs, adjectives and adverbs, as described in §3.3. We define features that count the number of repetitions for each of these tokens, and conjoin this with the POS class of each token. We also include binary variants of these features that indicate whether a token has support across 2, 3 or 4 input sentences. The constraint in (10) prevents these features from encouraging redundancy in the output.

Each scale-dependent feature is recorded absolutely as well as normalized by the average length of an input sentence. This was done in order to encourage the model to be robust to variation in sentence length during training.

3.5 Learning

The structured perceptron (Collins, 2002) was used in our experiments to recover good parameter settings $\mathbf{w}^*$ for the above features from training corpora. We used a fixed learning rate, averaged parameters over all iterations, and tracked performance in each epoch against a held-out development corpus. Following Martins et al. (2009), inference was sped up during training by only solving an LP relaxation of the fusion ILP.

4 Experiments

In order to evaluate our proposed fusion approach, we ran experiments over the corpus described in §2. For ease of reproducibility, we did not split the corpus randomly, rather, the 593 instances from the DUC evaluations covering the years 2005–2007 were chosen as a testing corpus, while the 1265 instances from the TAC evaluations over 2008–2011 were used as a training corpus. This yields an approximate 70/30 train-test split with near-identical proportions of 2-way, 3-way and 4-way fusions. In addition, we used 10% of the training section (all from 2011) as a development corpus in order to tune the features.

Dependency parses for features were generated using the Stanford parser[7] and LMs were constructed from the Gigaword corpus. All ILPs were solved using Gurobi.[8] All possible token orderings were permitted for fusion inference with the exception of those that flipped the order of two tokens from the same input sentence, which we assumed to be highly unlikely.

4.1 Baselines

The lack of a standard corpus and domain makes comparisons against previous systems difficult. Indeed, we propose that the pyramid fusion corpus described here may be well suited for comparing fusion systems in the future.[9]

We therefore use two baselines for this evaluation. First, we consider a compression baseline that is a variant of the system under study but without the fusion-specific modifications, i.e., the support features and the redundancy constraint from (10). This is not a strong baseline—we do not expect it to outperform our system for this task—but it serves as a useful measure of how linearization performs in the absence of content selection.

Our second baseline uses an identical system to the first but operates on different input data—the SCU *contributors* for each instance instead of the full source sentences. These are human-selected text spans that realize the SCU as defined in the pyramid evaluation guidelines and therefore approximate gold content selection. One-third of the instances in the corpus (659 instances) have SCUs that are exact string matches of one of the contributors;[10] the corresponding count for SCU-matching source sentences is less than half (300 instances).

[7] `http://nlp.stanford.edu/software/`

[8] `http://www.gurobi.com`

[9] We hope to eventually distribute the extracted corpus directly but interested researchers can currently retrieve the raw data from NIST and reconstruct it from our guidelines in §2.

[10] We chose to leave these contributors in the corpus in order to more accurately model the decisions of human annotators who were generating the fusions.

Configuration	Input	n-grams F_1 %				Content words			Syntactic rels F_1%	
		$n=1$	2	3	4	P%	R%	F_1%	Stanford	RASP
Compression	Sources	27.08	14.97	8.64	4.85	40.05	28.20	30.17	14.19	12.71
	Contribs	36.38	22.43	14.72†	10.04†	**55.27**†	36.79	39.95	**22.81**†	20.24†
+ Support	Sources	**40.46**†	**24.92**†	**16.33**†	**11.00**†	49.01	**45.09**†	**44.42**†	**22.81**†	**21.25**†

Table 2: Experimental results under various quality metrics (see text for descriptions). Boldfaced entries in each column indicate statistically significant differences ($p < 0.05$) over other entries under Wilcoxon's signed rank test while † indicates the same under the paired t-test.

4.2 Evaluation Metrics

Sentence fusion is notoriously hard to evaluate (Daumé III and Marcu, 2004) and previous work tends to rely on human evaluations with Likert scales. However, we choose to follow work in machine translation and, more recently, sentence compression (Napoles et al., 2011; Thadani and McKeown, 2013) in moving towards transparent automated metrics for fusion quality in order to engender more repeatable evaluations of fusion systems. As our test corpus is larger than most previously-studied fusion corpora in their entirety, statistical measures of text quality are preferable.

Our basic evaluation metric is n-gram F_1, used in numerous tasks and evaluation scenarios; we consider all $1 \leq n \leq 4$. In addition, since n-gram metrics do not distinguish between content words and function words, we also include an evaluation metric that observes the precision, recall and F-measure of only nouns and verbs as a proxy for the informativeness of a given fusion.

In addition to the direct measures discussed above, we consider syntactic metrics that act as surrogates for grammaticality. Napoles et al. (2011) indicates that F_1 metrics over syntactic relations such as those produced by the RASP parser (Briscoe et al., 2006) correlate significantly with human judgments of compression quality; we expect that the same holds for our fusion scenario. Output fusions were therefore parsed with RASP as well as the Stanford dependency parser and their resulting dependency graphs were compared to those of the gold fusions.

4.3 Results

Table 2 summarizes the results from the fusion experiments. We first observe that the proposed fusion system as well as the contributor+compression baseline outperform the source+compression baseline significantly on all metrics evaluated. We also observe a significant gain for the fusion system over all baselines for F_1 over unigrams and bigrams, vindicating the proposed content-selection extensions to the baseline compression approach. Results for trigrams and 4-grams are statistically indistinguishable under the paired t-test, indicating that the proposed system is at least competitive with the 'cheating' baseline.

Turning to the content-word metrics, we see that the primary contribution of the discriminative joint approach is in enhancing the recall of meaning-bearing words. The gain in recall is larger than the loss in precision against the contributor+compression baseline, leading to a significant improvement in content word F_1.

Finally, the results from the syntactic measures of fluency are less clear. The proposed fusion system outperforms the strong baseline on RASP F_1 but the gain is only statistically significant under Wilcoxon's signed rank test. Both systems significantly outperform the weaker baseline.

Table 3 contains an example of system output illustrating the quirks of the different systems. We note that the results are often noisy in all scenarios and that the supported tokens do not entirely override the LM. For example, 'ABC' appears in only one of the input sentences in the second example from Table 3 but is seen in multiple system fusions, likely due to the influence of an LM trained on newswire text.

5 Discussion & Future Work

While the focus of this paper is on linearization, we also considered expanding the objective from (2) to include syntactic structures as presented in Thadani and McKeown (2013); however, initial results were not promising. We hypothesize that this is partly to the vulnerability of such representations to parse errors—also noted in Filippova (2010)—and partly to the severe independence assumptions involved in arc-factored dependency representations which are exacerbated when

Input 1	**Elian returned to Cuba on June 28 , 2000** .
Input 2	After a final appeal by the Miami relatives was denied and the court order blocking his return expired , **Elian returned with his father to Cuba on June 28 , 2000** .
Input 3	**On June 28** , the Supreme Court rejected a final appeal ; **Elian returned home to Cuba** , was celebrated in the media and returned to his home and schooling .
Gold	Elian returned with his father to Cuba on June 28 , 2000
Comp	Elian returned to Cuba on June returned with his father rejected a final appeal
Contribs	Elian returned to home to Cuba
+Support	Elian returned to Cuba on June 28
Input 1	**Jennings , who quit smoking several years ago , will undergo chemotherapy in New York .**
Input 2	ABC announced that Jennings would continue to anchor the news **during chemotherapy treatment** , but he was unable to do so .
Input 3	Peter Jennings hoarsely announced he had lung cancer on April 5 , 2005 and **would begin outpatient chemotherapy in New York** .
Gold	Jennings will undergo chemotherapy in New York
Comp	ABC announced that 2005
Contribs	would begin outpatient chemotherapy chemotherapy treatment
+Support	ABC announced that Jennings would undergo chemotherapy in New York

Table 3: Examples of system outputs for instances from the corpus. Contributors are indicated by boldfaced text spans.

working with multiple input sentences. We are currently working on extending this approach to produce richer formulations of syntax that will be more appropriate for this task.

6 Related Work

Sentence fusion is the general label applied to tasks which take multiple sentences as input to produce a single output sentence. Barzilay & McKeown (Barzilay et al., 1999; Barzilay and McKeown, 2005) first introduced fusion in the context of multidocument summarization as a way to better capture the information in a cluster of related sentences than just using the centroid. The fusion task has since expanded to include other forms of sentence combination, such as the merging of overlapping sentences in a multidocument context (Marsi and Krahmer, 2005; Krahmer et al., 2008; Filippova and Strube, 2008b) and the combination of two (usually contiguous) sentences from a single document (Daumé III and Marcu, 2004; Elsner and Santhanam, 2011). Variations on the fusion task include the set-theoretic notions of *intersection* and *union* (Marsi and Krahmer, 2005; McKeown et al., 2010), which forego the problem of identifying relevance and are thus less dependent on context. Query-based versions of these tasks have been studied by Krahmer et al. (2008) and have produced better human agreement in annotation experiments than generic sentence fusion (Daumé III and Marcu, 2004). McKeown et al. (2010) produced an annotated fusion corpus which was employed in experiments on decoding for sentence intersection (Thadani and McKeown, 2011). While most work in the area has covered pairwise sentence combination, recent work by Filippova (2010) has also addressed fusion—referred to as multi-sentence compression—within a cluster of sentences.

7 Conclusion

We have presented a new corpus for sentence fusion which is built from readily-available data used for summarization evaluation. To our knowledge, this is the largest corpus of fusion data studied to date. In addition, we proposed a supervised discriminative approach for sentence fusion that jointly selects content from the input and recovers a linearization without an intermediate representation. Our system uses a flexible integer linear programming formulation for generating acyclic paths in token graphs, generalizing a state-of-the-art sentence compression approach to multiple sentences and a supervised setting which permits rich, linguistically-motivated features that factor over tokens and n-grams. We demonstrate that this approach leads to significant performance gains over a baseline compression system as well as comparable performance to an approach which directly leverages human content selection.

Acknowledgments

This work was supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center (DoI/NBC) contract number D11PC20153. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC, or the U.S. Government.

References

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

Regina Barzilay, Kathleen R. McKeown, and Michael Elhadad. 1999. Information fusion in the context of multi-document summarization. In *Proceedings of ACL*, pages 550–557.

Taylor Berg-Kirkpatrick, Dan Gillick, and Dan Klein. 2011. Jointly learning to extract and compress. In *Proceedings of ACL-HLT*, pages 481–490.

Ted Briscoe, John Carroll, and Rebecca Watson. 2006. The second release of the RASP system. In *Proceedings of the ACL-COLING Interactive Presentation Sessions*.

James Clarke and Mirella Lapata. 2008. Global inference for sentence compression: an integer linear programming approach. *Journal for Artificial Intelligence Research*, 31:399–429, March.

Michael Collins. 2002. Discriminative training methods for hidden Markov models. In *Proceedings of EMNLP*, pages 1–8.

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.

Micha Elsner and Deepak Santhanam. 2011. Learning to fuse disparate sentences. In *Proceedings of the Workshop on Monolingual Text-To-Text Generation*, pages 54–63.

Katja Filippova and Michael Strube. 2008a. Dependency tree based sentence compression. In *Proceedings of INLG*, pages 25–32.

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

Katja Filippova. 2010. Multi-sentence compression: finding shortest paths in word graphs. In *Proceedings of COLING*, pages 322–330.

Hongyan Jing and Kathleen R. McKeown. 2000. Cut and paste based text summarization. In *Proceedings of NAACL*, pages 178–185.

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-HLT*, pages 193–196.

Thomas L. Magnanti and Laurence A. Wolsey. 1994. Optimal trees. In *Technical Report 290-94, Massechusetts Institute of Technology, Operations Research Center*.

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

André F. T. Martins, Noah A. Smith, and Eric P. Xing. 2009. Concise integer linear programming formulations for dependency parsing. In *Proceedings of ACL-IJCNLP*, pages 342–350.

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

Kathleen McKeown, Sara Rosenthal, Kapil Thadani, and Coleman Moore. 2010. Time-efficient creation of an accurate sentence fusion corpus. In *Proceedings of HLT-NAACL*, pages 317–320.

Courtney Napoles, Benjamin Van Durme, and Chris Callison-Burch. 2011. Evaluating sentence compression: pitfalls and suggested remedies. In *Proceedings of the Workshop on Monolingual Text-To-Text Generation*, pages 91–97.

Ani Nenkova, Rebecca Passonneau, and Kathleen McKeown. 2007. The pyramid method: Incorporating human content selection variation in summarization evaluation. *ACM Transactions on Speech and Language Processing*, 4(2), May.

Kapil Thadani and Kathleen McKeown. 2011. Towards strict sentence intersection: decoding and evaluation strategies. In *Proceedings of the Workshop on Monolingual Text-To-Text Generation*, pages 43–53.

Kapil Thadani and Kathleen McKeown. 2013. Sentence compression with joint structural inference. In *Proceedings of CoNLL*.