The ICSI/UTD Summarization System at TAC 2009

Dan Gillick[1]**, Benoit Favre**[1]**, Dilek Hakkani-Tür**[1]**, Berndt Bohnet**[1]**, Yang Liu**[2]**, Shasha Xie**[2]
[1]International Computer Science Institute, Berkeley, USA
[2]University of Texas, Dallas, USA
{dgillick,favre,dilek}@icsi.berkeley.edu
{shasha,yangl}@hlt.utdallas.edu

Abstract

We describe improvements to our 2008 system that result in a top-performing summarization system. The motivating ideas are (1) improve sentence boundary detection to avoid damaging errors in preprocessing; (2) prune sentences that are unlikely to work well in a summary; (3) leverage sentence position to improve update summarization; (4) focus on high-precision sentence compression to improve readability rather than content.

1 Introduction

The system we built for the TAC 2009 Summarization task is a refinement and an expansion of the system described in our 2008 system paper (Gillick et al., 2008). This was a bare-bones system, with minimal preprocessing, intended to demonstrate the power of a new sentence selection method. This year, we made a number of improvements, primarily based on the results of our own evaluation of potential causes for low linguistic quality scores.

Our approach to sentence selection is based on the *maximum coverage* model for summarization introduced by Filatova and Hatzivassiloglou in 2004 (Filatova and Hatzivassiloglou, 2004), inspired by the well-known set-cover problem (Hochbaum, 1996). In our model, a summary is the set of sentences that best covers the relevant concepts in the document set, where concepts are simply word bigrams valued by their document frequency. The value of a summary is the sum of the unique concept values it contains, thus limiting redundancy implicitly. The maximization can be solved approximately with local search methods or exactly using an Integer Linear Program (ILP). Though the formulation is certainly NP-hard (it is a fancier version of the knapsack problem), a standard ILP solver finds solutions in less than 1 second for all problems in TAC 2008 and 2009 (more details in (Gillick and Favre, 2009)).

The system we submitted in 2008 produced very high ROUGE and Pyramid scores, especially for the non-update set. However, overall responsiveness was dragged down by low linguistic quality. After sorting out specific errors in the low-scoring summaries, we prioritized a list of issues to address:

1. Sentence boundary errors: Though rare, each one of these ruined an entire summary.
2. Unclear references: Selecting from the full set of sentences leaves us vulnerable to including sentences that don't make sense in isolation.
3. Relative dates and "said" clauses: A huge number of input sentences include phrases like "on Tuesday" or "the President said", which often don't belong in a summary.
4. Bad compression: We experimented with a syntactic compression algorithm that allowed for joint selection and compression, but it introduced many ungrammatical sentences.
5. Update summarization: No specific processing for the update portion of the task resulted in significantly lower scores for this set of problems.

The following sections describe our approaches to address these issues. We also show results relative to the other submissions, and outline future work based on a new linguistic quality evaluation.

2 Sentence Boundaries

The sentence segmentation problem—the disambiguation of periods that arises because periods signal abbreviations as well as sentence boundaries—is mostly disregarded because a few rules catch most of the common situations. But even the strongest rule-based system has an error rate (in English) of 1% (Aberdeen et al., 1995). Kiss and Strunk's *Punkt* (Kiss and Strunk, 2006) system is widely used (1.65% error rate on Wall Street Journal text; 3% error rate if used with the included model). But when a single segmentation error can ruin the entire summary, this problem becomes more important. Indeed, using the Punkt system in 2008, over 20% of our summaries contained at least one segmentation error.

Around the same time, Palmer and Hearst (Palmer and Hearst, 1997) and Reynar and Ratnaparkhi (Reynar and Ratnaparkhi, 1997) built Sentence Boundary Detection (SBD) systems by training a classifier with local context features. To achieve error rates competitive with the rule-based systems, they included special abbreviation features, essentially a list of common abbreviations. Error rates on the same Wall Street Journal corpus were between 1% and 2%.

Based on the observation that the really hard cases are abbreviations that also end sentences (Table 1 shows the most common sentence-ending abbreviations), we built a new SBD system that achieves an error rate of 0.25% on the Wall Street Journal corpus, and under 0.5% on the Brown corpus as well as the complete works of Edgar Allen Poe (Gillick, 2009). This involves training a classifier (SVM worked best) using a new set of features based solely on the word before and the word after the period in question. There is no dedicated abbreviation feature.

This system[1] is used to segment sentences for TAC 2009. While it is possible there were some errors, the resulting summaries appear to be nearly error-free.

Abbr.	Ends Sentence	Total	Ratio
Inc.	109	683	0.16
Co.	80	566	0.14
Corp.	67	699	0.10
U.S.	45	800	0.06
Calif.	24	86	0.28
Ltd.	23	112	0.21

Table 1: The abbreviations appearing most often as sentence boundaries. These top 6 account for 80% of sentence-ending abbreviations in the test set, though only 5% of all abbreviations.

3 Sentence Pruning

In last year's evaluation, we observed that about 50% of low linguistic quality summaries (scoring one or two out of five) contained at least one unclear reference, often a pronoun referring to a name not introduced in the summary. One way to address this issue is with coreference resolution, linking references to names, and then substituting the full name where appropriate. While theoretically appealing, this is impractical. Coreference resolution is rarely accurate above 70%, and deciding what sort of substitution to use is difficult (we looked at the output of a current state-of-the-art system before opting for a different approach).

Instead, we simply prune input sentences with unresolved references, since they are too much of a liability when it comes to assembling an extractive summary. To do this, we built a system for detecting pronouns that are unresolved in the sentence in which they occur. The following sentence, for example, contains two pronouns, "it" and "its", which refer to the company "Coda": *"Coda, an oil and gas concern, said **it** and **its** partners received $7 million in cash and $10 million in five-year notes for the Kansas intrastate pipeline"*. This version of the sentence contains unresolved pronouns: *"They received cash and bank-notes to back the investment."*

Our unresolved pronoun detector classifies each pronoun in the input as resolved or unresolved, using features extracted from the parse tree of the sentence. We use the OntoNotes 2.9 (Hovy et al., 2006) coreference resolution data as training data. A pronoun is considered to be resolved if a non-pronominal reference to the same entity is present in

[1] *Splitta*, with source code, is publicly available at http://code.google.com/p/splitta

the sentence. The processing pipeline for detecting unresolved pronouns is as follows:

1. Parse an input sentence using the Berkeley constituency parser (Petrov and Klein, 2007).
2. Locate a potential pronoun with the "PRP" and "PRP$" part-of-speech tags.
3. Extract features from the parse tree such as the quotes, words, and part-of-speech tags in the vicinity of the pronoun, repetitions, availability of noun phrases and pronouns elsewhere in the sentence, and general constituency features from the path between the current pronoun and other noun phrases.
4. Drop the sentence if one unresolved pronoun is detected.

An Adaboost classifier[2] is trained on about 6,400 pronoun instances of which fifty percent are positive examples. It performs at an F-score of 0.89 at the pronoun level, on a 1,000 example held-out set from the OntoNotes data. We observed that changing the decision threshold of the classifier to maximize ROUGE instead of F-score could result in small improvements but did not apply this trick in the submission to avoid optimizing for ROUGE.

Using a decision threshold of 0.4, which maximizes F-score, 31% of sentences were pruned from the TAC 2008 data set and 29% from TAC 2009.

Input	**ROUGE-2 (A)**	**ROUGE-2 (B)**
Unfiltered	0.1990	0.1987
Filtered	0.1942	0.1960

Table 2: Oracle results for set A and set B (update set) suggest that very little potential ROUGE score is sacrificed by pruning sentences with unresolved pronominal references.

We performed an oracle experiment where we use word bigrams valued by their frequency in the reference summaries (instead of the input documents) to create "max-ROUGE" summaries. Results, shown in Table 2, demonstrate that few important sentences are dropped in the process, as the filtered and unfiltered input yield similar results. We observed that the quantity of unresolved pronouns in the summaries decreased, but the approach does not suppress the problem of noun phrases that refer to an entity defined earlier in the text, such as "the president", which might only be detected properly with full coreference resolution.

We also experimented with content-based sentence pruning. We trained an SVM regression model to recover sentence-level ROUGE score based on a variety of standard frequency, position, and query features. While the resulting sentence scores looked quite reasonable, pruning low-scoring sentences did not improve our overall ROUGE results.

4 Compression

In our 2008 submission, we introduced a sentence compression component that would allow the summarizer to pick different versions of a sentence with some constituents removed. The compression was rather aggressive, achieving the highest reported ROUGE-2 scores at the expense of linguistic quality. 65% of the summaries with a low linguistic quality score contained ungrammatical sentences and 45% contained nonsensical but grammatical sentences.

This year, we took a different approach to compression. Rather than attempt to marginally increase potential content, we focused on *improving* linguistic quality by systematically removing temporal expressions, manner modifiers, and "said" clauses. This is a small subset of the space of removable clauses, but we chose them because their presence is almost always undesirable in a summary.

Temporal and manner modifiers are determined by semantic role labeling (SRL) according to the CoNLL shared task guidelines (Hajič et al., 2009), and "said" clauses are removed by re-rooting the dependency tree (using the quoted words as a new sentence). More specifically, the following processing is applied:

1. Generate a dependency parse tree with the MATE system (Bohnet, 2009).
2. Generate a semantic role labeling analysis on top of the dependency tree with the MATE system.
3. Mark temporal and manner semantic arguments labeled with ARGM-TMP or ARGM-

[2] http://code.google.com/p/icsiboost

MNR for removal when the confidence score of MATE is higher than a threshold.

4. Mark TMP arcs in the dependency tree for removal if the sub-tree contains a day-of-the-week and it is shorter than four words (to remove relative dates irrespective of their score).
5. Mark the OBJ child of "said" verbs ("said, says, tells, told, wrote, writes, write, reported") as potential new root of the dependency tree.
6. Generate all compression alternatives and keep the ten longest ones, including the original, to prevent short hypotheses with less information from entering the summary.
7. Skip hypotheses of a length less than half of the original and less than five words.
8. Detokenize and remove parenthesized content, drop sentences shorter than ten characters.

Original	**Compressed**
A ban against bistros providing plastic bags free of charge will be lifted at the beginning of March.	A ban against bistros providing plastic bags free of charge will be lifted.
December 19, 2000: Airbus officially launches the plane, calling it the A380.	December 19, 2000: Airbus launches the plane, calling it the A380.

Table 3: Example sentences compressed by removing temporal clauses or adverbial phrases.

We use the MATE system for semantic role labeling, and the output argument annotations as well as their confidence scores to filter out temporal arguments. To test the reliability of the estimated confidence scores, we automatically annotated the CoNLL-09 corpus English data with semantic role labels, and checked the ratio of correctly annotated examples in each confidence score interval (see Figure 1).

Testing on TAC 2008 data, we found that removing temporal expressions with confidence greater than 0.6 improved ROUGE-2 score by 3.5%. Adding manner semantic arguments and "said" clauses to the pruned set gave a ROUGE-2 improvement of 4.8%, almost enough for statistical significance at 95%. The 2009 results did not show a similar gain in ROUGE, but most importantly, compression actually improved linguistic quality slightly.

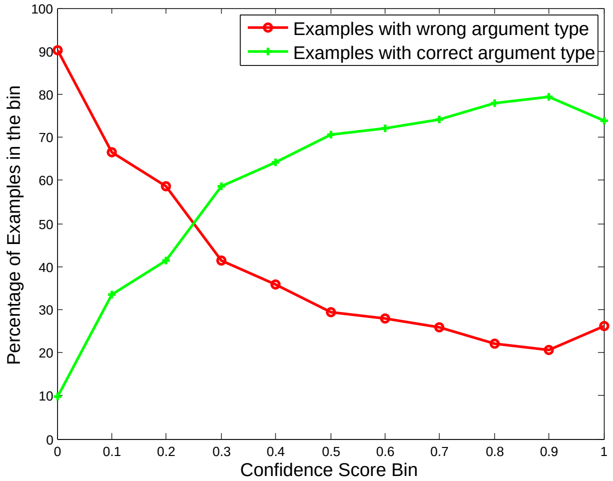

Figure 1: Percentages of correct versus incorrect SRL classifications at various confidence levels.

In order to account for compression candidates, our ILP decoder was modified with additional constraints that state that given a group of sentences derived from the same original (including the original), only one can be selected in the summary (Equation 6 in (Gillick et al., 2008)). We also observed that often the system would not compress a sentence if there was space remaining in the summary. We could significantly decrease the number of relative dates in the summaries by penalizing longer summaries that lead to the same concept selection. In order to do so, the length of the selection is subtracted from the objective function, scaled down by the maximum length so that it does not compete with the concepts.

Our two submitted systems had only one difference: the first did not use compression and the second did. 68% of the compression summaries were different in some way from the run without compression. In total, 34% of the sentences were compressed, leading to small differences in the output: more sentences selected (3.9 on average instead of 3.8 for the other run); shorter sentences (24.9 words per sentence instead of 25.5). The difference is mainly seen with temporal expressions, as there are 27% fewer days-of-week in the compressed summaries. This reduction in relative dates in likely responsible for improved linguistic quality.

5 Update Summarization

Last year, we used the same system for both update and non-update tasks. But clearly, some update task-specific processing can be beneficial. We made only one very small change to adapt our standard system to the update task, which managed to improve ROUGE-2 significantly. The key to this improvement was a careful study of sentence position.

While sentence position is often used as a feature in sentence classification approaches to summarization, there is little analysis of its value. The intuition is that sentences at the top of a document are more important. This is undoubtedly true, especially for short news articles, but what exactly is the relationship between position and importance?

Figure 2 shows the results of an experiment using ROUGE to measure the density of valuable words at each sentence position. Most striking is the disparity between first and second sentences: the first sentence of news document is really quite special. After the first sentence, ROUGE drops off nearly monotonically and much more gradually.

Also notable is the difference between update problems and non-update problems. While the first sentence stands apart in both cases, the total ROUGE value of subsequent sentences is considerably lower in the update problems. We hypothesize that articles about topics that have already been in the news tend to state new information first before recapping past details. Since the update task is concerned with what's new, the first sentence is especially valuable.

To test how we might leverage the disparity between first sentences and all the rest, we started with an extreme approach: only allow first sentences into summaries. As noted by (Schilder et al., 2008), first sentences make for good summary sentences. Leaving the system unchanged except that all non-first sentences are disallowed during inference gives surprisingly strong results, at least in terms of ROUGE. Had we submitted this system, it would have been ranked 3rd in ROUGE-2 (0.107), behind only our two actual submissions for set A, and 5th in ROUGE-2 (0.095) for set B. We might expect it to score well in linguistic quality as well given that first sentences tend to stand alone well.

We were able to improve on these ROUGE scores by including more sentences in the inference. Simply up-weighting the value of concepts appearing in first sentences had a dramatic effect on the ROUGE scores for the update set. Specifically, first-sentence concepts are up-weighted by a factor of 2 for set A and a factor of 3 for set B. Table 4 shows how this improves results.

Data	Before	After	Change
TAC 2008 (A)	0.1075	0.1169	+8.7%
TAC 2009 (A)	0.1048	0.1220	+16.4%
TAC 2008 (B)	0.0868	0.1137	+31.0%
TAC 2009 (B)	0.0906	0.1059	+16.8%

Table 4: Up-weighting first sentence concepts improves ROUGE-2 scores, especially for the update task.

Also interesting is the difference between sentence position values for each document source. In Associated Press documents, which tend to be short and snappy, the ratio in average ROUGE score between first and second sentences is 2.0. New York Times documents have a ratio of of 1.6 and Xinhua documents are closer to 1.4. We did not attempt to leverage these differences, but note that longer documents often do not start with a traditional newswire-type first sentence. Locating the more traditional opening sentence (probably one of the first 5 sentences) may prove fruitful.

6 TAC 2009 Results

Official results are shown in Table 6 and summarized visually in Figure 3. Both of our systems (Sys-34 and Sys-40) performed quite well, though the gap in Pyramid scores is most striking. Perhaps of little consequence, but our systems outperformed a number of the human annotators in terms of ROUGE. Example summaries are shown in Table 7.

6.1 Pre- and Post-Processing

We use a few regular expressions to clean the source documents, mostly to remove headers and bylines. After sentence segmentation, we re-attach multi-sentence quotations with a simple stack-based algorithm, and prune sentences that begin and end with a quotation mark. Sentences shorter than 10 words are not allowed in the final summaries.

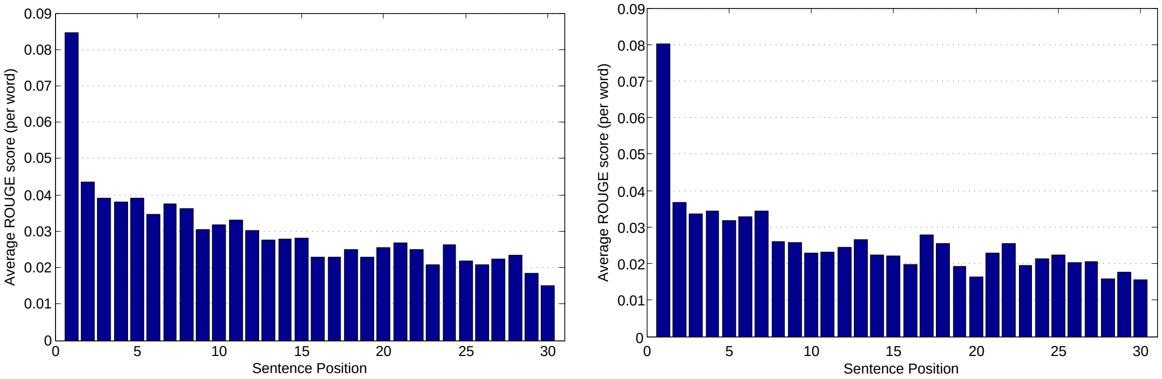

Figure 2: Average ROUGE-2 score at each sentence position for non-update topics (left) and update topics (right). Scores are normalized by the number of words in the sentence.

To order the final set of selected summaries, we make sure that the first sentence of the summary is the first sentence of one of the source documents (we found that forcing at least one such first sentence into the selected set had minimal effect on ROUGE). The rest of the sentences are ordered by source date, and then by position in their source documents.

6.2 Score Consistency

We noticed that across all pairs of summaries for each topic, a total of 226 are identical. This allows us to measure score consistency. We cannot measure inter-annotator agreement since a single judge scores every summary for a particular topic. Table 5 shows that the scores are fairly consistent, with Responsiveness scores a bit more stable than Linguistic Quality. The probability that the same annotator gives the same summary the same score is estimated as 0.53 for Responsiveness and 0.52 for Linguistic Quality. Responsiveness scores for identical summaries never differed by more than 2, and only 7 times did Linguistic Quality scores differ by 3 (scores are on a 10 point scale). A more complete analysis of score consistency, especially including inter-annotator agreement, would be valuable.

7 Future Work

To direct our future work, we conducted our own evaluation to help understand low linguistic quality scores. We read through all our systems' summaries receiving scores less than 5 (on the 10 point scale) and marked occurrences of various problems. Results are shown in Figure 4.

	Score Difference				
	0	1	2	3	mean
Resp	119	92	15	0	0.54
LQ	117	82	20	7	0.63

Table 5: 226 identical summaries often were given different scores by the same human judge. Counts of absolute score differences are shown for Responsiveness (Resp) and Linguistic Quality (LQ).

Targeted sentence compression helped reduce instances of relative dates, though further improvement is clearly possible. Since the ILP is choosing between compression alternatives (including the original sentence), there is no guarantee that the version of the sentence without dates will be selected. To address this issue, we may need to down-weight concepts that include dates.

The unreferenced-pronoun classifier certainly helped reduce instances of sentences with unclear references, though again, there is room for improvement. However, many of the remaining referential issues are more complex, and it is not clear whether some straightforward method would be fruitful.

Redundancy and structural issues certainly remain serious problems. But for the first time (in our system development), these complex issues may merit our attention. That is, it is not clear that there is a simple fix to address remaining redundancy and sentence ordering issues.

Code used for our submission is available at http://code.google.com/p/icsisumm.

System	Resp	LQ	Pyramid	ROUGE-2	BE	Rank Sum
sys-40	5.159 (1)	5.636 (5)	0.383 (1)	0.121 (2)	0.063 (2)	5
sys-34	4.841 (6)	5.273 (13)	0.374 (2)	0.122 (1)	0.064 (1)	18
sys-24	4.955 (2)	5.682 (4)	0.316 (10)	0.098 (13)	0.056 (7)	19
sys-10	4.909 (4)	5.636 (6)	0.312 (12)	0.102 (9)	0.057 (6)	24
sys-11	4.795 (7)	5.773 (3)	0.314 (11)	0.096 (15)	0.055 (10)	27
sys-24	5.023 (1)	5.886 (2)	0.296 (4)	0.096 (4)	0.064 (1)	4
sys-34	4.750 (2)	5.523 (6)	0.304 (2)	0.104 (1)	0.061 (3)	9
sys-40	4.568 (6)	5.500 (7)	0.290 (6)	0.104 (2)	0.062 (2)	17
sys-35	4.614 (4)	5.023 (19)	0.307 (1)	0.101 (3)	0.058 (4)	24
sys-51	4.568 (5)	5.114 (18)	0.299 (3)	0.096 (5)	0.055 (5)	27

Table 6: Overall system rankings for set A (above) and set B (below). Responsiveness, Linguistic Quality, Pyramid, ROUGE-2, and Basic Elements scores are shown with rank in parentheses. The rank sum gives some sense for the separation between systems, assuming each of these five scores are equally valuable. The top five ranking systems are shown for each set; our systems are in bold.

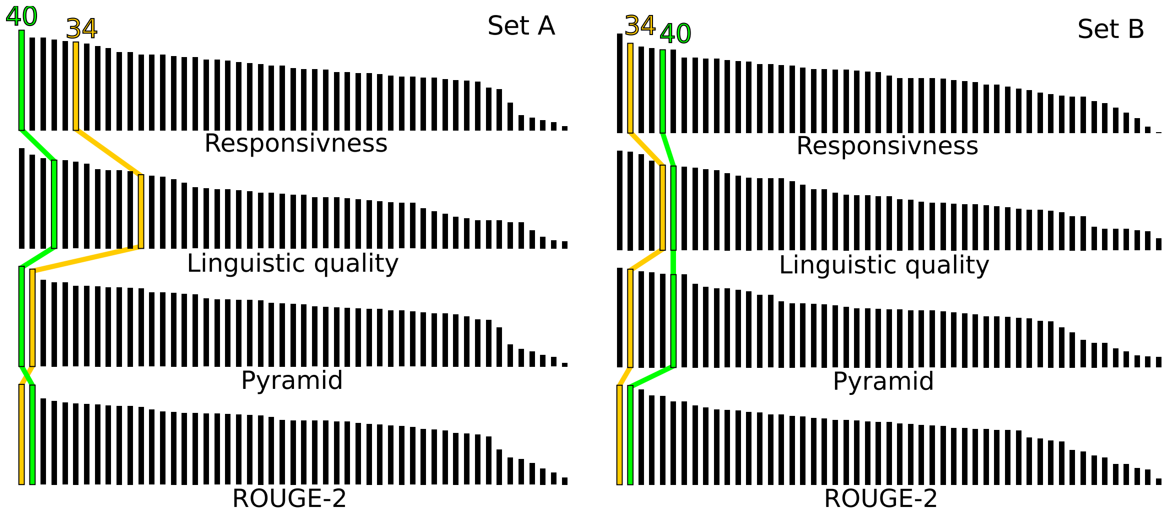

Figure 3: The big picture for Set A (left) and Set B (right) comparing our primary submission (34) and our run with compression (40) to the other systems on the main evaluation metrics.

References

J. Aberdeen, J. Burger, D. Day, L. Hirschman, P. Robinson, and M. Vilain. 1995. MITRE: description of the Alembic system used for MUC-6. In *Proceedings of the 6th conference on Message understanding*, pages 141–155. Association for Computational Linguistics Morristown, NJ, USA.

B. Bohnet. 2009. Efficient parsing of syntactic and semantic dependency structures. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL 2009): Shared Task*, pages 67–72.

E. Filatova and V. Hatzivassiloglou. 2004. Event-based extractive summarization. In *Proceedings of ACL Workshop on Summarization*, volume 111.

D. Gillick and B. Favre. 2009. A Scalable Global Model for Summarization. In *Proceedings of NAACL Workshop on Integer Linear Programming for Natural Language Processing*.

D. Gillick, B. Favre, and D. Hakkani-Tür. 2008. The ICSI Summarization System at TAC 2008. In *Proceedings of the Text Analysis Conference*.

D. Gillick. 2009. Sentence Boundary Detection and the Problem with the U.S. In *Proceedings of NAACL: Short Papers*.

J. Hajič, M. Ciaramita, R. Johansson, D. Kawahara, M.A. Martí, L. Márquez, A. Meyers, J. Nivre, S. Padó, J. Štěpánek, et al. 2009. The CoNLL-2009 shared task: Syntactic and semantic dependencies in multiple languages. In *Proceedings of the Thirteenth Confer-*

SYS-34	SYS-40	REFERENCE
The angry mother of a fallen U.S. soldier has met with advisers of President George W. Bush after staging a protest near his ranch to demand an accounting of how the president has conducted the war in Iraq. Earlier this year Sheehan formed Gold Star Families for Peace and has spoken to groups across the nation and overseas. Sheehan did meet with Bush in June 2004: She was among grieving military families who met with the president at Fort Lewis, Washington. Bush National Security Adviser Stephen Hadley and a deputy White House chief of staff talked to Sheehan.	The mother of an American soldier killed in Iraq led a protest march on President Bush's ranch here, prompting the White House to send two senior officials to meet her. Sheehan did meet with Bush in June 2004: She was among grieving military families who met with the president at Fort Lewis, Washington. Sheehan, a former Roman Catholic youth minister, formed a group called Gold Star Families For Peace and has spoken out against the war across the nation and overseas. Bush National Security Adviser Stephen Hadley and a deputy White House chief of staff talked to Sheehan.	Following her son's April 2004 death in Iraq, Cindy Sheehan has made regular public appearances across the nation protesting the war. She co-founded Gold Star Families for Peace. On August 8, 2005, Sheehan led an antiwar protest down the road leading to President Bush's ranch near Crawford, Texas. She and about 50 antiwar activists established a roadside camp not far from the ranch, vowing to remain until Bush met with her. She used the Internet to garner publicity. By August 11, several dozen protesters from across the country had joined Sheehan at Camp Casey, named for her fallen son.

Table 7: Example summaries for topic D0912-A: "Describe the anti-war protest efforts of Cindy Sheehan." One of the four model summaries is shown for reference. Both automatic summaries received scores of 5 and 4 for Linguistic Quality and Overall Quality, respectively.

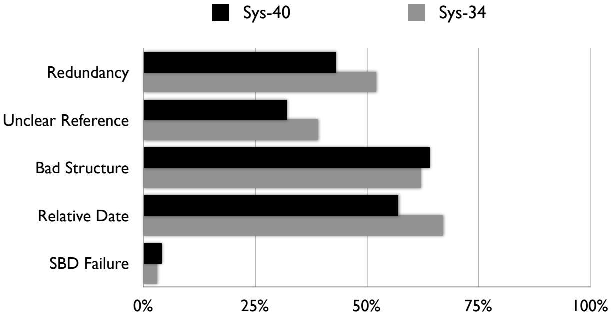

Figure 4: Summaries with linguistic quality scores below 5 (of 10) were reviewed. The x-axis indicates the fraction of summaries containing each type of error. Recall that Sys-40 employs sentence compression, while Sys-34 does not.

ence on Computational Natural Language Learning: Shared Task, pages 1–18.

D.S. Hochbaum. 1996. Approximating covering and packing problems: set cover, vertex cover, independent set, and related problems. *PWS Publishing Co. Boston, MA, USA*, pages 94–143.

E. Hovy, M. Marcus, M. Palmer, L. Ramshaw, and R. Weischedel. 2006. Ontonotes: the 90% solution. In *Proceedings of the Human Language Technology Conference of the NAACL*, pages 57–60.

T. Kiss and J. Strunk. 2006. Unsupervised Multilingual Sentence Boundary Detection. *Computational Linguistics*, 32(4):485–525.

D.D. Palmer and M.A. Hearst. 1997. Adaptive Multilingual Sentence Boundary Disambiguation. *Computational Linguistics*, 23(2):241–267.

S. Petrov and D. Klein. 2007. Learning and inference for hierarchically split PCFGs. In *Proceedings of the National Conference on Artificial Intelligence*, volume 22.

J.C. Reynar and A. Ratnaparkhi. 1997. A maximum entropy approach to identifying sentence boundaries. In *Proceedings of the Fifth Conference on Applied Natural Language Processing*, pages 16–19.

F. Schilder, R. Kondadadi, J.L. Leidner, and J.G. Conrad. 2008. Thomson Reuters at TAC 2008: Aggressive filtering with FastSum for update and opinion summarization. In *Proceedings of the Text Analysis Conference*.