Summarizing Large Document Sets Using Concept-Based Clustering

Hilda Hardy†, Nobuyuki Shimizu†, Tomek Strzalkowski†, Liu Ting†,
G. Bowden Wise‡, and Xinyang Zhang†

†NLIP Laboratory, University at Albany, SUNY
1400 Washington Avenue
Albany, NY 12222

‡GE Global Research Center
1 Research Circle
Niskayuna, NY 12309

hardyh@cs.albany.edu

ABSTRACT

This paper describes our multi-document summarizer XDoX designed to summarize large sets of documents (50-500). These documents are typically obtained from routing or filtering systems run against a continuous stream of data, such as a newswire. XDoX identifies the most salient or often-repeated themes within the set and composes an extraction summary reflecting these main themes. The summarizer uses a unique *n*-gram scoring method to give greater importance to clusters of passages that have significant common phrases. Our methods are robust, topic-independent, and easily extensible to multilingual applications. We show examples of summaries obtained in our tests as well as from our participation in the first Document Understanding Conference (DUC).

1. RELATED WORK

An automated multi-document summarization system requires techniques different from those necessary to summarize a single document. Many newswire articles on the same topic, for example, are likely to contain redundant material. Articles may reveal new developments of events over time, or they may include various information and opinions on an issue or a person. Most current automated systems work by identifying the most important sentences or paragraphs in the set of documents and building a summary with these passages. Several systems use Carbonell and Goldstein's Maximal Marginal Relevance measure [1], which selects passages based on a combination of relevance and anti-redundancy. Columbia's SIMFINDER [4] uses several varieties of word overlap in addition to other features to determine similarity values between passages in order to form clusters, followed by sentence extraction (CENTRIFUSER) or reformulation (MULTIGEN, FUF/SURGE). Radev et al's WebInEssence [12] clusters documents according to keyword overlap and performs centroid-based sentence extraction. Marcu [8] selects important sentences based on the discourse structure of the text. Lin and Hovy's NEATS system [7] creates a query based on single terms, bigrams and trigrams most characteristic of each document set, finally producing a ranked list of sentences. TNO's system [6] scores sentences by combining a unigram language model approach with a Bayesian classifier based on surface features. Our XDoX system clusters passages based on our unique *n*-gram scoring methods, and forms an extraction summary based on a representative passage from each cluster.

2. XDOX OVERVIEW

The XDoX system (Cross Document Summarizer) was built for information analysts and designed to summarize sets of 50-500 documents that have been retrieved or routed from a text database, the Internet, or a news source, according to a query or a user-defined profile. The system uses clustering techniques to subdivide the documents into groups of passages representing meaningful topics and themes, and to separate unrelated material. XDoX presents the user with two kinds of overall summary, one with more detail related to the complexity of the document set, and one with fewer details and limited length. In addition, a Graphical User Interface allows the user to view individual passages, full documents, and a summary of each topical cluster. Thus the system answers both the indicative and the exploratory needs of our customers.

XDoX has been developed over the last two years with several intermediate designs considered. An early version of the system [15] produced clusters of documents according to their mutual similarity, which was based primarily on straightforward term overlap between documents. In addition, WordNet lexical database [2] [10] was used to facilitate the matching of synonyms and other related terms. This early work focused on evaluating various document clustering techniques, but we found that most known clustering methods could not alone support an effective summarizer. Whether or not WordNet is used, the resulting clusters were often of poor quality, formed around common but semantically unimportant terms.

Our new approach improves the quality of the clusters and the summaries by implementing concept-based clustering and summarization. Instead of clustering entire documents or summaries of documents, we cluster passages [11], or sequences of text, which usually correspond to the natural paragraphs designed by the author or editor, or that may be obtained automatically [5]. Our similarity metric is based on *n*-gram matching rather than just single-term overlap. For example, a word in a sequence of six words receives proportionately more weight than the same word occurring in two distinct three-word sequences, which in turn is weighted more heavily than if it were found in three bi-grams, and so on. Individual term weights are computed at the document level using a variant of pivoted document length normalization metric [14] and added to *n*-gram weights.

We have found that our new approach produces excellent summaries in most cases. The summaries are based on high-quality clusters that form around significant common concepts or themes that occur repeatedly across the set of documents. The paragraph is a useful semantic unit for conceptual clustering, because most writers view a paragraph as a topical unit, and organize their thoughts accordingly. A few examples of themes detected in large document sets are: “lie detector test”, “South Georgia and South Sandwich Islands”, and “blood alcohol levels”.

3. XDOX SYSTEM ARCHITECTURE

The XDoX system is written in Java, except for our single document summarizer, which is written in C++. The system has several distinct modules. Their basic tasks, inputs and outputs are shown in Figure 1.

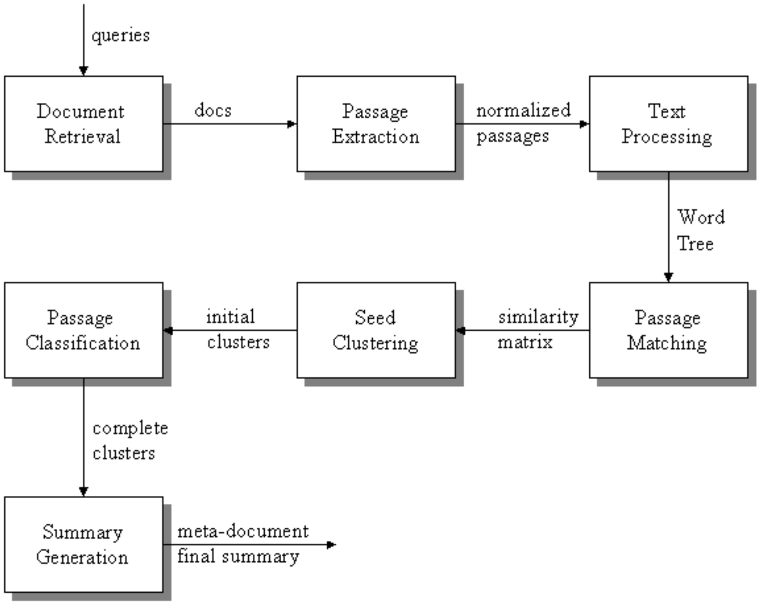

Figure 1. XDoX System Architecture

4. PASSAGE EXTRACTION, TEXT PROCESSING

Documents are first chunked into passages, according to existing paragraph boundaries. Passages are normalized by removing SGML tags, lists of keywords, author by-lines, news sources, locations, etc. Subtitles and one-line paragraphs are merged with the following text.

In preparation for comparing *n*-grams in passages, the Text Processing module removes stopwords and stems the remaining words, using the Porter algorithm. Each stem is mapped to a unique integer code. The stemmed words are represented as nodes in a simple binary `Word Tree` structure, using Java’s red-black `TreeMap` class. The value for each node is an ordered list of occurrences of the stem, including positional information and term weights.

We chose a term weighting scheme that uses average term frequency in a document as the normalization factor. In the function

$$\frac{1+\log(tf)}{1+\log(average(tf))},$$

tf(t) is the actual frequency of term *t* in document *D*. This normalization method has been shown to perform 5.7% better than maximum term frequency based normalization for 200 TREC queries on the entire TREC collection [14]. Combining this *tf* factor with the pivoted normalization used in the SMART system, we arrive at the weighting strategy:

$$\frac{\dfrac{1+\log(tf)}{1+\log(average(tf))}}{(1-c)average(L)+c\times L},$$

where *L* is the number of unique terms in document *D* and *c* is a constant between 0 and 1. This weighting scheme is related to BM25 used in the Okapi system [13], which has been reported to perform consistently better than standard cosine normalization in document retrieval applications.

5. PASSAGE MATCHING

The Passage Matching module compares passages and assigns similarity scores to every pair of passages in the document set (not including pairs from the same document). The output is a table of paragraph pairs and their similarity values, represented as a matrix.

In order to compare large numbers of documents efficiently using *n*-gram matching, we chose to work with a very small subset of all possible substrings in the documents: we look only at *n*-grams, where *n* is 1 to 6, that are actually matched somewhere among the passages in the document set. We construct *n*-grams on the fly, in a bottom-up manner, from the Word Tree data structure. The algorithm is as follows:

```
For each document $D_i$ ($D_1$ to $D_{n-1}$):
a. Get the first word w in the first paragraph.
b. From the Word Tree, get a list a of all instances of w
   occurring in documents $D_{i+1}$ to $D_n$. These are 1-grams.
c. For each word v which follows w in $D_i$:
    1) From the Word Tree, get a list b of all instances of v in
       documents $D_{i+1}$ to $D_n$. Each v either continues an n-
       gram, wv, or begins a new 1-gram.
    2) Add or insert each v from list b in the proper place in
       list a. If an n-gram in list a cannot be extended by an
       occurrence in list b, or if it would create a sequence
       longer than n, then the n-gram is removed and stored in
       a matrix structure.
```

The goal is to find the best matches between any two paragraphs, where "best" is defined as maximum length. For example, the phrase *Federal Reserve Board Chairman Alan Greenspan*, when found in two passages, is counted as a 6-gram and not six 1-grams, or three bi-grams, or any other combination.

For computing the similarity between any two passages, we use a cosine coefficient function, modified according to *n*-gram weights. The *n*-gram weight is given as $(n_i)/n^2$, where n_i is the length of the *n*-gram of which term T_i is an element, and n is the length of the maximum *n*-gram. The weight of term T_i in passage X_i is the weight of T_i in document X plus the *n*-gram weight. The final passage similarity function is as follows:

$$sim(X_i, Y_i) = \frac{\sum_{j=1}^{t} x_{ij} \times y_{ij}}{\sqrt{\sum_{j=1}^{t} (x_{ij})^2 \times \sum_{j=1}^{t} (y_{ij})^2}},$$

where x_{ij} is the weight of term T_j in paragraph X_i and y_{ij} is the weight of term T_j in paragraph Y_i.

6. SEED-CLUSTERING

In order to form small, initial seed clusters we apply the well-known complete-link algorithm to our similarity matrix [17]. This algorithm becomes computationally expensive when used over large numbers of multi-paragraph documents. We have found it both practical and effective to run the complete-link only to the point at which we reach a target number of candidate seed clusters. We want a target that avoids over-generalization on the one hand and too much detail on the other. For most sets of documents, a good target is $\log_2 N$, where N is the number of documents. Initially, each passage is a cluster. We run the algorithm as follows:

1. Merge the most similar two clusters (clusters i and j).
2. Update the similarity matrix to reflect the pairwise similarity between the new cluster (ij) and the original clusters. We remove all the entries for i and j and replace them with new ij entries.
3. Repeat steps 1 and 2 until the target number of seed clusters is reached.

We add the restrictions that a seed cluster must contain three or more passages, and that there must be at least two common terms among the first three passages in the cluster. Clusters with larger common stem sets are preferred, as well as clusters whose common stem sets do not overlap much with another cluster.

7. PASSAGE CLASSIFICATION

In order to complete the clusters, all remaining passages are classified as satellites around the seed clusters. For this stage we perform M-bin classification, where M is the number of seeds. If a passage has no similarity to any of the seeds, it is placed into a miscellaneous “trash cluster”. Passages in a cluster are presented in descending order: seed passages come first, in the order in which they were added to the cluster, so that those with the tightest similarity to one another are shown first. Next come the satellite passages, ordered according to their degree of similarity with the seed cluster.

8. GENERATING 2 KINDS OF SUMMARY

After the clusters are formed, we create a “meta -document”, selecting one of the highest-scoring, or most characteristic, passages from each cluster, and concatenating them together. Next, our single-document summarizer creates a summary of this meta-document, using the query terms, if any, as the “title”. The user, then, has two types of summary to view. The meta-document is more suitable for groups of documents that describe similar but isolated events, such as alcohol-related traffic accidents, or different people who have something in common, such as winners of the Nobel Peace Prize. The summary of the meta-document is more appropriate for groups of documents which are all related to the same topic or event, such as oil exploration in the Falkland Islands, or the U.S. Presidential election in the year 2000.

9. DUC PARTICIPATION, EVALUATION

In August 2001 NIST completed the evaluation of single- and multiple-document summaries submitted by DUC participants. For the multiple-document summarization track, participants were required to submit fixed-length summaries of 50, 100, 200, and 400 words. Of the 25 groups who signed up, 12 submitted multi-document summaries, and 11 submitted single-document summaries. Ten information analysts from NIST compared human-written model summaries with peer summaries (system-created, baseline, or human). The analysts formed intrinsic judgments based on peer grammaticality, cohesion, and organization; coverage of each model unit by the peer (recall);

and other characteristics of peer material. In Figures 2 and 3, R represents the Albany/GE summarizers.

On grammaticality, our XDoX system ranked 6th of 12. Because we used only text extraction, no language generation, any grammatical errors must have come from the documents themselves. We chose to group subtitles and one-line text units, such as dates standing alone, with the following text, so these may have been interpreted as sentence fragments. On cohesion and organization our system ranked 3rd and 2nd. We attribute these high numbers to our method of extracting paragraphs as a whole, rather than individual sentences.

On per-unit content, assessors marked peer units—simple, automatically determined sentences—which expressed at least some of the same facts as the current model unit, or elementary discourse unit. When recall is defined as dividing the number of model units matched with peer units by the total number of units in the model summary, our single-document summarizer ranked 2nd of the 11 participants. Like our multi-document summarizer, the single-document system segments text into passages or equal-sized chunks. Adjacent passages that are strongly linked to one another are reconnected. All passages are scored with respect to a query derived from the title, topic description, and terms occurring frequently in the text. Scores are then normalized by the length of the passage. Passages are combined into groups of two or more and re-scored, until a clear winning passage or passage group emerges to form the summary. Our techniques have been shown to achieve an excellent balance between text compression and content preservation [16]. The DUC evaluation confirms this assessment.

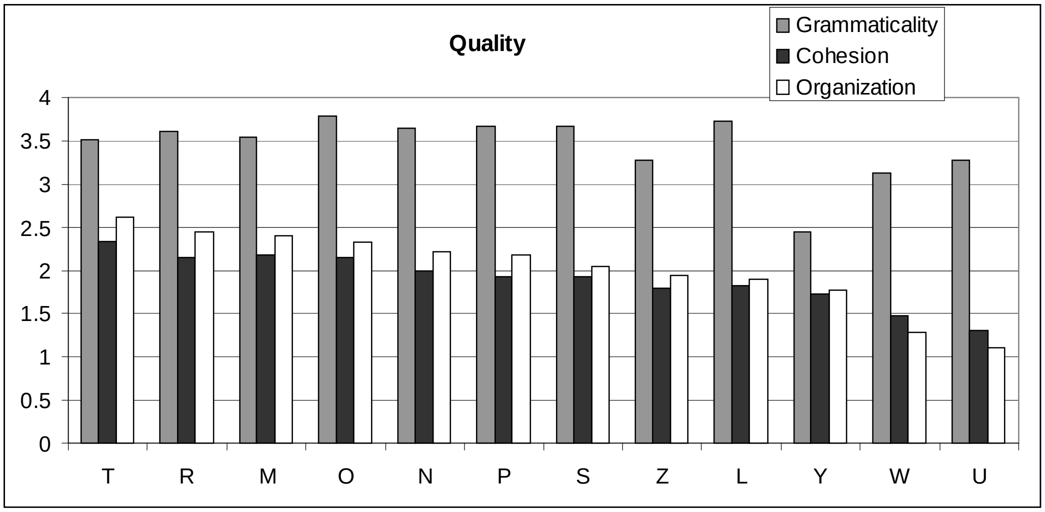

Figure 2. NIST intrinsic quality assessments for multi-document summarization systems.

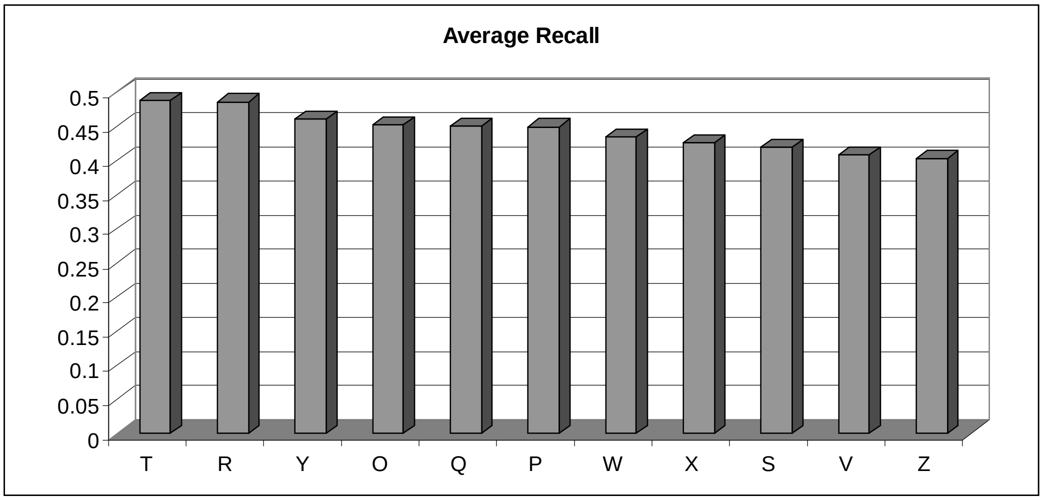

Figure 3. Average recall for single-document summarization systems.

On the whole, we are pleased with our performance in the 2001 Document Understanding Conference. We consider our summaries readable, coherent, and excellent at capturing the main points of the document sets. We think the DUC assessors did well given the program guidelines and the difficulties inherent in judging summaries. We look forward to extrinsic methods of evaluation being incorporated into future DUC conferences.

10. EXAMPLES

The following examples were generated by the XDoX system from the DUC 2001 data.

docset="d13c"

President Bush on Monday nominated Clarence Thomas, a conservative Republican with a controversial record on civil rights, to replace retiring Justice Thurgood Marshall on the Supreme Court.

CLARENCE THOMAS; Born: June 23, 1948, in Pinpoint, Ga. Education: B.A. from Holy Cross College, 1971; J.D. from Yale Law School, 1974.

docset="d39g"

Officials estimate the tunnel trains may carry 28 million passengers in the first year of operation. Eurotunnel doesn't expect a profit until the end of the century.

In London, the conservative Daily Express newspaper noted today that Britons will be able to walk to France for the first time since the Ice Age.

The tunnel's cost has soared from an initial estimate of $9.4 billion to $16.7 billion, including an extra $1.97 billion in case of unforeseen cost overruns.

Eurotunnel PLC announced Oct. 8 that it had reached an agreement with its banks on $3.5 billion in new credit. More than 200 banks are involved in financing the world's costliest tunnel.

The following summary was obtained from the set of top 100 documents retrieved from the TREC data collection with Topic 358 on the subject of alcohol related driving fatalities.

docset="TRECtopic358"

Drinkers beware. When the New Year begins at midnight, a tough new law will take effect making it all the more risky to drink and drive.

Under the new law that takes effect Jan 1, a driver with 0.08% or higher is presumed to be drunk; however, those with lower levels could also be cited for drunk driving.

According to court records, Giunta's blood-alcohol content was 0.29%, more than three times the level at which a motorist is considered legally drunk.

The Department clearly and specifically limited the NPRM to consideration of whether blood testing should be used for situations in which breath testing was not readily available for reasonable suspicion and post-accident tests, or in shy lung situations. For this reason, the issue raised by some commenters of whether employers should have the flexibility or discretion to use blood testing as an alternative to breath testing, even when breath testing is readily available in reasonable suspicion and post-accident testing or even in random or pre-employment testing, is outside the scope of the rulemaking.

The author, Sen. Bill Leonard (R-Big Bear), predicted that the measure will win legislative approval and be sent to Gov. George Deukmejian, whose Administration supports it.

11. CONCLUSION

Our latest version of the XDoX system, using our new passage-clustering techniques, implements a reasonable approximation of conceptual clustering. The overall quality is significantly better than that of our previous version, and compares favorably with the other DUC participants. The summaries are more readable and coherent. In most cases the system successfully presents main points, skips over minor details, and avoids redundancy. XDoX works best on large document sets that have multiple themes. For sets that have fewer than 20 documents, as in the DUC data, our system works well with some parameter adjustments, as long as the documents contain sufficient common concepts around which to form clusters. We have seen excellent results from document sets pertaining to a single event, topic, or person, such as gun control, the eruption of Mt. Pinatubo, the Charles Keating scandal, or the Rodney King beating. XDoX has difficulty generating a longer, more detailed summary when there are 10 or fewer documents concerning isolated incidents, such as gas explosions or earthquakes, or when there is little repetition. We are considering augmenting the system with selective use of our single-document summarizer [16].

12. ACKNOWLEDGMENTS

This paper is based on work supported in part by the Defense Advanced Research Projects Agency (DARPA) and by the Advanced Research and Development Activity (ARDA). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the U.S. Government.

13. REFERENCES

[1] Carbonell, J., and Goldstein, J. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In Proceedings of SIGIR (1998), 335-336.

[2] Fellbaum, C. (ed.). WordNet – An Electronic Lexical Database. MIT Press, 1998.

[3] Firmin, T., and Chrzanowski, M. J. An Evaluation of Automatic Text Summarization Systems. In I. Mani and M. Maybury (eds.), Advances in Automatic Text Summarization. MIT Press, 1999.

[4] Hatzivassiloglou, V., Klavans, J. L., Holcombe, M. L., Barzilay, R., Kan, M., and McKeown, K. R. SIMFINDER: A Flexible Clustering Tool for Summarization. In NAACL 2001 Workshop on Automatic Summarization (Pittsburgh, PA), 41-49.

[5] Hearst, M. Multi-paragraph segmentation of expository text. In Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics (Las Cruces,

NM, 1994), Association for Computational Linguistics, 9-16.

[6] Kraaij, W., Spitters, M., and van der Heijden, M. Combining a mixture language model and Naïve Bayes for multi-document summarization. In SIGIR 2001 Workshop on Text Summarization (New Orleans, LA), 95-103.

[7] Lin, C. and Hovy, E. NEATS: A Multidocument Summarizer. In SIGIR 2001 Workshop on Text Summarization (New Orleans, LA), 131-134.

[8] Marcu, D. Discourse-Based Summarization in DUC–2001. In SIGIR 2001 Workshop on Text Summarization (New Orleans, LA), 109-116.

[9] McKeown, K. and Radev, D. Generating summaries of multiple news articles. In Proceedings, 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (Seattle, WA, 1995), 74-82.

[10] Miller, G.A. WordNet: A Lexical Database. Communication of the ACM 38, 11(1995), 39-41.

[11] Mitra, M., Singhal, A., and Buckley, C. Automatic text summarization by paragraph extraction. In Proceedings of the ACL' 97/EACL' 97 Workshop on Intelligent Scalable Text Summarization (Madrid, Spain, 1997).

[12] Radev, D. R., Fan, W., and Zhang, Z. WebInEssence: A Personalized Web-Based Multi-Document Summarization and Recommendation System. In NAACL 2001 Workshop on Automatic Summarization (Pittsburgh, PA), 79-88.

[13] Robertson, S. E., Walker, S., Jones, S., Hancock-Beaulieu, M. M., and Gatford, M. Okapi at TREC-3. In Harman, D. (ed.), The Third Text Retrieval Conference (TREC-3). National Institute of Standards and Technology Special Publication 500-225, 1995, 219-230.

[14] Singhal, A., Buckley, C., and Mitra, M. Pivoted Document Length Normalization. SIGIR 1996, 21-29.

[15] Stein, G., Strzalkowski, T., and Wise, B. Interactive, Text-Based Summarization of Multiple Documents. Computational Intelligence 16, 4 (2000), 606-613.

[16] Strzalkowski, T., Stein, G., Wang, J., and Wise, B. A Robust, Practical Text Summarizer. In I. Mani and M. Maybury (eds.), Advances in Automatic Text Summarization. MIT Press, 1999, 137-154.

[17] Willett, P. Recent trends in hierarchical document clustering: A critical review. Information Processing and Management, 24, 5 (1988).