Domain-specific informative and indicative summarization for information retrieval

Min-Yen Kan
Computer Science Department
Columbia University
New York, New York 10027
USA
min@cs.columbia.edu

Kathleen R. McKeown
Computer Science Department
Columbia University
New York, New York 10027
USA
kathy@cs.columbia.edu

Judith L. Klavans
Center for Research on Information Access
Columbia University
New York, New York 10027
USA
klavans@cs.columbia.edu

ABSTRACT

In this paper, we propose the use of multidocument summarization as a post-processing step in document retrieval. We examine the use of the summary as a replacement to the standard ranked list. The form of the summary is novel because it has both informative and indicate elements, designed to help different users perform their tasks better. Our summary uses the documents' topical structure as a backbone for its own structure, as it was deemed the most useful document feature in our study of a corpus of summaries.

1. INTRODUCTION

Today's information retrieval (IR) systems typically give results of a search as a ranked list of documents. Unfortunately a ranked list of documents seldom directly fulfills the information needs of users. With the recent advances in automatic text summarization, it is now possible to use query-based multidocument summarization to close the gap between a user's information need and easily computable results.

In this paper, we employ both informative and indicative summarization in a novel way to satisfy this need. We argue that both types of summarization are needed to fulfill two basic classes of information need – browsing and searching – which we will detail later. In a nutshell, an informative summary can be in the form of a synopsis synthesized from the commonalities between the documents, fulfilling general information needs of a browser. An indicative summary can highlight the differences between documents, assisting a searcher in finding an appropriate document to retrieve.

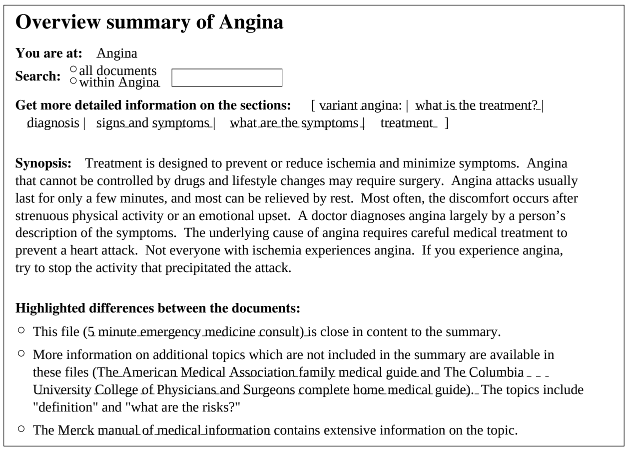
Alternative user interfaces to the traditional ranked lists are common in Human Computer Interaction (HCI) studies. Many of these interfaces have been graphical and specific to retrieval of multimedia documents (e.g., [9]). However, little has been done to apply natural language text processing to this problem. We utilize multidocument text summarization to replace the traditional IR ranked list interface. By performing a content analysis of existing summaries, we establish several summary design principles that can be used in an implementation. We take these principles and apply them in our prototype implementation of such a summarization system which works for mutiple documents within specific domain and genre combinations.

This paper is structured in two parts: theory and implementation. The first part examines how informative and indicative summarization can be used to fulfill different information needs; needs that are common in information retrieval tasks. Once we have established the need for both types of summarization, we detail our study of the content of existing summaries. This results in several design principles for producing summary content. The second part details Centrifuser, our operationalization of these design principles for such a query-based multidocument summarization system. Centrifuser is a prototype system that works for specific documents that are of the same domain and genre. It generates summaries like the one show in Figure 1. Structured around the distribution of topics, we show how the informative and indicative modules of Centrifuser fulfill their assigned objectives.

2. INFORMATIVE AND INDICATIVE SUMMARIZATION

Summaries are written to serve specific purposes and thus the summarization task itself is varied. Dimensions of summarization include (but are not limited to) a) indicative or informative summaries, b) number of documents to be summarized, c) generic or query-based, as well as others. In the case of the IR ranklist, the value of two of these dimensions have already determined: the substituting summaries must be multidocument and they must be query-based. One axis remains. Should we choose to structure such a summary as an indicative or informative one? We first define these terms and some give examples.

An informative summary is meant to represent (and often replace) the original document. Therefore it must contain all the pertinent information necessary to convey the core



Overview summary of Angina

You are at: Angina

Search: ○ all documents ○ within Angina

Get more detailed information on the sections: [variant angina: | what is the treatment? | diagnosis | signs and symptoms | what are the symptoms | treatment]

Synopsis: Treatment is designed to prevent or reduce ischemia and minimize symptoms. Angina that cannot be controlled by drugs and lifestyle changes may require surgery. Angina attacks usually last for only a few minutes, and most can be relieved by rest. Most often, the discomfort occurs after strenuous physical activity or an emotional upset. A doctor diagnoses angina largely by a person's description of the symptoms. The underlying cause of angina requires careful medical treatment to prevent a heart attack. Not everyone with ischemia experiences angina. If you experience angina, try to stop the activity that precipitated the attack.

Highlighted differences between the documents:

- This file (5 minute emergency medicine consult) is close in content to the summary.
- More information on additional topics which are not included in the summary are available in these files (The American Medical Association family medical guide and The Columbia University College of Physicians and Surgeons complete home medical guide). The topics include "definition" and "what are the risks?"
- The Merck manual of medical information contains extensive information on the topic.

Figure 1: A CENTRIFUSER summary composed of indicative and informative halves

information and omit ancillary information.

An indicative summary's main purpose is to suggest the contents of the article without giving away detail on the article content. It can serve to entice the user into retrieving the full form. Book jackets, card catalog entries and movie trailers are examples of indicative summaries.

In our IR context, the definition of both informative and indicative summaries needs to be extended. A multidocument, query-specific informative summary captures the most important aspects across documents, in which importance can be equated with relatedness to the user's query. As the documents have the query in common, summarizing the documents' overlapping areas (their similarities) implements an informative summary. A multidocument, query-specific indicative summary suggests the content of the documents and helps user to distinguish an appropriate document for full-text retrieval.

Which is more appropriate? Our hypothesis is that both types are useful for IR. Since both types of summaries have different forms, there are distinct scenarios in which they are useful. Informative summaries are good at satisfying broad information needs because they capture the salient commonalities between the documents. A user who is browsing for information, who just wants to learn about a topic in general, might be satisfied with our multidocument informative summary. On the other hand, indicative summaries are good at routing: matching a specific information need to particular (subset of) documents. A user who is searching for an answer to a particular question often will not find the information they need in a multidocument synopsis. These searchers are best served with an indicative summary that highlights the documents' differences and route them to a particular document.

Although our naïve user model only accounts for two tasks – browsing and searching – we believe that informative and indicative summaries differ in their power to aid each task. Multidocument informative summaries capture broad similarities which are good for browsing, and multidocument indicative summaries capture salient differences which are good for searching.

3. SUMMARY CONTENT ANALYSIS

Identifying similarities and differences between documents is a well known strategy in multidocument summarization. However, documents can be similar or different with respect to many dimensions, such as in topic, in format, in their intended audience. To help decide which dimension is most important to include, we conducted a study of existing indicative summaries from our local library's online catalog to see which types of document features are included in actual' summaries.

We extracted a corpus of single document summaries of consumer health publications, containing a total of 82 summaries, averaging a short 2.4 sentences per summary. We manually identified several types of document features used in the summaries and characterized their percentage appearance, presented in Table 1.

Document Feature	% appearance in corpus
Document-derived features	
Topicality (e.g. "Topics include symptoms, ...")	100%
Content Types (e.g. "figures and tables")	37%
Internal Structure (e.g. "is organized into three parts")	17%
Readability (e.g. "in plain English")	18%
Special Content (e.g. "Offers 12 credit hours")	7%
Conclusions	3%
Metadata features	
Title	32%
Revised/Edition	28%
Author/Editor	21%
Purpose	18%
Audience	17%
Background/Lead	11%
Source (e.g. "based on a report")	8%
Media Type (e.g. "Spans 2 CDROMs")	5%

Table 1: Distribution of document features in library catalog summaries of consumer healthcare publications.

This study reports results for a specific domain, but we feel that some general conclusions can be drawn. Not surprisingly, information about the topical contents of the document were contained in all summaries. This supports a core design principle of performing summarization based on the articles' topics.

Perhaps more surprising is how often other document features were included in the summary. These document features were often included when the features indicated a value out of the ordinary – that there was something special about the document. For example, most documents addressed lay consumers, but in specific documents that targeted medical students, the audience feature often was explicitly mentioned. A design principle that results from this is to report

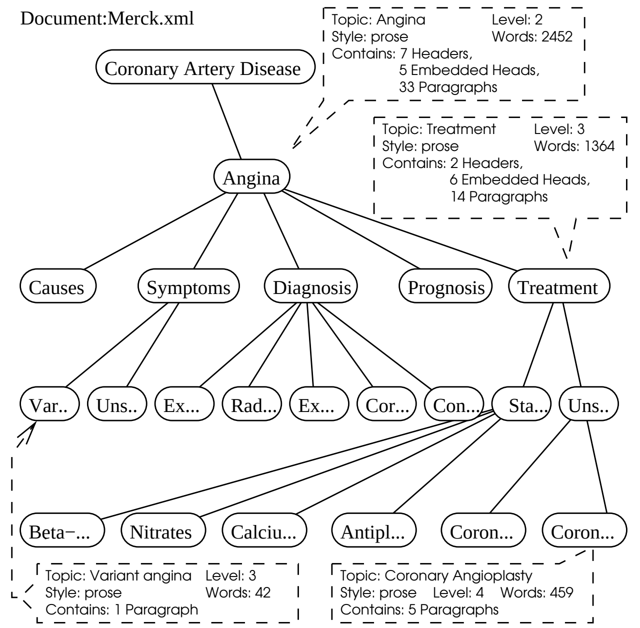


Figure 2: A topic tree for an article about coronary artery disease from *The Merck manual of medical information***, constructed from its section headers**

these optional pieces of information when the document feature differs from the norm.

4. INTRODUCTION TO CENTRIFUSER

We created Centrifuser, a summarization system to meet the needs of browsers and searchers in highly structured domains. In a nutshell, Centrifuser relies on an infrastructure of document topics. First, documents are represented by topic trees and tree structure-based similarity calculations are used to find which topics are similar and which are different. The browser summarization module takes the similar topics and performs sentence extraction of salient sentences for the synopsis. It additionally utilizes relationships between the topics to generate navigation links to related topics. The searcher module uses text generation techniques to create text that describes high level differences between documents. In the remainder of the paper we detail each of these steps.

4.1 Document topic trees

A topic is important to us as a granularity level; it is smaller than a document and larger than individual sentences or paragraphs. A suitable representation for documents for our task is one where documents contain a number of topics, structured into a tree-like hierarchy. Each document is then represented by a *document topic tree*, which breaks each document's topic into its subtopics. Calculations of similarity and difference are then done at the topic level, employing textual features as well as their structural relationship.

4.2 Composite topic trees

For cross document comparison, topics need to be compared to others and judged as similar or different. The concept of the *composite topic tree* allows this comparison to happen.

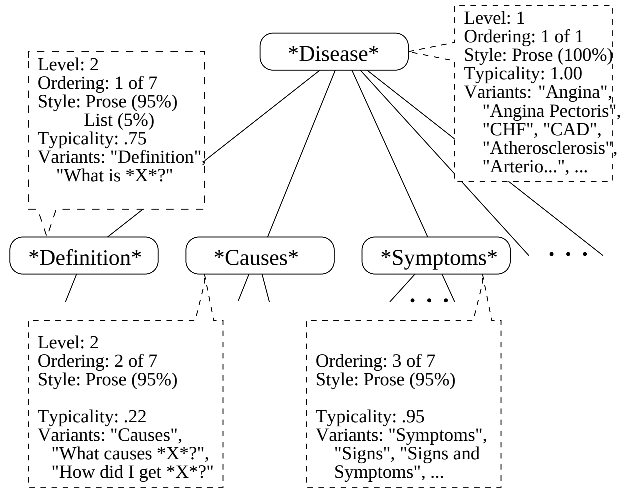


Figure 3: A sample composite topic tree for the patient information on diseases text type

The topic tree describes what topics are contained in the average document of a particular text type (*text type* meaning documents of the same domain and genre). The composite topic tree is termed *composite* as all documents of the same text type are instances of it. In the composite topic tree, each topic is given a typicality score, its ordering among its siblings, and variant forms that denote alternative ways to express the topic. For instance, in the partial composite topic tree in the patient information text type in Figure 3, the topic "Symptoms" is very typical (.95 out of 1), usually comes before the other sibling topics "Definition" and "Cause" and may be expressed as the variant "Signs".

While the computation of document and composite topic trees is not a focus of this paper, Centrifuser does automatically construct them for our current text type focus of patient information on diseases. In this text type, documents are highly structured, such that we can use the section and subsection headers to serve as the document topic tree. In other less structured text types, document topic trees can be generated by natural language approaches such as hierarchical text segmentation [10], or by doing full discourse parsing [2].

To build the composite topic tree, Centrifuser merges together instances of sample document topic trees of the same type. By analyzing the trends of which topics occur more frequently in the documents of the same type, we can get a good estimate for each topic's typicality, and also find alignments between topics that have similar content to deduce variant lexical forms of topics. We followed this approach in previous work to automatically generating this *composite topic tree* resource from training documents of the same text type [4].

4.3 Query mapping

Given document and composite topic trees, Centrifuser maps the query to the single most similar topic in both types of trees. This can be done using a function as simple as word overlap; Centrifuser enriches this similarity function with

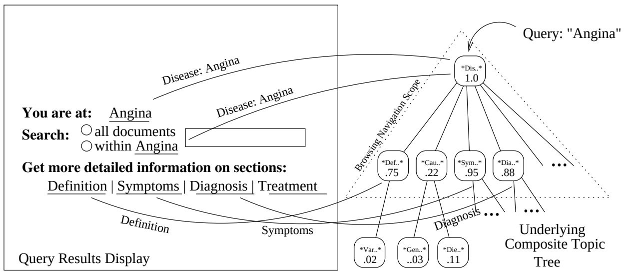


Figure 4: Navigation control construction from the composite topic tree. Since space is limited, only nodes with high typicality (as indicated by the number on the composite tree node) are placed on the navigation controls.

structural information from the topic trees [4]. Currently, queries are mapped to the best single topic node, and as such, complex queries that may best represented as mapping to several nodes are currently not handled well. Future work on extending the framework is needed to handle this problem.

Equipped with these three pieces of information – document topic trees for each of the documents in the result set, a composite topic tree for the text type, and the query mapped to topic nodes in the document and composite trees – CENTRIFUSER produces summaries suited for browsing and searching. Let's examine how the system handles each of these two tasks in turn.

5. SUPPORTING BROWSING WITH NAVIGATION LINKS AND EXTRACTED SIMILARITIES

From the user analysis, browsing support consists of giving a general synopsis of commonalities between the documents, enabling the user to freely wander around related topics. We discuss the navigation of related topics first.

5.1 Navigational links

The navigation links are built directly from the composite topic tree. Once the query is mapped to a topic node in the composite tree, it defines a browsing scope, a region of topics which are instantiated as browsing targets. Figure 4 illustrates this browsing scope with the dotted outline. In CENTRIFUSER, the browsing scope is all the immediate children of the query node (e.g. "Causes", "Symptoms", "Treatment", etc), plus all of its direct ancestors to the root of the composite topic tree (e.g. none in this scenario). Each of these browsing scope topics is mapped to text representing the topic and used as hyperlinks in our display. When space constraints force us to limit the number of topics we can show, we use the top n most typical topics (thus "Causes" would not be shown, having a lower typicality of .22). Activating one of these links would cause a new query equivalent to the topic to be posed to the entire document collection.

Although not implemented in the current version of the system, we plan to add restricted search. An option to apply the search string in the text box only to the current topic (and subtopics) could be added. The subsequent search would be restricted to map the query only within the current topic's subtopic tree.

5.2 Synopsis based on similarities

The browser's synopsis is created from the text of the topics of the individual documents, the length in sentences determined by a user-controlled parameter. Similar to the browsing scope used in the navigational links, the query node is used here to establish a scope of relevant topics in the composite topic tree. Descendant topics within a set depth k away from the query node form this scope of relevent topics. "Relevant" means relevant to the query; non-relevant topics are either too *intricate* in detail for use in the synopsis (over k deep descendants from the query node) or just *irrelevant* (outside the subtree defined by the query node).

To build the synopsis, appropriate text must be chosen to represent each relevant topic. In our implementation, we use the method of sentence extraction [8], which is well-accepted since it is simple, fast and easy to evaluate. In this technique, sentences from the original documents are selected and put together to form a summary.

The composite topics do not have actual texts to represent the topic; the text resides in the individual documents. To retrieve appropriate text, we need to map the topics in the composite topic tree to the individual document trees, in the same way that the query was mapped to a composite topic node for the navigational controls. Figure 5 shows how the composite topic tree might be mapped to a specific document topic tree. In the figure, we can see that the composite topic "Definition" is not mapped to any topic in the specific document topic tree. Occasionally, composite tree topics will not map to any topic in the document topic trees. These topics cannot be included in the synopsis since there is no text to represent them.

We can now restate the task of creating the browser's extract as three steps:

1. dividing the summary's allotment of sentences among the topics that are relevant, and instantiated by physical text;
2. selecting the sentences in the physical text;
3. ordering the selected sentences into an extract.

A synopsis is often too short to encompass all the possible relevant instantiated topics. To divide the summary's allotment of sentences fairly we rank the topics using their typicality rating. Unlike from relevancy, a topic's typicality is not a factor of the query. It is a property associated with each topic at the document collection level in the composite topic tree. The typicality scores allow us to prioritize which topics should be allotted space first in the summary when space is limited. In the composite topic tree in Figure 5, the topic "Causes" has lower typicality than "Symptoms" and "Treatment", and thus is more likely to be omitted over the other topics if space is limited.

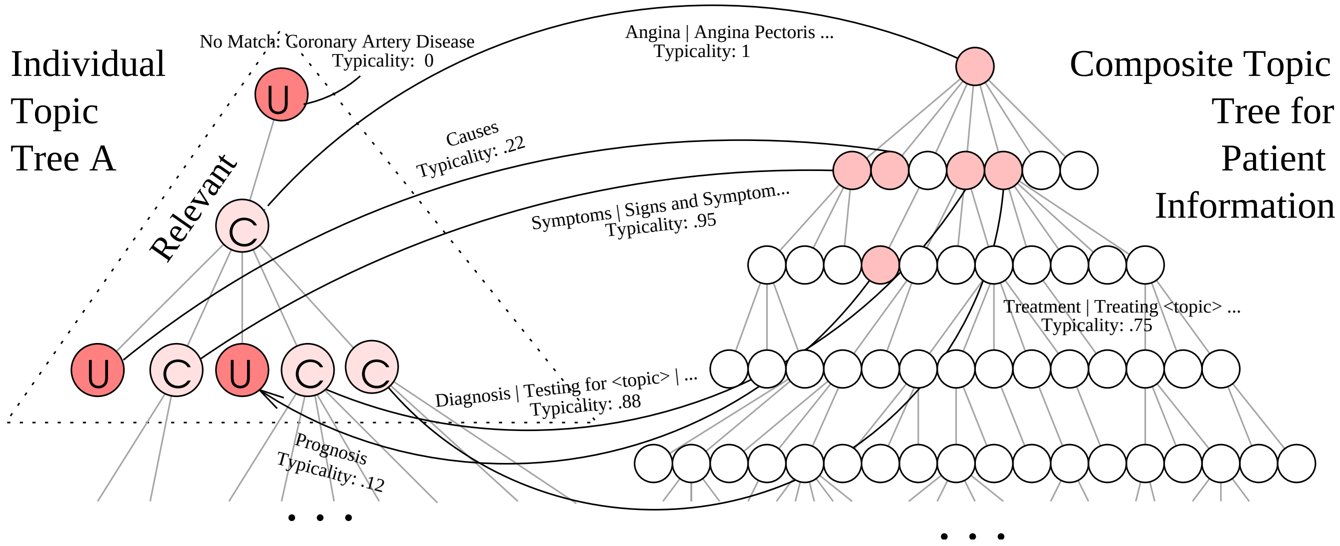


Figure 5: Aligning topics from the composite tree an individual topic tree. Typicality scores are then propagated across

To assign an allotment of sentences to each topic, Centrifuser follows a "card dealing" algorithm, with a sentence being analogous to a card. Each topic is dealt a sentence, in order of descending typicality, until the sentence quota for the synopsis is exhausted. This approach ensures that the highest possible breadth in topics is covered within the sentence quota, and that the quota is used on the most typical first. For example in Figure 5, six topics were aligned. Given a seven sentence synopsis, "Symptoms" would receive two sentences whereas "Variants" would receive only one.

Once a topic receives a sentence allotment, we must choose the sentences to represent it. To perform this task, we utilize a sentence clustering technique [7, 3] that takes as input a set of sentences and organizes them into clusters based on their sentential similarity. For each topic, we run the clustering program on the sentences of the topic instances, producing clusters of similar sentences as output. We now chose a single sentence to represent each cluster. As similar sentences may come from different documents and may contain redundant information, it is important to cluster them and chose a single representative sentence for the cluster to eliminate redundancy, similar to the goal of Maximal Marginal Relevance [1].

We re-rank the clusters according to the number of different individual documents represented and resolve ties by size (number of sentences found to be similar). For example, if the texts for "Symptoms" result in two sentence clusters, one with sentences from three different documents, and the other with sentences only from a single document, we would select sentences from the cluster that represents information found in the three document cluster first.

The cluster's representative sentence is chosen based on rules. The system prefers sentences from paragraph text, over list items or bullet points, over section headings. If sentences are of the same type, the sentence that occurs earlier in its instance text is chosen. If the sentences occur in the same location, the longer one (in words) is chosen.

The final subtask is to order the selected sentences into a summary. We can do this by first ordering the selected topics and then internally ordering each topic's sentences. Topics are organized by their typical ordering found in the composite topic tree (e.g. "Symptoms" before "Diagnosis" before "Treatment"). Within a topic, sentences are ordered by their physical position. Sentences that come earlier in its instance text are positioned first.

The result is an ordered extract, which we use as the browser's synopsis. By choosing breadth over depth in our use of the card dealing algorithm to allot sentences, we fulfill our goal to create an overview. By clustering similar sentences and using only a single sentence per cluster, we attempt to eliminate redundant information. It is an informative summary, since by embedding relevant sentences, it represents a typical document.

6. SUPPORTING SEARCHING WITH GENERATED DIFFERENCES

The text for the searcher's differences is created primarily using topic information from the document topic trees. The distribution of the topics within each document allows us to categorize each in a meaningful way for the searcher to pinpoint which document may be useful to retrieve.

Some documents will be more relevant for specific searches and less relevant for others. For example, a document that specializes in treatments will be useful for the patient looking for the side effects of certain drugs, but may be useless for another person who is unsure of whether she has angina, and is interested in ways to diagnose the disease.

This type of query interaction can be modeled by examining the query nodes in relation to the individual document topic trees. In each document, the query node defines three

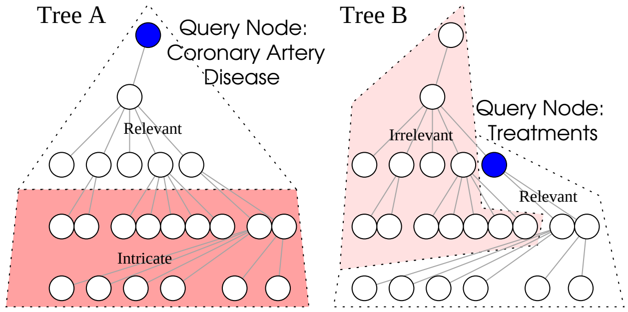


Figure 6: A pictorial representation of how *relevant*, *irrelevant*, *intricate* topic types are defined by the interaction of the topic tree and the query node, for k=2

regions, shown in Figure 6: nodes that are relevant to the query, ones that are too *intricate*, and ones that are *irrelevant* to the query, just as in the relevant topic determination used in constructing the browser's synopsis. Each individual document's ratio of topics in these three regions can help us assess the document's importance. In our earlier example, a document mostly on treatments would consist mostly of relevant topics to the treatment query but would consist mostly of irrelevant topics to the diagnosis query.

Some documents will also be more interesting than others according to the type of relevant information they possess. For example, if "Prognosis" is a rare topic to find in an angina document, it may be worthwhile to report to the searcher in case they are looking specifically for this hard to find information.

In our implementation, this is done using the typicality values from the composite topic tree. The relevant topics in each document topic tree are each mapped to a composite node, if possible. The typicality score is inherited from the composite topic, or if no mapping was possible, it is considered unique (0 typicality). For convenience, we set a threshold α, above which we consider a relevant topic *typical* and below which we consider it *rare*. Thus, a document that has many rare topics, such as "Prognosis", can be reported to the searcher as criteria for retrieval.

The distribution of these four topic categories – *rare*, *typical*, *irrelevant* and *intricate* – classify the document into a distinct document category. In developing the categories, we assumed that the most important topics are those that are most related to the query topic. Thus, our document categories consider the topic distribution in descending order of relevance to the query: first rare and typical topics, then intricate ones, and finally irrelevant topics. We explain the categories below and give specific details on the threshold set to detect them in Table 2. The examples in the list below pertain to a general query of "Angina".

1. **Prototypical** - This kind of document has a topic distribution that matches the distribution of topics in the composite topic tree. This is interpreted as two symmetric relationships. 1) Most of the typical topics in the composite topic tree are present as topics in the document. 2) Its relevant topics are mostly ones that are listed as typical in composite tree. An example would be an average document about angina – *American Medical Association's Guide to Angina*.
2. **Comprehensive** - If only the first requirement of the prototypical document type is met, then we have a document that has typical content but also contains other topics. The document thus covers more topics than usual, and is usually longer than other documents. An example of a comprehensive document could be a chapter of a medical text on angina.
3. **Specialized** - On the other hand, if only the second requirement of the prototypical document type is met (that its relevant topics are mostly typical), we know that the document treats only a portion of the normal amount of typical topics relevant to the query. These documents specialize in its typical topics. A specialized example might be a drug therapy guide for angina.
4. **Atypical** - An atypical document (characterized by many a high rare-to-typical topic ratio) contains information that may relate the document's text type to other text types (interdisciplinary), or may contains information on special cases. If the topic "Prognosis" is rare, then a document about life expectancy of angina patients would be an example.
5. **Deep** - These documents are often barely connected with the query topic but have much underlying information about a particular subpart of the query. An example of this type is a whole document on "Treatments of Angina" when "Angina" is the query node and "Treatments" registers as a kth level topic (k again is the beam depth from the query node for which topics under k levels away are consider relevant).
6. **Irrelevant** - An irrelevant document contains a high irrelevant-to-relevant (= rare + typical) ratio of topics. The text contains information about the subject in question, but not in the particular area of interest. A document about all cardiovascular diseases that mentions angina briefly may be considered irrelevant.
7. **Generic** - These documents do not display tendencies towards a particular distribution of information.

Since the criteria for these categories are not mutually exclusive, we apply the first applicable category to the topic. For example, if a particular document is comprised of 50% rare topics and 50% irrelevant topics, we would report it as a specialized topic.

Once each document has been categorized, we generate a short description of each document type category that contains at least one document. As with the extract portion of the summary, the length of this description text is controlled by the user. The generated descriptions vary in content according to the number of documents placed in the category. As the number of categories are limited (there are only seven) and since only a single description is generated per category (regardless of the number of documents belonging to it), it is possible to compress a query result set of hundreds of documents onto a single screen.

Document Type	Topic Distribution	Description
Prototypical	*typical* >= 50% *typical* >= 50% of in-scope prototype	The typical document, which is well represented by the extract
Comprehesive	*typical* >= 50% of in-scope prototype	Contains more than just the typical topics
Specialized	*typical* >= 50%	Contains some of the typical topics
Atypical	*rare* >= 50%	Contains rare information
Deep	*intricate* >= 50%	Contains content that is too detailed for this query
Irrelevant	*irrelevant* >= 50%	Contains mostly information outside query focus
Generic	*n/a*	Contains a mix of topic types, no strong trends

Table 2: Conditions used to categorize documents into document types

To decide exactly what information to generate in the textual description, we conducted a study of indicative card-catalog summaries from the Library of Congress, described in further detail in [5]. The main result was that topical information was most important, leading us to design the description with obligatory topical information but having optional information about other document features (e.g. content type "does the document contain pictures or tables" or audience "does it target medical students") if space allows.

The obligatory topical information describes the document category and lists the documents belonging to it (or at least a subset or exemplar if the number of documents belonging to the category is large). For document categories that are defined by their ratio of typical to rare topics, we have a lexical choice of referring to the browsers' synopsis, as it is constructed from the typical topics, as in the first description of the *prototypical* category in Figure 1 – "is close in content to the extract". The details of the generation process can also be found in [5].

7. EVALUATION

Currently, the implemented system has been utilized to produce summaries for patient health documents for several diseases. By using the automated methods to construct both composite and document topic trees, the current system can be used to generate summaries of documents of any text type.

We plan to formally evaluate the system in the near future, using a task-based approach. The *in situ* aspect is crucial because the hypothesis of the system is that such query-based multidocument summarization system can be more effective that traditional IR approaches for specific browsing and searching problems. During the evaluation, we will ask subjects to think aloud, which will allow us to gather exploratory data about what other types of information users may find helpful in completing a task.

7.1 Complexity

Another important aspect of the system is time efficiency of the summarization process. Fortunately, a large part of the necessary processing (e.g. computation of composite and individual topic trees) can be done offline and stored for later use. Aside from the computationally expensive sentence clustering step in the browser's synopsis, the time complexity of this unoptimized system is $O(mn)$, where m is the number of topics and n is the number of documents, and takes a few seconds to produce a five document summary on a PC workstation.

8. FUTURE WORK

In future work, we plan to augment the browser's extract by utilizing information in the user's session history. An example of this is referencing information that the user has seen before (e.g. "The treatment for unstable angina is similar to those for stable angina, which you've read about previously"). Definitions from other sources such as dictionaries and acronym lists that might be found automatically [6] can also be added in when appropriate (e.g., the bolded text in "Treatment options include CABG **(Coronary Artery Bypass Graft – surgery done to relieve blockages of the blood vessels of the heart muscle)**, ...").

The classification we use for differences is also a target for improvement. Currently the thresholds we use for classifying documents into categories are static. Sometimes the topic distribution within the document set is not so prominent (e.g. most documents get classified as *generic*). In these cases it would be good to dynamically lower the thresholds to reclassify the documents into a wider variety of categories. A backoff method for thresholding could ensure that at least two or three categories get instantiated.

9. CONCLUSION

In this paper, we have posited that query-based multidocument summarization can be used as a means of presenting query results of search from a retrieval system. Our user analysis provides a clear statement of how summarization systems can help retrieval system better match the commonalities and navigational needs of the browser and differentiation needs of the searcher.

We introduced CENTRIFUSER, our summarization system that aims to fulfill the goals of the needs analysis in specific domain and genre combinations. We hypothesize that the system is an improvement over the ranked lists because the rationale for each element in the display is derived from user needs. The user is presented with both a multi-document informative synopsis of the relevant documents as well as indicative qualities that differentiate them. The sentence extraction based synopsis provides users with commonalities between the documents. Its aim is to provide a surrogate for retrieving an actual document for broad information needs on salient subtopics of the query subject. Topical indicative differences between the documents are also generated to differentiate the documents in terms of topical content as well as in terms of their meta document features. The differences aim to assist the user in choosing the appropriate text document. Finally, both search and browsing navigational controls are fully integrated with the system to allow for flexibility in posing follow-up queries.

The system relies on algorithms for decomposing documents into topics as well as algorithms for collecting collection-wide information that form the composite topic tree. These technologies are currently available and we believe will continue to improve in performance.

While our implemented summarization system has yet to be formally evaluated, we believe that our topic-based and user-motivated query display portends that automatic text summarization has a prominent role in the future of information retrieval systems.

10. REFERENCES

[1] J. Goldstein. Automatic text summarization of multiple documents. Thesis Proposal, Carnegie Mellon University, 1999.

[2] U. Hahn. Topic parsing: Accounting for text macro structures in full-text analysis. *Information Processing & Management*, 26(1):135–170, 1990.

[3] V. Hatzivassiliglou, J. L. Klavans, M. L. Holcombe, R. Barzilay, M.-Y. Kan, and K. R. McKeown. Simfinder: A flexible clustering tool for summarization. In *Proc. of the Workshop on Automatic Summarization, NAACL 2001*, 2001.

[4] M.-Y. Kan, J. L. Klavans, and K. R. McKeown. Synthesizing composite topic structure trees for multiple domain specific documents. Technical Report CUCS-003-01, Columbia University, 2001.

[5] M.-Y. Kan, K. R. McKeown, and J. L. Klavans. Applying natural language generation to indicative summarization. In *Proc. of 8th European Workshop on Natural Language Generation*, Toulouse, France, 2001.

[6] J. L. Klavans and S. Muresan. Definder: Rule-based methods for the extraction of medical terminology and their associated definitions from on-line text. In *Proceedings of AMIA Symposium 2000*, page 1096, 2000.

[7] K. R. McKeown, J. Klavans, V. Hatzivassiloglou, R. Barzilay, and E. Eskin. Towards multidocument summarization by reformulation: Progress and prospects. In *"Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI-99)"*. ACL, July 1999.

[8] C. D. Paice. Constructing literature abstracts by computer: techniques and prospects. *Information Processing and Management*, 26(1):171–186, 1990.

[9] P. Pirolli, P. Schank, M. Hearst, and C. Diehl. Scatter/gather browsing communicates the topic structure of a very large text collection. In *Conference on Human Factors in Computing Systems, CHI-96*, pages 213–220, 1996.

[10] Y. Yaari. *The Texplorer*. PhD thesis, Bar Ilan University, Israel, April 1999.