Available online at www.sciencedirect.com
ScienceDirect
Information Processing and Management 43 (2007) 1606–1618
INFORMATION PROCESSING & MANAGEMENT
www.elsevier.com/locate/infoproman

Beyond SumBasic: Task-focused summarization with sentence simplification and lexical expansion

Lucy Vanderwende [a,*], Hisami Suzuki [a], Chris Brockett [a], Ani Nenkova [b,1]

[a] *Microsoft Research, One Microsoft Way, Redmond, WA 98052, USA*
[b] *Columbia University, New York, NY 10027, USA*

Received 18 July 2006; received in revised form 18 January 2007; accepted 22 January 2007
Available online 19 April 2007

Abstract

In recent years, there has been increased interest in topic-focused multi-document summarization. In this task, automatic summaries are produced in response to a specific information request, or topic, stated by the user. The system we have designed to accomplish this task comprises four main components: a generic extractive summarization system, a topic-focusing component, sentence simplification, and lexical expansion of topic words. This paper details each of these components, together with experiments designed to quantify their individual contributions. We include an analysis of our results on two large datasets commonly used to evaluate task-focused summarization, the DUC2005 and DUC2006 datasets, using automatic metrics. Additionally, we include an analysis of our results on the DUC2006 task according to human evaluation metrics. In the human evaluation of system summaries compared to human summaries, i.e., the Pyramid method, our system ranked first out of 22 systems in terms of overall mean Pyramid score; and in the human evaluation of summary responsiveness to the topic, our system ranked third out of 35 systems.

Keywords: Summarization; Multi-document summarization; Sentence simplification; Lexical expansion; Query expansion; NLP

1. Introduction

In recent years, there has been increased interest in topic-focused multi-document summarization. In this task, automatic summaries are produced in response to a specific information request, or topic, stated by the user. In response to this interest and to stimulate research in this area, the National Institute of Standards and Technology (NIST) conducted a series of workshops, the Document Understanding Conference[2] (DUC), to provide large-scale common data sets for the evaluation of systems. Participants in DUC2005 and DUC2006 were provided with a topic and a set of relevant documents (newswire articles), and the task was

* Corresponding author. Tel.: +1 425 706 5560; fax: +1 425 936 7329.
E-mail addresses: lucyv@microsoft.com (L. Vanderwende), hisamis@microsoft.com (H. Suzuki), chrisbkt@microsoft.com (C. Brockett), anenkova@stanford.edu (A. Nenkova).
[1] Present address: Stanford University, Stanford, CA 94305, USA.
[2] http://duc.nist.gov.

0306-4573/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ipm.2007.01.023

to produce an automatic summary of not more than 250 words in length. Consider the topic description for clusters D0652G and D0655A, for example:

D0652G: Identify the world's top banana producers and their levels of production. Describe the main markets for imports and relate any major issues in the industry.
D0655A: Discuss the causes, effects, and treatment of childhood obesity. How widespread is it?

In order to evaluate the summaries produced by the participants' systems, called *peer summaries*, DUC provides four human summaries, model summaries, for comparison.

Several methods for summarization evaluation have been proposed. The automatic metric used in DUC is ROUGE (Lin, 2004) specifically, ROUGE-2, which calculates the overlap in bigrams between the peer and the four model summaries, and ROUGE-SU4, which calculates the bigram overlap, but allowing up to 4 words to be skipped in order to identify a bigram match. Automatic metrics are useful as they potentially allow a comparison between different system settings, as we show in Section 6.1. However, since we are primarily interested in maximizing the content of our system's summaries, which, due to many issues of semantic realization such as paraphrase, cannot always be captured by measuring bigram overlap with four model summaries, we also evaluate our results with human evaluation metrics. Human evaluation is time-consuming and must be conducted carefully, in the same experimental setting, in order to ensure comparison across systems. We participated in DUC2006 in order to gain access to a human evaluation of our system, specifically the NIST content responsiveness score. In addition, a second human evaluation is coordinated by Columbia University, the Pyramid method (Nenkova et al., 2004). The Pyramid method requires two steps: first, a set of semantic equivalence classes are built for the sets of model summaries, with higher scores assigned to content represented in multiple model summaries, and second, a person identifies the content units in a peer summary that are found in the set of semantic equivalence classes.[3] The scores reported measure the amount of content overlap between the peer summary and the four model summaries.

In this paper, we describe the multi-document summarization system we submitted to DUC2006. Our contribution in DUC2006, identified as System 10, builds on an earlier system, SumBasic (Nenkova & Vanderwende, 2005) which produces generic multi-document summaries; we provide a description of SumBasic in Section 2. We then describe each of the remaining three main components that comprise our system: a task-focused extractive summarization system, sentence simplification, and lexical expansion of topic words. We will provide experiments using automatic metrics designed to quantify the contributions of each component. Human evaluation metrics, discussed in Section 6.2, indicate that this is a relatively successful approach to multi-document summarization; in the Pyramid evaluation, our system ranked first out of 22 systems and in the NIST metrics for content responsiveness, our system ranked third out of 35 systems.

With regard to our system design, it must be noted that this system, similar to almost all multi-document summarization systems, produces summaries by selecting sentences from the document set, either verbatim or with some simplification. Using sentence simplification is a step towards generating new summary text, rather than extracting summaries from existing text. There is, however, no consideration for sentence ordering or cohesion other than that sentence ordering is determined exclusively as a result of the sentence selection process (see Section 2 for details).

2. Core system: SumBasic

SumBasic (Nenkova & Vanderwende, 2005) is a system that produces generic multi-document summaries. Its design is motivated by the observation that words occurring frequently in the document cluster occur with higher probability in the human summaries than words occurring less frequently. Specifically, SumBasic uses the following algorithm:

[3] In order to have summaries evaluated according to the Pyramid method, each participant was required to volunteer annotation effort, which we did. In order to further contribute to the community effort, Microsoft Research assisted in the creation of the semantic equivalence classes. Creation of the semantic equivalence tasks, which requires the model summaries, was carried out after final test results had been submitted, and therefore, we did not have access to the model summaries at any time prior to the test.

Step 1. Compute the probability distribution over the words w_i appearing in the input, $p(w_i)$ for every i; $p(w_i) = \frac{n}{N}$, where n is the number of times the word appeared in the input, and N is the total number of content word tokens in the input.

Step 2. For each sentence S_j in the input, assign a weight equal to the average probability of the words in the sentence, i.e.,

$$\text{Weight}(S_j) = \sum_{Wi \in Sj} \frac{p(wi)}{|\{wi|wi \in Sj\}|}.$$

Step 3. Pick the best scoring sentence that contains the highest probability word.

Step 4. For each word w_i in the sentence chosen at step 3, update their probability:

$$p_{\text{new}}(w_i) = p_{\text{old}}(w_i) \cdot p_{\text{old}}(w_i).$$

Step 5. If the desired summary length has not been reached, go back to Step 2.

Steps 2 and 3 enforce the desired properties of the summarizer, i.e., that high frequency words from the input are very likely to appear in the human summaries. Step 3 ensures that the highest probability word is included in the summary, thus each time a sentence is picked, the word with the highest probability at that point in the summary is also picked. Step 4 serves a threefold purpose:

1. It gives the summarizer sensitivity to context. The notion of "what is most important to include in the summary?" changes depending on what information has already been included in the summary. In fact, while $p_{\text{old}}(w_i)$ can be considered as the probability with which the word w_i will be included in the summary, $p_{\text{new}}(w_i)$ is an approximation of the probability that the word w_i will appear in the summary twice.
2. By updating the probabilities in this intuitive way, we also allow words with initially low probability to have higher impact on the choice of subsequent sentences.
3. The update or word probability gives a natural way to deal with the redundancy in the multidocument input. No further checks for duplication seem to be necessary.

The system resembles SUM_{avr} as recently described in Nenkova, Vanderwende, and McKeown (2006) except that the update function in SumBasic uses squaring rather than multiplication by a very small number.

Comparing SumBasic to Luhn (1958). The idea that simple frequency is indicative of importance in automatic summarization dates back to the seminal work of Luhn (1958). The SumBasic algorithm is distinct from Luhn's algorithm, however, in several significant but illustrative ways. Luhn first collects the frequencies of words in the document and identifies a subset of *significant* words, excluding the most frequent (what would now be termed "stopwords") and the least frequent (generally, words occurring less than four times). Whereas SumBasic uses true initial probabilities and computes the weight of a sentence as equal to the average probability of the words in a sentence, Luhn treats all significant words as having equal weight and computes the weight of a sentence as a function of the concentration of significant words in the sentence. This weight is obtained by defining *windows* as sequences of significant and non-significant words, such that there are no more than four non-significant words between any two significant words in a window; the weight of a sentence is equal to the weight of the highest scoring window, namely, the square root of the number of significant words in the window, divided by the total number of words in the window. Finally, since Luhn's system is designed to summarize single-documents, there is little need to prevent redundancy, in sharp contrast to multidocument summarization, where the likelihood that several documents might convey highly similar, or even identical, important information necessitates mechanisms to avoid redundancy, a functionality that SumBasic provides by updating the probability of the words on the basis of preceding selected sentences.

3. SumFocus

Nenkova and Vanderwende (2005) document the impact of frequency on generic multi-document systems, but in the case of task-focused summarization, document frequency alone may not be predictive, since the

topic may not be related to the most common information in the document set. Thus, in order to incorporate topic constraints in generic summarization, our new system, called SumFocus, captures the information conveyed by the topic description by computing the word probabilities of the topic description. Having done so, the weight for each word is computed as a linear combination of the unigram probabilities derived from the topic description, with backoff smoothing to assign words not appearing in the topic a very small probability, and the unigram probabilities from the document, in the following manner (all other aspects of SumBasic remain unchanged):

$$\text{WordWeight} = (1 - \lambda) * \text{DocWeight} + \lambda * \text{TopicWeight}.$$

The optimal value of λ, 0.9, was empirically determined using the DUC2005 corpus, manually optimizing on ROUGE-2 scores (henceforth R-2).

Since sentence selection is controlled by choosing the words with the highest weights, it is, in principle, possible for these "best words" to come from either the document or from the topic description. In practice, however, the best word is nearly always a word from the topic description due to the high value assigned to λ. For DUC2005 overall, 618 document words were identified as best on the basis of topic statements and only 22 independently on the basis of frequency alone. For DUC2006 overall, all 600 best words were chosen from the topic statements. On the basis of DUC2005 data, we added a small list of "topic stopwords" (*describe*, *discuss*, *explain*, *identify*, *include*, *including*, *involve*, *involving*) that did not receive any weight. The topic statements in DUC2006 appear to contain more instructions to the summarizer than in DUC2005, suggesting additional words may warrant similar handling (e.g., *concerning*, *note*, *specify*, *give*, *examples*, and *involved*).

4. Sentence simplification

Our goal is to create a summarization system that produces summaries with as much content as possible that satisfies the user, given a set limit on length.[4] Since summaries produced by SumFocus alone are extractive, we view sentence simplification (also known as *sentence shortening* or *sentence compression*) as a means of creating more space within which to capture important content.

4.1. Approaches to simplification

The most common approach to sentence simplification for summarization purposes has been to deterministically shorten the sentences selected to be used in the summary. The CLASSY system (Conroy, Schlesinger, & Goldstein Stewart, 2005) for example, incorporates a heuristic component for sentence simplification that preprocesses the sentences used in their sentence selection component. Columbia University's summarization system uses a syntactic simplification component (Siddharthan, Nenkova, & McKeown, 2004) the results of which are sent to their sentence clustering component. Daumé and Marcu (2005a) employ a post-processing approach and report that deleting adverbs and attributive phrases improve ROUGE scores in the Multilingual Summarization Evaluation, although this post-processing was not found to be useful in DUC2005 (Daumé & Marcu, 2005b) conceivably because the summaries are 250 words long rather than 100 words long.

In these approaches, simplification operations apply to all sentences equally, and the core sentence selection component has only either the original or the shortened sentence available to choose from. This may not be optimal, however, because the best simplification strategy is not necessarily the same for all sentences. For example, it might be desirable to delete material X from a sentence only if X is already covered by another sentence in the summary. For this reason, simplification strategies have so far remained conservative, presumably to avoid possible side-effects of oversimplification.

An alternative approach to sentence simplification is to provide multiple shortened sentence candidates for the summarization engine to choose from. Multi-Document Trimmer (Zajic, Dorr, Lin, Monz, & Schwartz, 2005) for instance uses a syntactic simplification engine (Dorr, Zajic, & Schwartz, 2003) initially developed for headline generation, to output multiple simplified versions of the sentences in the document cluster. Each

[4] For DUC2005 and DUC2006, the summary length was no more than 250 words.

Table 1
Syntactic patterns for sentence simplification (underlined parts are removed)

Pattern	Example
Noun appositive	One senior, Liz Parker, had slacked off too badly to graduate
Gerundive clause	The Kialegees, numbering about 450, are a landless tribe, sharing space in Wetumka, Okla., with the much larger Creek Nation, to whom they are related
Nonrestrictive relative clause	The return to whaling will be a sort of homecoming for the Makah, whose real name which cannot be written in English means "people who live by the rocks and the seagulls"
Intra-sentential attribution	
Lead adverbials and conjunctions	Separately, the report said that the murder rate by Indians in 1996 was 4 per 100,000, below the national average of 7.9 per 100,000, and less than the white rate of 4.9 per 100,000

of these candidates are submitted to the feature-based sentence selection component, which includes the redundancy score of the sentence given the current state of the summary and the number of trimming operations as features.

4.2. Simplified sentences as alternatives

Our approach to sentence simplification most closely resembles that used in the Multi-Document Trimmer (Zajic et al., 2005): we apply a small set of heuristics to a parse tree to create alternatives, after which both the original sentence and (possibly multiple) simplified versions are available for selection. Unlike MDT, however, in our system both original and alternative simplified sentences are provided for selection without differentiation, i.e., without retaining any explicit link between them, under the assumption that the SumBasic-based multi-document summarization engine is inherently equipped with the ability to handle redundancy, and the simplified alternatives only add to that redundancy. SumBasic's method for updating the unigram probabilities given the sentences already selected allows the simplified sentence alternatives to be considered independently, while maintaining redundancy at a minimum.[5] Given that this approach to sentence simplification allows the sentence selection component to make the optimal decision among alternatives, we are thus freed to pursue more aggressive simplification, since the original non-simplified version is always available for selection. The approach is also extensible to incorporating novel sentence rewrites into a summary, moving in the direction of generative rather than extractive summaries. An example of this is Jing and McKeown (2000) who propose a set of operations to edit extracted sentences, not limited to sentence reduction. It is straightforward to integrate such sentence rewrite operations into our framework, as candidate generation works independently of sentence selection, and word probability alone suffices to compute the sentence score.

4.3. The syntax-based simplification filter

Our simplification component consists of heuristic templates for the elimination of syntactic units based on parser output. Each sentence in the document cluster is first parsed using a broad-coverage English parser (Ringger, Moore, Charniak, Vanderwende, & Suzuki, 2004). We then run a filter on the parse tree that eliminates certain nodes from the parse tree when the node matches the patterns provided heuristically. Table 1 lists the syntactic patterns used in our DUC2006 system. These patterns are inspired by and similar to those discussed in Dunlavy et al. (2003) the principal difference being that extraction makes use of a full-fledged syntactic parser rather than employing a shallow parsing approach. For the first three patterns in Table 1 (noun appositive, gerundive clause and non-restrictive relative clause), the parser returns a node label corresponding exactly to these patterns; we simply deleted the nodes with these labels. For the identification of intra-sentential attribution, we added specific conditions for detecting the verbs of attribution (*said* in Table 1), its subject (*the report*), the complementizer (*that*) and adverbial expressions if any, and deleted the nodes when conditions were matched. In the case of sentence-initial adverbials, we delete only manner and time adverb expressions,

[5] Note, however, that the probability update by SumBasic must be computed based on the original document cluster.

using the features returned by the parser. Currently, these patterns all apply simultaneously to create maximally one simplified sentence per input, but it would in principle be possible to generate multiple simplified candidates. Finally, the punctuation and capitalization of the simplified sentences are cleaned up before the sentences are made available to the selection component along with their original, non-simplified counterparts in the document cluster. The results of applying this simplification filter are discussed in Section 6.

5. Lexical expansion

In addition to sentence simplification, we also investigated the potential for expanding the task terms with synonyms and morphologically-related forms. The use of query expansion has frequently been explored in Information Retrieval tasks, but without notable success (Mitra, Singhal, & Buckley, 1998). However, since the purpose of summarization is not to extract entire relevant documents from a large data set, but smaller sentences, it might be hypothesized that individual expansions could have more evident impact. In constructing our system, therefore, we augmented the task terms with lexical expansions supplied by morphological variants (chiefly derived forms) and synonyms or closely-related terms drawn from both static hand-crafted thesauri and dynamically-learned sources extracted from corpora.

Lexical expansions are applied only at the point where we choose the inventory of ''best words'' with which to determine candidate sentence selection. As already noted in Section 3, in DUC2006, the ''best words'' are drawn only from the topic statements, and not from the document collection. Where a term in the sentence matches a lexical expansion, the formula for computing ''best word'' scores is as follows, where d is the document score of the lexical item in question, and e is the score for each type of matched expansion:

$$\text{Expansion Score} = (1-\lambda) * d + \lambda \sum_{i}^{n} e_i.$$

The summation $\sum_i^n e_i$ is the expanded analog of the *TopicWeight* in the formula given for unexpanded topics in the discussion of SumFocus in Section 3. The λ is a uniform default weight applied to the cumulative scores of matches from all expansion types. For the purposes of our DUC 2006 submission, the value of λ was set at 0.5 after hand inspecting results for R-2 while tuning the system on the DUC 2005 dataset. Lexical expansion scores were not used to recompute the lexical probabilities of the sentences once they had been selected.

5.1. Morphological variants

Morphologically-derived forms were looked up in the version of the American Heritage Dictionary used by our parser, thereby allowing us to obtain pairs such as *develop* ↔ *development*. We also employed a small inventory of paired geographical names and their adjective counterparts, e.g., *United States* ↔ *American*, *China* ↔ *Chinese*, *Tanzania* ↔ *Tanzanian*. Expansion scores for these morphological variants were computed on the basis of simple occurrence counts of the number of times the forms were encountered in the task description. Thus, if both forms appeared in the topic text, both received a boost. Since our implementation did not lemmatize the topic words, only exact matches were considered.

5.2. Learned thesaurus

A primary objective in investigating lexical expansions was to explore the potential impact of a 65,335-pair synonym list that we automatically acquired from clustered news articles available on the World Wide Web, the hypothesis here being that a thesaurus derived from news data might be prove more useful, and potentially domain-relevant than static general-domain synonym resources. Starting with an initial dataset of 9.5 million sentences in ~32,400 pre-clustered news articles, we created a monolingual bitext of ~282,600 aligned sentence pairs using a crude heuristic similarity measure based on three criteria, namely, that the sentence pairs should have at least three words in common, have a minimum sentence length ratio of 66.6%, and have a word-based edit distance (i.e,, number of words inserted and deleted) of $e \leqslant 12$. The reader is referred to Quirk, Brockett, and Dolan (2004) Dolan, Quirk, and Brockett (2004) for further information about the construction of this corpus. This was augmented by a further ~20,000 sentence pairs extracted using a Support Vector Machine

Table 2
Sample aligned sentence pair used to extract word associations

Indonesia's electoral commission has formally announced that Susilo Bambang Yudhoyono, a former general and security minister, has won the country's first direct presidential election
Indonesian election officials today formally declared ex-general Susilo Bambang Yudhoyono as victor in the country's first presidential poll after a final vote tally was completed

applying lexical bootstrapping methods described in Brockett and Dolan (2005) who demonstrate that it is possible to apply Giza++ (Och & Ney, 2003) to this (more or less comparable) corpus to yield an overall word Alignment Error Rate as low as 12.46%. Sentence pairs obtained in this fashion typically exhibit substantive semantic overlap, providing a dataset from which a potentially useful thesaurus might be extracted. An example sentence pair is shown in Table 2.

The paired sentences were then tokenized, named entities and multiple word expressions were identified, and the words then lemmatized and tagged for part of speech. Identical words were deleted from the pairs and the remainder aligned using a Log Likelihood Ratio-based Word Association technique described in Moore (2001) using the formula given in Moore (2004) modified here for readability:

$$LLR(t,s) = \sum_{t\in\{1,0\}} \sum_{s\in\{1,0\}} C(t,s) \log \frac{p(t|s)}{p(t)},$$

where t and s are variables ranging over the presence (1) or absence (0) of the two words under consideration, and $C(t,s)$ is the observed joint count for their values. The probabilities used are maximum likelihood estimates.

After extraction, the word pairs were further filtered to remove typographical errors, mismatches relating to numerical expressions and other artifacts of unnormalized news data. The pairs were then chained up to three steps to expand the dataset, and the association scores coerced into a range between 1.0 and 0.0 in order to generate a distribution that could be utilized by the system. These initial weights were then updated at runtime in the same manner as for morphological variants, by multiplying them by the number of times either member of a pair occurred in the topic description. No attempt was made to discriminate among word senses, in particular, where chaining might have resulted in mismatched terms. (In general, such mismatches would tend to have very low scores.) Example synonyms and near synonyms for the word "virus" extracted in this manner are shown in Table 3. These range over two major senses, and also are not limited to the same part of speech. On the other hand, the data acquired in this manner can be relatively limited, and does not include, for example, names of major computer viruses, a limitation that in fact appears to have affected results in some newsclusters.

5.3. Static thesauri

In addition to the learned synonyms, we also considered expansions using data extracted from two static thesauri. The first was a list of 125,054 word pairs in the *Encarta Thesaurus* (Rooney, 2001) which we deployed

Table 3
Synonyms of the noun "virus" learned from monolingual parallel corpora with initially assigned weights

Synonym	Weight	Synonym	Weight
Coronavirus	0.173206	Heart_disease	0.018887
Disease	0.113415	Severe	0.018278
Computer_virus	0.105600	Infect	0.014361
Worm	0.094918	SARS	0.013873
Illness	0.051918	West_Nile_Virus	0.006470
Cancer	0.038996	Condition	0.006335
Epidemic	0.038778	Disease-causing	0.004464
Viral	0.023326	HIV	0.001398
Flu	0.020093		

in the submitted system. Heuristic weights were precomputed in the form of a distribution based on number of synonyms for each word, allowing this data to be easily integrated with our acquired Word Association list.

We also experimented with, but did not include in the system submitted to DUC 2006, simple undisambiguated synonym expansion using WordNet 2.0 (Fellbaum, 1998) No attempt was made to provide weights for the lexical expansions found in WordNet; instead raw occurrence counts were used as with morphological variants, with the uniform λ applied in the same manner as to the other resources.

6. Results

6.1. Automatic evaluation using ROUGE

Table 4 compares the average ROUGE recall scores for word unigram (R-1), word bigram (R-2) and skip-4 bigram (R-SU4) models achieved on DUC 2005 data (used in training) and on DUC 2006 data by different versions of our system.[6] Numbers for SumBasic are presented as baseline.

It is difficult to derive meaningful conclusions on the basis of the ROUGE results in Table 4. On DUC 2005 data, sentence simplification consistently appears to improve matters modestly, but on DUC 2006 data it paradoxically seems to introduce a small degradation in recall when deployed in conjunction with SumFocus. This appears to be partially offset by the application of lexical expansion. However, the differences in scores consistently fall within the 95% error margins computed by the ROUGE tool. Moreover, *p*-scores computed using the Wilcoxon Matched-Pairs Signed-Ranks Test indicate that differences over a baseline Sumbasic system without simplification were significant ($p < 0.05$) only on the DUC 2005 data used in training.

Despite the somewhat inconclusive nature of the ROUGE results, they are suggestive of the relative contributions of different system components. Table 5 presents the individual contributions made by various expansion strategies when used in conjunction with sentence simplification. The biggest observable impact comes from morphological expansion (MRF), while other lexical expansions, namely the Encarta Thesaurus (ENC) and Word Association data (WA) may have contributed to the overall performance of the system to a lesser extent, if at all. MRF + ENC + WA corresponds to the system submitted to DUC 2006, i.e., SumFocus with lexical expansion and simplification. None of the differences are statistically significant at the level of $p < 0.05$ on the Wilcoxon Matched-Pairs Signed Ranks Test. Table 5 also shows the potential impact of simple synonym expansion using WordNet 2.0 (Fellbaum, 1998) which was not included in the submitted system, but which performs at or below baseline level in the table.

6.2. Pyramid and NIST

Results obtained from human evaluations are potentially more diagnostic and can inform the direction that future work should take. Unfortunately, however, NIST or Pyramid, as one-time human evaluations, do not lend themselves to comparing system settings. Nevertheless, since both NIST and Pyramid evaluation techniques measure content directly, these are the metrics we focus on, given our primary goal of maximizing summary content. Accordingly, we report NIST and Pyramid metrics only for the system we submitted, which is SumFocus with sentence simplification and with lexical expansion of the topic words.

Pyramid evaluation. System 10 ranked first in the overall mean Pyramid score of the 22 systems that participated in the Pyramid evaluation. It must be noted, however, that the maximum Pyramid score for each cluster differs (Nenkova et al., 2004) since the agreement observed in the model summaries varies from cluster to cluster, and so we find that the average rank is a better method to compare systems, presented in Table 6. System 10 is ranked first for 5 out of 20 clusters, and is in the top 3 for half of the clusters. Overall, our per-cluster mean ranking (5.90) is the best among the 22 systems. On the other hand, our performance across the

[6] DUC reports ROUGE numbers that are computed implementing the "jackknife" in which each human model summary is successively removed from the evaluation and added to the system ("peer") summaries. This permits the human summaries to be directly compared with the system summaries using the same metric. For all experiments in this paper, however, we compute ROUGE without jackknifing, which allows us to use 4 model summaries. As a result, the official DUC 2006 scores vary slightly from the numbers reported here; the differences, however, are not significant.

Table 4
ROUGE results, with and without sentence simplification (not using stopwords)

		DUC 2005			DUC 2006		
		Not simplified	Simplified	$p \leqslant$	Not simplified	Simplified	$p \leqslant$
	SumBasic	0.25605	0.26054		0.30026	0.30753	
R-1	SumFocus	0.25358	0.26011		0.29995	0.29693	
Recall	SumFocus + expansion	0.25489	0.26006		0.30064	0.29943	
	SumBasic	0.03642	0.03653		0.05326	0.05509	
R-2	SumFocus	**0.04061**	**0.04267**	0.039	**0.05900**	0.05725	
Recall	SumFocus + expansion	0.04104	0.04244		0.05896	**0.05949**	0.055
	SumBasic	0.06631	0.06721		0.08572	0.08681	
R-SU4	SumFocus	0.06974	**0.07232**	0.031	**0.09107**	0.08852	
Recall	SumFocus + expansion	**0.07049**	0.07221		0.09093	**0.08998**	0.125

Table 5
Contributions of different lexical expansion components on DUC2006 data (with sim-plification, not using stopwords)

	R-1	R-2	R-SU4
SumFocus (Baseline)	0.29693	0.05725	0.08852
ENC	0.29642	0.05719	0.08841
WN	0.29575	0.05729	0.08834
WA	0.29678	0.05736	0.08849
MRF	**0.29862**	**0.05918**	**0.08979**
MRF + ENC + WA	**0.29943**	**0.05949**	**0.08998**

ENC: Encarta Thesaurus, MRF: morphological variants, WA: word Association, WN: WordNet.

Table 6
Pyramid and NIST results for system 10, including the rank per cluster according to Pyramid evaluation, average number of unweighted SCUs out of the average number of SCUs attainable for each cluster, and the NIST content responsiveness score, which is on a scale of 5 to 1

Cluster	Rank acc. to Pyramid score	System 10 SCUs/average SCUs attainable	NIST content responsiveness (5 = very good)
D0601	1	0.3056	1
D0603	2	0.1818	3
D0605	6	0.0976	2
D0608	1	0.3200	3
D0614	10	0.1739	5
D0615	1	0.2500	1
D0616	3	0.3043	4
D0617	1	0.3409	2
D0620	2	0.3103	3
D0624	1	0.5000	3
D0627	15	0.1429	1
D0628	15	0.1613	1
D0629	21	0.0435	2
D0630	13	0.1935	3
D0631	1	0.5625	4
D0640	12	0.2564	2
D0643	6	0.3077	2
D0645	8	0.1786	3
D0647	6	0.1600	2
D0650	3	0.2750	2

clusters, also shown in Table 6, is not evenly distributed; system 10 scores worse than half of the systems for 5 out of 20 clusters. Table 6 also presents the NIST content responsiveness score, i.e., the human judgment of

Table 7
NIST evaluation results

	System 10	Avg. peer	System 10 rank
Grammaticality	3.12	3.58	31
Non-redundancy	4.42	4.23	10
Referential clarity	2.64	3.11	31
Responsiveness	2.94	2.54	3

how well the content responds to the needs expressed in the topic description, on a scale from 1 to 5, where 5 is the highest achievable score. Overall, System 10 ranked third on responsiveness. On a per-cluster basis, providing a system ranking according the NIST content score is less than instructive since there are only 5 values that can be given. Initial investigation showed there is no correlation between Pyramid score and NIST content score, but we have reached no conclusion yet. It is unexpected that for cluster D0601, system 10 is ranked highest among the peers in the Pyramid evaluation, with a relatively high degree of SCU overlap, while receiving a poor NIST content score for the same cluster.

Looking at the Pyramid analysis a bit more deeply, we also computed the percentage of the number of SCUs that were scored in System 10 and the number of SCUs that could be attained within the 250 word limit. We note that for only one cluster are more than half of the attainable SCUs found in System 10's peer summaries. Improving this percentage is a clear direction for future research, though it remains an open question whether improving this percentage will lead to higher content responsiveness scores.

NIST evaluation. Three of the five NIST linguistic quality questions are relevant to sentence simplifcation: grammaticality, non-redundancy and referential clarity, along with the responsiveness question. NIST evaluated these questions by eliciting human judgments on a five point scale. Table 7 shows our DUC2006 scores relative to the peers, along with the rank of our system among 35 peer systems.

Though our system (System 10) implements only a small number of simplification patterns, as described in Section 4, its effect was quite extensive. In DUC2006, of all the sentences selected in summary, 43.4% of them were the result of simplification. This resulted in adding on average another sentence to the summary: the average number of sentence in a summary increased from 11.32 to 12.52 when we used sentence simplification.

7. Discussion

7.1. Contribution of simplification

The Pyramid results suggest that making room for more content and removing redundant material by simplifying sentences is a promising operation for extractive summarization systems. It is also interesting to note that 33.6% of the sentences selected in the summaries were original non-simplified versions, even though a simplified counterpart was also available. Manual investigation of the summary for one cluster (D0631D) established that four out of eleven sentences were non-simplified despite the availability of a simplified alternative. These four sentences are shown in Table 8. Of these, two were incorrectly parsed as noun appositives, resulting in an unexpectedly large portion of the text being deleted and rendering the simplified version less likely to be selected. The other two sentences present interesting cases where the deleted portion of text (indicated in boldface in Table 8) included important content, corresponding to Summary Content Units (SCUs) observed in three or four of the model summaries according to the Pyramid evaluation. This manual examination confirms that the best simplification strategy may not be the same in all cases, and that there are benefits in using the summarizer itself to choose the best sentence alternative given the context.

Sentence simplification undoubtedly contributes to our low score for grammaticality. In part this may be because the parser-based simplification filter can produce ungrammatical sentences and sentences with degraded readability owing to misplaced punctuation marks. However, grammaticality judgments may also be affected by whether or not a system allows incomplete last sentences. Since the length limit in DUC2006 was 250 words, some systems kept the summaries below 250 words, but permitted only complete sentences, while other systems truncated the summary at 250 words, even if the last sentence was incomplete. SumFocus

Table 8
Examples of full sentences chosen instead of their simplified counterpart

Reason: parser error
London British aviation authorities on Wednesday formally ruled the Concorde supersonic airliner unfit to fly unless its manufacturers took steps to prevent the problems that led to last month's fatal Air France Concorde crash near Paris
Le Figaro newspaper on Wednesday quoted Gayssot, the transport minister, as raising the possibility that the ban on Air France Concorde flights could remain in place until the Accident and Inquiry Office releases a preliminary report on the crash at the end of August
Reason: deleted material contains important information
Paris **French investigators looking into the crash** last month of an Air France Concorde said Thursday it was probable that a 16-in. piece of metal found on the runway caused a tire to blow out, sending debris from the tire through fuel tanks and triggering a fire that brought down the plane
The sleek, needle-nosed aircraft could cross the Atlantic at an altitude of 60,000 feet and at 1350 mph, **completing the trip from London to New York in less than four hours – half the time of regular jets**

Underlined portion of text was deleted in the simplified sentence. Text corresponding to an SCU is indicated by boldface.

falls into this latter category. Only 2 systems allowing incomplete last sentences scored higher on grammaticality than those with complete last sentences, suggesting that while the low scores for grammaticality may be due to sentence simplification, they may also be a function of system design.

Referential quality may also be negatively affected by the current simplification filter, since deletion of intra-sentential attribution can result in deleting the antecedent of pronouns in the summary. On the other hand, it is encouraging to note that our methods perform well on non-redundancy and content responsiveness. In particular, providing sentence alternatives did not increase redundancy, even though the alternatives were not explicitly linked, indicating that the method of updating unigram weights given context used in SumFocus is robust enough to handle the greater redundancy introduced by providing simplified alternatives.

7.2. Contribution of lexical expansion

It is difficult to argue on the basis of either human or automated evaluation that lexical expansion significantly enhanced the performance of SumFocus, even when using simplification. We earlier noted the paucity of evidence that thesaurus-based query expansion (as opposed to adding search terms) significantly improves the relevance of IR search results: Voorhees (1988) has suggested that use of WordNet may improve results of short queries, but not of longer ones. Since the DUC2006 tasks constitute moderately extended queries, it is perhaps unsurprising that the expansions had minimal effect. It is likely also that system-related factors mitigated the impact of lexical expansions: for example, it is probable that the means by which the expansion scores were computed may have resulted in many word pairs being scored too low to have any effect, while the use of a uniform expansion λ failed to differentiate different expansion types adequately.

Analysis of ROUGE scores for individual clusters indicates that depending on the domain and type of task, results were in fact sometimes negatively impacted by the approach taken in SumFocus as opposed to SumBasic. In DUC 2006, we observed low individual cluster scores when topic descriptions required the summary to identify a class of items: for example, a description that contains the instruction "Identify computer viruses detected worldwide," is expected to yield sentences containing the names of specific computer viruses, such as *Melissa, Love Bug* and *CIH*, rather than multiple exemplars of the term *computer virus*. Here, it seems that even the use of dynamically-learned word associations was inadequate to this domain-specific task. In the long term, however, acquisition of better domain-relevant resources might be expected to ameliorate system performance in such cases.

8. Conclusions and future work

The Pyramid annotation shows us that only rarely does a peer summary match 50% or more of the content in the combined model summaries. A detailed analysis of the percentage of SCUs per weight must still be

done, but anecdotally, our system matches only half of the high-scoring SCUs, i.e., those SCUs that were found in all of the model summaries. Clearly, finding methods to model this data more closely offers opportunities for improving the overall content of summaries. One direction will lead us to find more sophisticated methods of modeling the relation between a topic word and any of its lexical expansions, the document words, and the target summaries. And, as we continue to leave purely extractive summarization behind and pursue summarization that generates novel sentences that better capture the information need of the user, we will also expand our system to take full advantage of sentence simplification component. In particular, we plan to include more drastic simplifying and rewriting operations (such as splitting coordinated clauses) and produce multiple candidates per sentence in order to see the full potential for the proposed approach.

Acknowledgments

We are indebted to Arul Menezes for implementing several features described here. We also thank the NIST team that supports DUC for the NIST evaluation, and we thank the DUC community who contributed to the Pyramid evaluation. We extend our appreciation to members of the Butler-Hill Group, especially Ben Gelbart, for their assistance with the Pyramid creation and annotation. Finally, we are grateful to the anonymous reviewers for their many constructive comments.

References

Brockett, C., & Dolan, W. B. (2005). Support vector machines for paraphrase identification and corpus construction. In *Proceedings of the third international workshop on paraphrasing (IWP2005)*, Jeju, Republic of Korea.

Conroy, J. M., Schlesinger, J., & Goldstein Stewart, J. (2005). CLASSY query-based multi-document summarization. In *Proceedings of DUC 2005*.

Daumé, H., III, & Marcu, D. (2005a). Bayesian multi-document summarization at MSE. In *Proceedings of MSE 2005*.

Daumé, H., III, & Marcu, E. (2005b). Bayesian summarization at DUC and a suggestion for extrinsice evaluation. In *Proceedings of DUC 2005*.

Dolan, W. B., Quirk, C., & Brockett, C. (2004). Unsupervised construction of large paraphrase corpora: exploiting massively parallel news sources. In *Proceedings of COLING 2004*, Geneva, Switzerland.

Dorr, B., Zajic, D., & Schwartz, R. (2003). Hedge trimmer: a parse-and-trim approach to headline generation. In *Proceedings of HLT-NAACL 2003 text summarization workshop* (pp. 1–8).

Dunlavy, D., Conroy, J., Schlesinger, J., Goodman, S., Okurowski, M., O'Leary, E., et al. (2003). Performance of a three-stage system for multi-document summarization. In *Proceedings of DUC 2003*.

Fellbaum, C. (Ed.). (1998). *WordNet: An electronic lexical database*. The MIT Press.

Jing, H., & McKeown, K. (2000). Cut and past based text summarization. In *Proceedings of the 1st conference of the North American Chapter of the Association for Computational Linguistics (NAACL'00)*.

Lin, C. Y. (2004). ROUGE: a package for automatic evaluation of summaries. In *Proceedings of the workshop on text summarization branches out*, 25–26 July 2004, Barcelona, Spain.

Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of Research and Development, 2*(2), 159–165.

Mitra, M., Singhal, A., & Buckley, C. (1998). Improving automatic query expansion. In *Proceedings of the 21st Annual International ACM-SIGIR conference on research and development in information retrieval*.

Moore, R. C. (2001). Towards a simple and accurate statistical approach to learning translation relationships among words. In *Proceedings, workshop on data-driven machine translation*, Toulouse, France.

Moore, R. C. (2004). Association-based bilingual word alignment. In *Proceedings, workshop on building and using parallel texts: data-driven machine translation and beyond*, Ann Arbor, MI.

Nenkova, A., & Vanderwende, L. (2005). *The impact of frequency on summarization*. MSR-TR-2005-101.

Nenkova, A., Vanderwende, L., & McKeown, K. (2006). A compositional context sensitive multidocument summarizer. In *Proceedings of SIGIR 2006*.

Nenkova, A., & Passonneau, R. (2004). Evaluating content selection in summarization: the Pyramid method. In *Proceedings of the HLT-NAACL 2004*.

Och, F. J., & Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics, 29*(1), 19–52.

Quirk, C., Brockett, C., & Dolan, W. B. (2004). Monolingual machine translation for paraphrase generation. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, 25–26 July 2004, Barcelona Spain (pp. 142–149).

Ringger, E., Moore, R. C., Charniak, E., Vanderwende, L., & Suzuki, H. (2004). Using the Penn treebank to evaluate non-treebank parsers. In *Proceedings of LREC 2004*.

Rooney, K. (2001). *Encarta Thesaurus*. Bloomsbury Publishing.

Siddharthan, A., Nenkova, A., & McKeown, K. (2004). Syntactic simplification for improving content selection in multi-document summarization. In *Proceedings of COLING 2004*.

Voorhees, E. (1988). Using WordNet for text retrieval. In C. Fellbaum (Ed.), *WordNet: An electronic lexical database*. The MIT Press.

Zajic, D., Dorr, B., Lin, J., Monz, C., & Schwartz, R. (2005). A sentence-trimming approach to multi-document summarization. In *Proceedings of DUC2005*.