Customization in a Unified Framework for Summarizing Medical Literature

N. Elhadad [a], M.-Y. Kan [b], J.L. Klavans [c], and K.R. McKeown [a]

[a] *Department of Computer Science, Columbia University, New York, NY 10027, USA*

[b] *School of Computing, National University of Singapore, Singapore, 117543* [1]

[c] *Center for Research on Information Access, Columbia University, New York, NY 10027, USA*

Correspondence to: Noemie Elhadad, (phone) +1 212 939 7113, (fax) +1 212 666 0140

Abstract

Objectives: We present the summarization system in the PERSIVAL medical digital library. Although we discuss the context of our summarization research within the PERSIVAL platform, the primary focus of this article is on strategies to define and generate customized summaries.
Methods and Material: Our summarizer employs a unified user model to create a tailored summary of relevant documents for either a physician or lay person. The approach takes advantage of regularities in medical literature text structure and content to fulfill identified user needs.
Results: The resulting summaries combine both machine-generated text and extracted text that comes from multiple input documents. Customization includes both group-based modeling for two classes of users, physician and lay person, and individually driven models based on a patient record.
Conclusions: Our research shows that customization is feasible in a medical digital library.

Key words: medical digital library, user modeling, multi-document summarization, multi-document information extraction, clinical information system.

1 Introduction

In the healthcare domain, healthcare providers and consumers alike increasingly turn online to find information of interest. In the medical community, the number of journals relevant to even a single specialty is unmanageably large, making it difficult for physicians to keep abreast of all new results reported in their fields. [2] Similarly, patients and family members

Email addresses: noemie@cs.columbia.edu (N. Elhadad), kanmy@comp.nus.edu.sg (M.-Y. Kan), klavans@cs.columbia.edu (J.L. Klavans), kathy@cs.columbia.edu (K.R. McKeown).

[1] The work presented here was done previously, while the author was at Columbia University.

[2] For example, there are five journals which publish papers in the narrow specialty of cardiac anesthesiology but 35 different anesthesia journals in general; approximately 100 journals in the closely related fields of cardiology (60) and cardiothoracic surgery (40); and over 1,000 journals in the more general field of internal medicine.

who need consumer information about a specific illness presented in lay terms can also be overwhelmed with the choice of online information related to their interest. While search engines have improved in accuracy and provide advanced strategies for searching, or presenting the results of a search, little effort has gone into using summarization to help the user navigate through the search results or browse more efficiently.

In this paper, we discuss a system to summarize documents as part of a medical digital library being developed at Columbia University, called PERSIVAL (PErsonalized Retrieval and Summarization of Images, Video and Language) [1], which aims to provide tailored presentation of the relevant medical literature for both physicians and lay consumers. The PERSIVAL summarization component takes as input documents relevant to a user's query, retrieved by a search component. It generates a one- or more-paragraph English summary of the group of documents, highlighting facts that are common to all documents and pointing out differences among them. PERSIVAL uses the summary to provide a gist of key facts of interest in the documents, highlight topics covered and provide links to the documents, and, in some cases, to topics related to the query.

A key feature of the PERSIVAL summarizer is the ability to personalize its content; the result is a summary that highlights information that is more likely to be relevant to the user. Personalization is influenced by a user model in three ways:

- Whether the user is a medical health provider or consumer influences both the types of documents summarized and the type of summary provided.
- Whether the user wants a focused, precise summary in response to a specific search or is just browsing for information in a more exploratory mode influences the type of summary produced.
- Information about the patient's medical history helps to determine summary content. For a physician, information must be relevant to the patient under care.

To best address the information needs of the different user groups, we designed and implemented two summarization strategies in PERSIVAL. One synthesizes the results of clinical studies pertaining to a specific patient for physicians as end users, while the other generates summaries of general health information for lay consumers. Both strategies use text mining, extraction and generation to produce the summaries.

In the next sections, we first give an overview of the PERSIVAL project. Then we show how our user model represents both physicians and lay users as two ends of a continuum, and how this motivates the development of two summarization strategies within a unified framework. We describe each strategy in turn and conclude with current work.

2 PERSIVAL: Personalized Access to Medical Literature

PERSIVAL is designed to provide personalized access to a distributed digital library of multimedia medical literature. It is an interdisciplinary project that involves researchers in computer science, electrical engineering, medical informatics and library and information science. The PERSIVAL architecture in Figure 1 illustrates the many components in the system, each of which is described in references cited below.

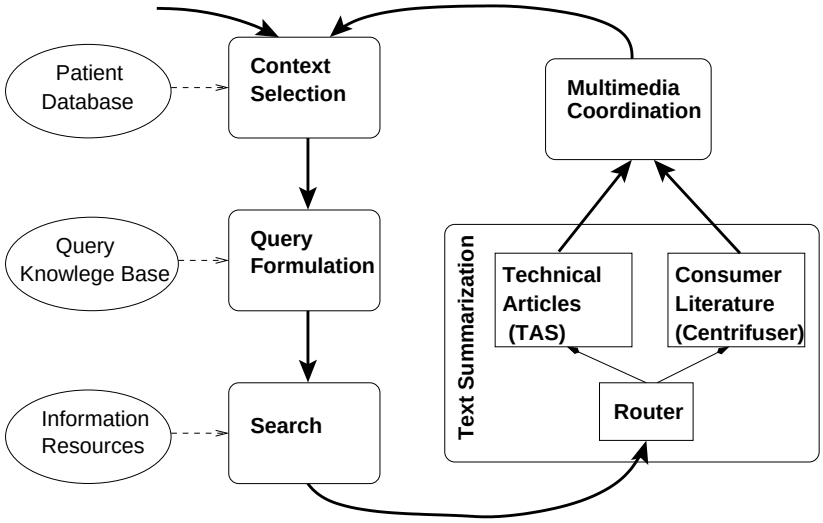

Fig. 1. PERSIVAL system architecture with focus on the text summarization modules

A key feature of PERSIVAL is the ability to present information relevant to the user's query given the context of patient information. PERSIVAL links to the large online patient record database available at the New York Presbyterian Hospital, which serves as part of the user model [2,3]. The interaction with PERSIVAL begins with access to a specific patient record. After viewing the patient record, the user may decide to access the online medical literature and pose a question in natural language. The Query Formulation module helps the user to formulate a good question related to the patient information within the context and translates the natural language question into a query [4]. The query is then sent to the search engine, which allows access to distributed online textual resources [5] as well as a library of digital echocardiograms. The results of the text search are re-ranked by matching the articles returned against the patient record, scoring those articles which discuss results related to the patient's case as more relevant [6]. A text summarizer and a video summarizer [7] each generate a summary of the relevant results, and a multimedia coordination component produces explicit links between the two. The resulting multimedia summary and search results are presented to the user as part of a novel user interface, that uses a sophisticated layout component to dynamically determine where to place information on the screen [8].

The focus in this paper is on the text summarizer. The other components of PERSIVAL are being developed by other researchers on the project. The summarizer receives as input the user query as finalized by the query interface, a set of relevant documents returned by search, and a pointer to the patient record. It uses the values of the user model to decide which summarization strategy to apply. We have implemented two strategies: one for physicians which provides the user with a synthesis of clinical studies that are relevant to the patient under care, and one for lay users that generates summaries of general health information. The summarization strategies follow the same conventions to encode both the input data (user model and articles) and output data (text summary and meta-data).

From a functional perspective, both strategies act as a way to filter out irrelevant information further and to synthesize new summary text. They follow the same high-level architecture, consisting of three phases: 1) salient information is first selected from the input articles

(content selection phase), 2) the extracted information is ordered (content organization phase), and 3) a text is regenerated by weaving together extracted phrases from the input articles (text regeneration phase). Both strategies are appropriate for a wide range of medical texts and exploit the structure that is present in medical documents to help determine content [9].

3 Related Work

The summary component of PERSIVAL makes two key contributions to work in automatic summarization: refined handling of multiple documents and customization at different levels of granularity. We discuss related work with respect to these two aspects.

Automatic text summarization identifies and extracts the key points from text and then presents a condensed version to the user. Text summarization techniques fall into two categories: those that use sentence extraction and those that use sentence reformulation. The first approach identifies the most meaning-bearing sentences in the input and concatenates these sentences to form a summary [10–12]. This approach was mainly used for single-document summarization, although it has begun to be used for multi-document summarization also using a variety of similarity based models [13–15]. A second, generative approach extracts key concepts and uses these concepts to compose a new summary text. This is used mainly in multi-document summarization, where salient points can come from different documents. Our approach uses sentence regeneration – a balance between full sentence extraction and full sentence generation – that is most similar to [16,17]. In multi-document summarization, the repetition of information from different sources is often used as a measure of importance [18,19]. PERSIVAL exploits structural regularities in medical documents to guide the identification of repetition and contradiction. This is similar in purpose to discourse-based approaches in both single- [20] and multi-document summarization [21]. In addition to identifying repetitions and contradictions across source articles, PERSIVAL customizes its summaries along several dimensions, including target audience and the patient's medical record. This guides the summarization process in pinpointing useful information and organizing it appropriately.

There are several information systems targeted at physicians. The most widely used literature resource for medical specialists is PubMed. [3] The user searches the literature by running queries and getting a list of related articles. The level of customization is limited: the user can save query sessions, but no user model is stored to make the search more relevant to the user's information needs. Pratt and Sim [22] describe a dynamic user profile for physicians which evolves over time: the Physician's Information Customizer (PIC) elicits the stable, long-term attributes of a user model through a simple questionnaire about the physician and basic patient characteristics. The utility of this user model was demonstrated in a search application using Medline. These two systems – PubMed and PIC – provide the user with ways to narrow the search mechanism and return a list of relevant articles. PERSIVAL, in contrast, provides physicians with customized summaries of the relevant articles. This is similar in spirit to the work by Becher *et al.* [23], which examine this issue from a user interface perspective.

[3] `http://www.ncbi.nlm.nih.gov/entrez/query.fcgi`

Osman *et al.* [24] showed that health consumers are able to make more effective choices with better information. Several systems are designed to help the patients be more informed [25–28]. These systems automatically generate personalized pamphlets or treatment information based solely on patient information. PERSIVAL, in contrast, provides lay people with summaries of health literature from a digital library.

Customization of medical applications can be achieved in many ways. Carenini *et al.* [25] use a patient profile collected through a questionnaire filled out by the user. Becher *et al.* [23] propose a set of basic information-need scenarios to be chosen from by the user and filled out with specific details. By using a question/answer format, the user can inform the system of his/her priorities and of preferences for organizing results. A concern with these approaches is that the cognitive load of providing the user model is on the user, not the system. In the case of a physician, for each new patient, a new scenario must be filled out, while in the case of a lay person, as information changes over time, it needs to be restated [29]. Binsted *et al.* [26] and Bental *et al.* [28] rely on an existing patient record from a hospital. Similarly, PERSIVAL's customization relies both on the existing patient record and on user expertise (physicians vs. lay people). This method requires no intervention on the part of the user. The burden is then shifted to selecting automatically the most pertinent information from a very large and complex record.

4 User Modeling in PERSIVAL

Digital libraries often serve a wide spectrum of users. In domain-specific digital libraries, users can be classified according to their level of domain expertise. In the medical setting users can range from naïve lay consumers, to educated non-specialists, to medical students in training to specialized clinicians. In order to address this range in PERSIVAL, our initial approach is to distinguish between the two endpoints of this spectrum: domain experts and lay people. This is augmented with a complex individualized user model, tailored to a specific patient, as reflected in the patient record. While differing populations have access to medical information, their differing user needs dictate what types of sources should be accessed and what information should be presented to them. By addressing the two opposing user groups, PERSIVAL lays the groundwork for modeling the range of groups lying along the user continuum. Such modeling is currently beyond the scope of this project, but our approach enables a more shaded user model to be inserted.

As domain experts, physicians are highly knowledgeable about their field of practice; they need access to the latest findings published in the medical literature to keep abreast of new developments in the field. When treating a specific patient, physicians usually refer to medical literature to answer their questions. When it comes to open-ended questions, such as "what is the best treatment for this patient?", one source for questions such as these are clinical studies.[4] In the search and selection process for relevant clinical studies, the physicians are guided by their medical knowledge about the patient. In the requirements gathering phase of the project, we observed that physicians do not read a study from begin-

[4] Another source is textbooks which seems to be especially useful when looking for answers to specific question. Summarizing relevant information from textbooks is the focus of another researcher on the project.

ning to end [30] to determine if an article is relevant. Rather, they quickly glance through the "Methods" section describing the patient population, and then focus on the "Results" section. If a clinical study is found to be relevant to the specific patient, then the physician will read the article in more detail.

At the other end of the spectrum are lay people, who often do not possess the medical knowledge required to understand technical articles. Consumer-oriented literature is more appropriate for this population, as it gives high-level descriptions of medical conditions, drugs or treatments, with explanations for technical terms when needed. A less obvious but essential difference from physicians is that lay people, because they do not master the technical terminology, are less able than physicians to express their questions in a precise manner. Belkin *et al.* refer to this as the Anomalous State of Knowledge [31,32]. A high-level overview that facilitates browsing for information would help those lay users who cannot precisely express what they are seeking and who are not sure of what kind of information is available. Other patients who are well-informed and search for specific facts, can be provided with a specific synopsis in response to a direct request.

The type of presentation PERSIVAL provides depends on the information access task a user performs. [5] Following the literature [33], we identified three possible information access tasks that a user can perform in the framework of PERSIVAL:

Browse: the user does not know in advance what he is looking for and needs to be provided with a high-level overview and strong navigational anchors to help him navigate through the information;

Search: the user is well-informed and has a specific query in mind and needs to be provided with an indicative summary so he can decide which document to focus on;

Get a briefing: the user is knowledgeable, but would like to find some guidance in answering an open-ended question. He looks for a mix between an informative and an indicative summary.

The user model in PERSIVAL is designed to capture these differences in information need and access strategies between lay people and physicians. Based on our user population analysis done in accordance with the guidelines in [34], we propose that three basic dimensions need to be encoded in the user model: 1) the domain expertise of the end user (physician versus layperson), 2) the identity of the patient being treated (a patient of the physician, or the patient end user himself), 3) the user's access task (browsing, searching, or getting a briefing). The attributes and their possible values in our implemented user model are shown in Figure 2. In PERSIVAL, we address three instantiated user models: (physician, specific patient, get briefing), (lay, self, search) and (lay, self, browse). In practice, we have found these dimensions to be more important than others. When information is needed at the point of patient care, physicians will prefer just one access task (get briefing), while lay people will tend to look for information for themselves (self).

We have developed different summarization strategies to address these instantiated user models. For the summarization for physicians, the users' queries involve specific patients

[5] Search and retrieval are not a focus of this paper. See [5] for a description of PERSIVAL's approach to search of heterogeneous resources.

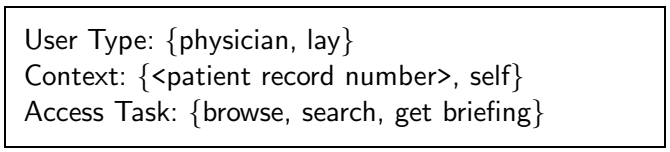

Fig. 2. The PERSIVAL user model, its dimensions and their possible values

and thus we need to take elements from the patient record into account. In contrast, the strategy for lay users does not use data from the patient record, but provides a generic type of interface that better conforms to a layperson's spectrum of information need. The outputs of the two modules also differ. The strategy for physicians constructs a briefing that provides the user with the findings that are specific to the patient at hand. The strategy for lay users handles the two other types of information access tasks: for browsing, it provides high-level overviews, whereas for searching, it generates text indicating differences between documents. In the next two sections we describe the two summarization strategies in turn.

5 Summarization of Technical Articles for Physicians

The PERSIVAL summarizer aims to provide physicians with summaries of clinical studies tailored to specific patients. The strategy responsible for this is implemented as a component of PERSIVAL's summarizer, called TAS (Technical Article Summarizer). In this section, we first describe the challenges of this summarization task and our contributions. We then explain the characteristics of TAS; the types of input it expects, the main properties of the output, and its architecture. Lastly, we report on our current status in the development cycle and the evaluation task.

5.1 Challenges and Contributions

One of the biggest challenges when designing a summarizer is to specify what constitutes a good summary. Sparck-Jones [35] argues that "*context factors*" (what information is available as input, what are the requirements for the output, but most importantly what is the purpose of the summaries) must be taken into account. In our framework, the purpose of the summaries is to synthesize information relevant to the patient under care, helping the physician to make more informed treatment decisions. The requirements for the output are harder to define. One way of capturing them is to use human-produced summaries as models [36,37]. However, to our knowledge, there is no collection of existing summaries of multiple journal articles pertaining to specific patients. As an alternative, we have designed TAS in an iterative fashion [38,39]. At each round of iteration we implement a prototype summarizer and have it informally evaluated by medical experts. They give us feedback as to which features to keep, which ones to drop, and which ones are missing. In addition to asking for specific feedback on the summaries themselves, we also observe how summaries might be used in the context of their task. The next prototype takes the feedback into account, responding more closely to user needs with each successive iteration.

The key contributions of TAS can be found at different stages of the summarization process. Our approach features the ability to tailor the generated summary to an individual patient. This is achieved by pairing information extraction techniques and filtering through the user

model derived from the existing patient record. In contrast with other approaches, TAS is able to merge and dynamically order the different extracted pieces of information to obtain a coherent and fluent text, thus avoiding repetitions and highlighting possible contradictions across the input articles. This is accomplished using a semantic representation of the extracted information. Finally, when presented to the user, TAS links the summary text to the original documents so that the physician can focus on reading any specific input article when needed.

5.2 Input Characteristics

The articles. The set of documents to be summarized is the result of a search on the digital library, restricted to medical journals. However, there are many different types of publications (letters to the editor, case reports, reviews, or clinical studies). Based on our initial user study, we restricted ourselves to summarizing clinical studies. To ensure that documents fed to TAS would only be clinical studies, we implemented a categorization tool that automatically filters out documents that do not fall into this category. The categorizer was trained on 2,700 articles and tested on 1,000 articles with a general accuracy of 96%. The classification for the "clinical study" category achieves 92.2% precision and 97.7% recall.

Articles in the digital library are stored in HTML format. In a preprocessing stage, each input article is transformed into an XML file, in which the title, authors and sections are identified. In addition, the words are tagged with part-of-speech information (*e.g.*, *noun*, *verb*). Using the comprehensive medical ontology of medical concepts UMLS [40], medical terms are identified and tagged with their unique UMLS concept identifier, or CUI. For instance, the phrases "coronary artery disease," "coronary heart disease," and the acronym "CAD" are all encoded in UMLS under the same CUI, *C0010068*, along with 36 other spellings/terminologies for this specific concept. [6]

The patient record. To select the salient pieces of information from the clinical studies, TAS uses the patient record stored in the user model. An electronic version of the patient record is available to us on WebCIS (Web-based Clinical Information System) [3]. WebCIS is used by physicians at New York Presbyterian Hospital to review and enter data in the electronic medical report. However, the patient record in its raw form contains a large amount of information collected over time in many reports. Some are in tabular form (*e.g.*, laboratory tests), while others (*e.g.*, discharge summaries) contain non-structured text. Only a subset of the information present in the patient record is relevant for our task. We use the preprocessing tool MedLEE [42,43] to extract and structure the most recent information contained in the patient record. A simplified extract of a pre-processed patient record is shown in Figure 3. As in the case of the input articles, the medical terms are identified and annotated with their UMLS CUIs.

The search query. As described in section 2, the user selects a question to ask; the question is expanded into a query, and a search on the digital library is triggered. Most questions

[6] For more details on the preprocessing of articles, see [41].

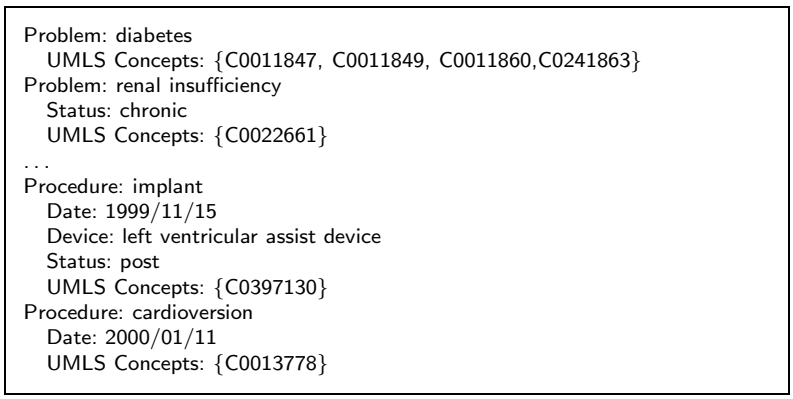

Fig. 3. Extract from a pre-processed patient record

are open-ended (for instance "What is the best treatment for atrial fibrillation given this patient?"). The answer requires subjective judgment and is not explicitly represented in the input articles. TAS, therefore, presents all the information relevant to the patient in the summary needed for the physician to make his/her own judgment about what is best. When organizing the summary content, the query terms are used to increase the importance of individual pieces of information.

5.3 Output Characteristics

Following the analysis of the physicians' needs described in the previous section, we identified the following output characteristics. The summaries produced by TAS are briefings containing results reported in clinical studies (as opposed to the patient group descriptions, methods or discussion of the study). In addition, the results that are included in the summaries are ones that directly pertain to the patient record, often mentioning problems that are characteristic of the patient. Finally the summaries do not contain repetitive information (repetitions are identified and merged together) and signal to the reader any contradictory results found in the input articles.

Figure 4 shows an example of a summary currently generated by TAS for a specific patient. The numbers inside brackets are links to the input articles that were summarized. This patient has a diagnosis of coronary artery disease, diabetes, and renal failure. She also had atrial fibrillation (AF) requiring cardioversion. In this example scenario, she now comes back to the hospital with chest pains and shortness of breath, which indicate recurrence of AF. The physician is primarily focusing on its treatment.

5.4 Architecture

TAS follows a pipeline architecture, shown in Figure 5. First, each input article is classified according to its main clinical task (*i.e.*, diagnosis, prognosis, or treatment). This is used later in the pipeline to help organize the pieces of information in the summary in a coherent manner. Using a corpus of 700 manually annotated clinical studies from 11 journals, we trained a classifier on 511 articles and tested it on 189 articles. The features were extracted from the abstract and the title and included unigrams and bigrams, as well as the semantic types of the medical terms present (automatically provided by the UMLS tags obtained

Summary:
By multivariate analysis, predictors of sotalol efficacy included age < 60 years, higher left ventricular ejection fraction, and absence of hypertension (1). Neither prior electrical cardioversion nor coronary artery disease predicted sotalol efficacy (1).

(2) identified left atrial size as an independent predictor of conversion, but (3) did not. Age and NYHA class were found not to predict conversion (3). Age, sex, and heart rate were not associated with conversion (2).

In a multivariate analysis, the mode of cardioversion was not associated with the recurrence of atrial fibrillation (4). In both univariate and multivariate analysis, coronary artery disease was an independent predictor of recurrence of atrial fibrillation (4).

Articles summarized given this patient record and this question:

(1) "*Efficacy and Safety of Sotalol in Patients with Refractory Atrial Fibrillation or Flutter*" Gallik et al. The American Heart Journal.
(2) "*Efficacy of amiodarone for the termination of persistent atrial fibrillation*". Kochiadakis et al. The American Journal of Cardiology.
(3) "*Spontaneous Conversion and Maintenance of Sinus Rhythm by Amiodarone in Patients With Heart Failure and Atrial Fibrillation: Observations from the Veterans Affairs Congestive Heart Failure Survival Trial of Antiarrhythmic Therapy (CHF-STAT)*". Deedwania et al. Circulation.
(4) "*Patient Characteristics and Underlying Heart Disease as Predictors of Recurrent Atrial Fibrillation After Internal and External Cardioversion in Patients Treated with Oral Sotalol*". Alt et al. The American Heart Journal.

Fig. 4. A summary generated by the PERSIVAL summarizer for the user question, "*What is the best treatment for atrial fibrillation given this patient?*"

during preprocessing). We achieve an accuracy of 84.13% on the testing set. [7] At the content selection stage, a set of templates is instantiated for each input article (Results Extraction). The templates that are not specific to the input patient record are filtered out (Patient Matching). During the content organization stage, the relevant templates are clustered into semantically related units, and ordered (Merging and Ordering). Finally, we use language generation techniques to produce a fluent text for the user (Surface Generation). In this paper, we focus on the content selection and organization modules. [8]

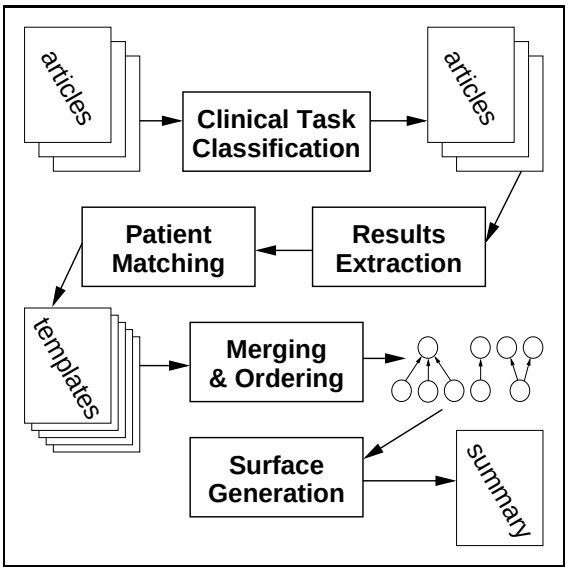

Fig. 5. TAS architecture

[7] For comparison, the baseline classifier, which always considers a clinical study a treatment article, achieves 68.3% accuracy on the testing set.
[8] For more details on the other components of TAS, see [44].

Results extraction. In our processing, we exploit the fixed structure of the articles. Typically, a clinical study follows strict stylistic conventions, providing an abstract followed by the "Introduction," "Methods," "Results," and "Discussion" sections. Information extraction techniques [45] can take advantage of this structure to locate particular pieces of information. When looking for results of a study, we can safely assume that they will be described in the Results section and may be repeated in the abstract.

Based on interviews with our medical experts, we formally defined a result as the tuple (*Parameter(s), Relation, Finding(s)*). We have focused on results that relate disease, patient characteristics, or therapies with outcome. After manually analyzing a small corpus of clinical studies, we identified six types of relations between parameters and findings, namely *association, prediction, risk, absence of association, absence of prediction,* and *absence of risk*. The relations are not independent from one another; for instance, the *prediction* relation subsumes an *association*. However, they reflect the language used by physicians when reporting findings. The relations were approved by our medical experts as being precise and comprehensive.

We collected a set of patterns that match the language used in the Results section, and we use them to instantiate templates. An example of a filled template is given on the left part of Figure 6. For instance the simple pattern "*<parameter> is associated with <finding>*" will match the sentence "*Chest pain is associated with unstable angina,*" such that "*chest pain*" will be assigned to the slot *<parameter>* while "*unstable angina*" will be assigned to the slot *<finding>*. In some cases, a result is reported across several sentences, for instance, "*Age, diabetes mellitus and hypertension predicted heart failure. Heart rate and left ventricular ejection fraction were also predictors.*" We collected patterns dedicated to match sentences such as the second one in this example. The missing slot is then derived from the first sentence; in this example, we infer that the finding for the second sentence is "*heart failure*". In order not to introduce any incorrect inference, we allow such matches only for adjacent sentences.

Our patterns rely heavily on shallow syntactic information; for instance, given the sentence "*Chest pain and male gender were identified as the only independent predictors for unstable angina,*" we can identify the set {*"chest pain", "male gender"*} as the parameters, and "*unstable angina*" as the finding. To do so, we need to identify the noun and verb phrases present in sentences such as the one from Figure 6. Existing state-of-the-art parsers are statistical and have been trained on a large corpus of manually parsed sentences from the Wall Street Journal. When used in technical medical texts, their accuracy drops significantly. To overcome this problem, we customized the shallow parser CASS [46] by writing our own grammar, targeting the style of clinical studies. We focused on identifying accurately noun phrases in sentences that are likely to contain a result (or "result sentences"), like the sentence in Figure 6. The resulting grammar, as expected, would not work well on sentences from the Wall Street Journal. When tested on 59 "result sentences," from 10 different articles, our parser is able to identify noun phrases with 79% accuracy (289 out of 366 noun phrases). [9]

[9] A noun phrase is considered to be accurately identified, when its head, along with its dependencies are identified and nothing else. Examples of correct noun phrases for our task are "*duration of atrial fibrillation < 24 h at presentation*" and "*sinus rhythm (OR 1.8; 95% CI 1.4 to 2.4, p < 0.0001).*"

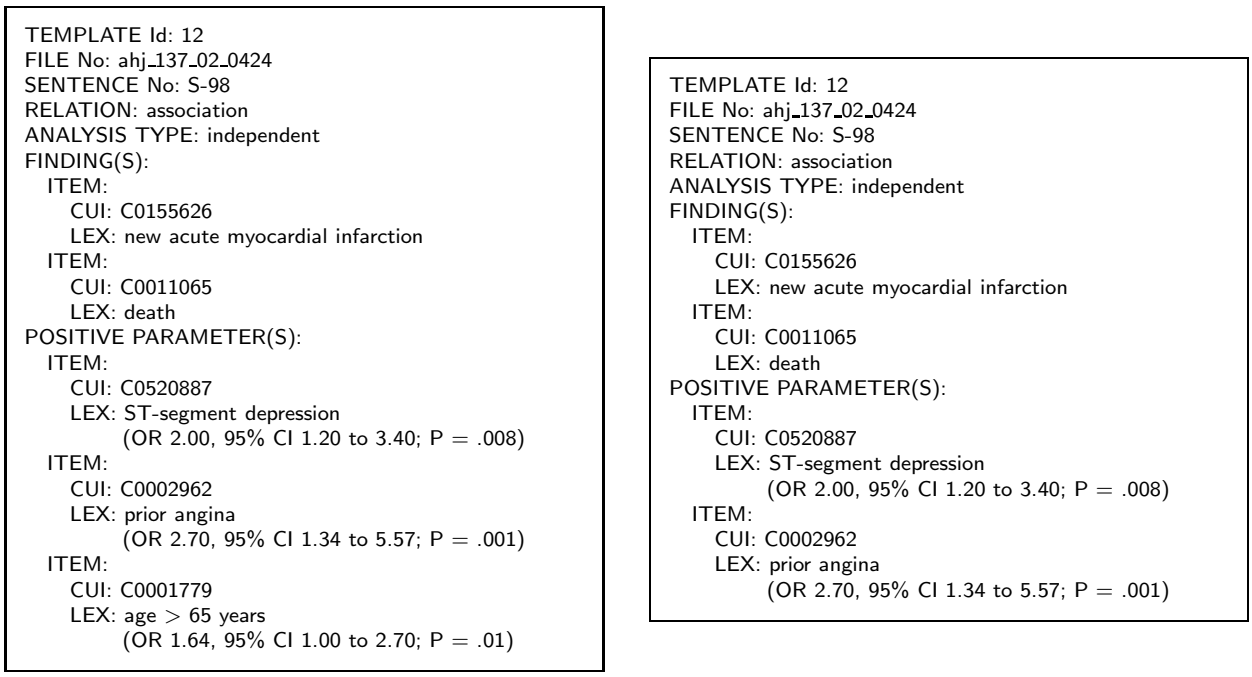

Fig. 6. Template before and after patient matching. The template is instantiated from the sentence "*New acute myocardial infarction or death was associated with ST-segment depression (OR 2.00, 95% CI 1.20 to 3.40; P = .008), prior angina (OR 2.70, 95% CI 1.34 to 5.57; P = .001), and age > 65 years (OR 1.64, 95% CI 1.00 to 2.70; P = .01).*"

Patient matching. Using the patient record stored in the user model, this module determines whether a result, which relates disease, patient characteristics, or therapies with an outcome, is relevant given the characteristics of a patient. If it is, the corresponding template is selected as part of the summary content; otherwise, the template is filtered out.

Our strategy for matching is the following: given a result, with its parameter(s) and finding(s) identified, relevance to the patient is determined by considering the parameter(s) only, not the finding(s). Consider the process of matching the following sentence from an article against a female patient with chest pain: "*In a multivariate analysis, chest pain and male gender were identified as the only independent predictors for unstable angina.*" From a medical point of view, only one result pertains to the patient, namely, "chest pain independently predicts unstable angina". The other parameter (male gender) does not match the characteristics of the patient, and therefore, the result "male gender predicts unstable angina" is not relevant for this specific patient. Whether the patient is diagnosed with unstable angina or not does not affect the matching decision. As an additional confirmation, consider the similar sentence, "*In a multivariate analysis, chest pain and male gender were identified as the only independent predictors for death.*" There is still only one relevant result for our patient, namely, "chest pain independently predicts death." Even though death is obviously not a characteristics of the patient, physicians would consider this result to be related to a patient with chest pain.

Looking up parameters in the patient record is not enough to determine accurately whether a result matches; one also has to take into account the degree of dependence of the parameters with respect to the relation. For instance, in our example sentence, chest pain and male gender are independent predictors. It is medically accurate to infer that chest pain

alone is a predictor. This inference, however, would be medically incorrect if the parameters were dependent predictors, *i.e.*, their combination as a whole relates to the finding. We take this fact into account in our matching policy: for each parameter in the template, we match it against the patient data. When the result contains independent parameters, then the non-matching parameters are simply discarded from the template. In contrast, when the parameters are dependent, we consider the template to match the patient if *all* the parameters match; otherwise the whole template is discarded. The right part of Figure 6 shows a template after matching it against a 44-year old female patient with unstable angina and ST segment depression. Since the parameters are independent, it is possible to discard the non-matching parameter (age $>$ 65 years) from the template.

At this point of the pipeline, the content selection is completed. The next module is responsible for organizing the summary content.

Merging and ordering. The findings and parameters present in the templates can be reduced to the medical terms they contain (identified by their UMLS CUIs). A medical term can be both a parameter in one template, and a finding in another. For instance, the term "*unstable angina*" is a finding in the sentence "*Chest pain was found to predict angina*", whereas it is a parameter in the sentence "*Angina predicts acute myocardial infarction.*" [10] Merging consists of combining all the results extracted in the different input articles into one single internal representation, namely a semantic graph of results.

In the graph, nodes are concepts (parameters and findings of the templates), while vertices are relations from the templates (*e.g.*, prediction or risk). A vertex of type r connects two nodes p and f if there exists a template containing the result (p, r, f), where p is a parameter, f is a finding and r is the relation between them. Vertices have different types, as many as there are different relation types (in our case, six). Nodes are indexed by their CUIs and their optional value. The graph is built in an incremental fashion. For each template, each finding and each parameter is converted into a node; if another node with the same CUI and a similar value already exists in the graph, the two nodes are merged. Figure 7 shows a graph built from different templates.

When building the graph, the templates are simplified as much as possible by splitting them into atomic results. For instance, the template in Figure 6 is turned into four separate templates: *(C0520887 [ST-segment depression], association, C0155626 [acute myocardial infarction])*, *(C0520887 [ST-segment depression], association, C0011065 [death])*, *(C0002962 [angina], association, C0155626 [acute myocardial infarction])*, and *(C0002962 [angina], association, C0011065 [death])*. This is done both for independent and dependent results. From a medical point of view, our representation is accurate as long as no relation from a dependent result is presented as independent in the summary, or vice versa. To ensure that we are accurately representing the results, we store into vertices the template id of the result, so that later in the pipeline when ordering and generating the relations, we present relations from dependent results together.

[10] Some terms are exceptions, for example "*death*" can only be a finding while "*age*" can only be a parameter.

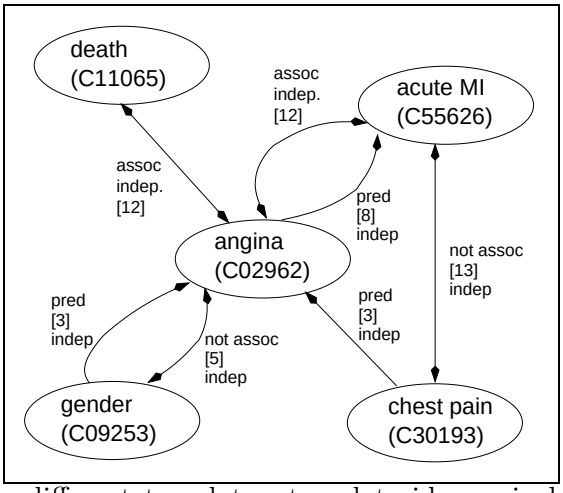

Fig. 7. A graph built from different templates. template ids are in brackets, indep represents a relation from an independent result, pred stands for predict relation and assoc stands for association

These atomic results are used to identify sets of related results. This is done by clustering the atomic templates (using hierarchical complete link clustering). The similarity function between two templates is computed as the sum of the value of three features: whether the parameters are the same, whether the two findings are the same, and how similar the relations are. [11] In the case of templates with an association relation, since it is bidirectional, the tags "finding" and "parameters" are interchangeable, and the similarity function takes this fact into account. The merging step achieves two purposes: it identifies strictly identical templates (that is, repetitions across or inside articles) and it dynamically groups together templates that are semantically related to each other. This step is equivalent to a dynamic paragraph planning, where each cluster represents a paragraph.

The last task left is to decide in which order to present the paragraphs. This ordering is also done in a dynamic fashion. Each cluster gets an ordering weight based on the sum of the value of several features: the number of templates it contains, the number of repetitions, the number of contradictions, and the number of different input articles that contributed to the cluster. The rationale behind this is based on user studies we conducted in the initial phase of the TAS design: as a general policy, physicians want to see the important pieces of information first. For instance, a paragraph which reports on a contradiction between two results is considered important and therefore its corresponding cluster should have a higher weight. An additional feature is whether the cluster contains any node whose medical term is used in the input user query.

Surface generation. Once the content is organized, the final step is to turn the internal semantic representation into a readable text. Each template cluster represents a paragraph; the order in which to present the paragraphs is given by the ordering weight determined in the previous module. For each template cluster, we use phrasal generation [47] to produce the actual text of the summary. The original text for each node in the semantic graph (stored in the LEX slot of the templates; see Figure 6) is used to fill in the phrases. When a node in the semantic graph is referred to by several templates, the shortest string is chosen among

[11] Each possible relation pair was manually assigned a weight. For instance, (prediction, prediction) is weighted higher than (prediction, association).

the present ones. For instance, given the two possible strings "*acute myocardial infarction*" and "*acute MI*", the latter one will be chosen. This simple lexical choice is based on the rule "the shorter, the better".

5.5 Evaluation and Status in the Iterative Design

Following the literature, we want to perform two evaluations of TAS: an extrinsic evaluation which examines the value of the system as a whole, and an intrinsic evaluation which reveals the performance of each module individually. Sparck-Jones [48] emphasizes that while intrinsic evaluation is needed for any system, extrinsic or task-based evaluation is essential to evaluate the system fully. Since our goal is to produce a real system to be used by physicians, it is therefore critical to have it evaluated by physicians in their clinical environment. For instance, existing medical NLP systems which do not deal with summarization have been evaluated as part of clinical trials [49,50]. However, Sparck-Jones [48] also notes that so far, no domain-specific summarization system has been evaluated in context, because a task-based evaluation for summarization is extremely difficult to design and expensive to perform.

We hope that iterative design can help us set up such an evaluation [51]. At each iteration, we conducted a small, informal task-based evaluation with our medical experts. Each subject was presented with an example scenario consisting of a patient, a list of articles, a query, and the summary produced for it. In addition to giving feedback on the system, they also commented on the evaluation setting, helping us construct and refine our set of evaluation scenarios. While all the modules in TAS are implemented, we want to wait until we have a robust prototype to do such a thorough task-based evaluation. By the time we implement the final version of our prototype, we will have already a set of robust evaluation scenarios.

We now turn to intrinsic evaluation. For the Results Extraction module, we computed its precision and recall against the number of findings manually extracted on a test set of 40 articles. For the test set, we manually tagged all sentences that should be extracted along with the slots that should be instantiated. We then ran our extraction module and compared the manual and automatic tagging. A template was considered correct when all its slots were correctly identified. We obtain a precision of 90% and a recall of 65%.

For the Patient Matching module, we want to evaluate both whether our matching strategy has a good influence on content, and whether our representation of the data allows such a strategy to be implemented accurately. However, there is no gold standard to which to compare our output. Instead, at each round of the iterative design, we asked medical experts to evaluate the module qualitatively. While they agreed with the strategy itself — match only with the parameters and not with the findings; perform different types of matching depending on the type of relation — they raised two issues related to our representation of the data.

The first issue concerns the patient data representation. As an example, take the term "*heart rate*". It occurs many times in the patient record, each time with values that are most likely different. In the current prototype, however, when trying to match this term and its associated value against the record, we simply try to match it against each occurrence of

the term in the record until there is a positive match or there are no more occurrences of the term. A more medically accurate approach would take into account the context in which the term appears in the patient record to decide whether or not to match it. For instance, if the input article refers to heart rate post-surgery, there is no point in trying to match its value to the value of the heart rate mentioned in the patient record in the context of pre-surgery.

The second, more critical, issue concerns the tagging of the values corresponding to terms, both in the articles and in the patient record. In a sentence reporting a result in a clinical study, it often happens that the value associated with the term is not precise enough to allow for a direct match against the patient record. For instance, in the sentence "*Other multivariate correlates of heart failure were advancing age, the absence of unstable angina, reduced left ventricular ejection fraction, and an elevated left ventricular end-diastolic pressure.*" the values "*advancing,*" "*reduced,*" and "*elevated*" must be converted into numerical values to match against the data in the patient record. For the moment, TAS does not contain any medical knowledge that would allow inference of such a translation; rather, the matching is done at the term level and these values are dismissed. We are working on building automatically a knowledge base that maps for common terms their qualitative values to quantitative ranges.

The content organization phase (Merging and Ordering) has not yet been fully formally evaluated. Identifying repetitions and contradictions across documents is known to be a hard task [19]. Our method tackles this issue by exploiting a semantic representation, which is similar to the approach described in [52] in the news domain. However, in our domain, contradictions and repetitions are more complex to identify. For instance, in Figure 7, physicians would consider the two relations (*chest pain, predict, angina*) in conjunction with (*angina, predict, myocardial infarct*) a contradiction with the relation (*chest pain, not associated, myocardial infarct*). But, in the general case, we cannot assume transitivity for the relations in the graph, for this might produce inaccurate medical inferences. Physicians would most likely prefer to have more subtle repetitions or contradictions reported to them. This involves formalizing the concepts of repetition and contradiction as well as extending our representation to take these new definitions into account.

From an interface standpoint, in our latest round of informal evaluation, physicians approved the presence in the summary of links to the original articles. Currently a link points to the beginning of an article. Physicians indicated that they would prefer to have a more precise bookmark. In our next implementation, we plan to replace the links to articles by bookmarks that will point directly to the information selected in the articles. The physicians also requested to see displayed next to the summary the portions of the patient record that matched with the results reported in the summary. We plan to incorporate this information in our next implementation.

So far, we have gone through three iterations in the design cycle. In our latest implementation, we use phrasal generation. This approach has been so far appropriate for our generation needs. However, if, for instance, we plan to produce a different style for physicians and for medical students, the use of slotted phrases will be too cumbersome and will not scale up. At this time, we have not formally evaluated the fluency of the produced summaries. In addition to being subjective, fluency is a hard quality to evaluate: if users disagree with the content

of a text, fluency is less important for them. When the system is stable from the content selection and organization viewpoints, we will investigate more closely alternative generation techniques as well as ways to obtain judgments about the fluency of the summaries.

6 Summarization of Healthcare Documents for Lay Users

For lay people such as patients and their families, PERSIVAL needs a different strategy to provide summaries of relevant consumer health documents.

6.1 Challenges and Contributions

Often a lay user's initial search query is not precise, although he may desire a specific type of information. This mismatch between information needs and skill at communicating them is well-known in the library science community [31,32]. Although this problem is not limited to seekers of medical knowledge, we agree with [23] that the problem of underspecification is particularly acute in medicine. In addition, Berland *et al.*'s [53] study of internet-accessible consumer healthcare information concludes that identification of conflicting viewpoints needs to be clearly shown. This motivated us to develop a summarization strategy for lay people that provides resources to help bridge this cognitive gap.

We have developed within the PERSIVAL framework a second summarization module, Centrifuser, to generate three distinct summary components that form an *overview plus details* textual user interface [33] that specifically target this cognitive gap. The three components consist of: 1) hyperlinks to topics related to the query to facilitate navigation and query reformulation, 2) a high-level overview of the commonalities in the documents targeted for browsers, and 3) a description of differences between the retrieved documents to help searchers select relevant items. These components can be seen in Figure 10, which shows a generated summary in response to the search query *"What is atrial fibrillation?"*.

A second key contribution in our work is the ability to highlight information that is typical or unusual across documents. To do this, we have developed an automatic alignment process that discovers regularities in topic structure across a sample of authoritative documents. These regularities are then used to identify which parts of the input articles are rare, atypical information as well as common, typical information, similar to TAS's ability to find repetition and contradiction. The notion of typicality is paired in the content selection process with the standard notion of query relevance to help select summary material. This aspect of the system directly addresses the issues of coverage highlighted by Berland *et al.* and works towards collocating conflicting consumer health information so they can easily be identified.

6.2 Input Characteristics

The articles. For lay user queries, PERSIVAL's search component is set to retrieve documents from authoritative websites and full-text resources that target consumers.

Since these articles often have a strong sense of text structure and organization, we capital-

ize on this aspect in PERSIVAL's consumer health summarization component. Centrifuser begins by building a *document topic tree* for each document. Also known as a document structure tree [54] or a document map [55], a document topic tree represents the document as a hierarchy of topics (*e.g.*, *Treatment* topic consisting of *Dieting*, *Surgery* and *Medication* subtopics). Topic trees represent a balance between representations that focus primarily on structural encoding (*e.g.*, schemas and document type definitions) and representations that focus on content (*e.g.*, document vectors). They enable more precise retrieval by localizing query relevance to specific document sections. For example, while a document describing all cardiac conditions would be relevant to a search on AF, the section on AF would be most relevant; the section on septal defects in the same document would not be as relevant.

Topic trees can be derived from semi-structured data in a number of fashions, including inference from rich markup such as SGML or HTML, from spatial layout recognition [56,57] and from lexical cohesion or chaining [58,59]. We see these two approaches as complementary, and have implemented a system that uses the stronger spatial layout and formatting cues and backs off to use the weaker lexical cues when layout and formatting are unavailable [60].

Consumer-oriented healthcare articles often are well-structured, but their structure is often not codified or regulated, unlike technical articles. *Treatment* and *Symptoms* sections in different articles may use different terminology, or may be arranged in different orders; sometimes they may be missing entirely. Despite this difficulty, consumer health documents from different organizations share similar structure and content. For example, a query on atrial fibrillation retrieves pages on AF from the American Heart Association, British Heart Foundation and local full-text digitized documents from the Merck Manual of Medicine and the Columbia Home Health Guide from PERSIVAL's search component. These documents contain content structured into similar subtopics (*i.e.*, definition, causes, treatments, etc.).

We need to recover these structural regularities to organize and select materials for the summary. To do this, we use an automatic alignment procedure to find structural commonalities and smooth out irregularities. It works by taking a large sample of such authoritative consumer health document topic trees (tens of documents) and iteratively aligning most similar topics across documents using a single metric that combines lexical overlap, topic placement (in terms of node ordering and depth) and parent topic similarity [61]. The resultant *composite topic tree* encodes the typical topic length, frequency, organization and composition. The composite topic tree derived for consumer health, is shown in Figure 8, and shows that information on symptoms usually comes before treatment, and that information on prognosis is rarely given in consumer health.

The search query. The user's search query is used to focus the summarization process and refine the notion of document relevance in PERSIVAL's search module to a fine-grained topical one. The alignment of the query to a node in the composite topic tree achieves this. Child topics within a set depth k away from the query topic form the scope of topics relevant to the query. Topics that are not relevant are either too *intricate* in detail for use in the summary (over k deep descendants from the query topic) or *irrelevant* (outside the subtree defined by the query topic). These relationships partition a topic tree into these three (possibly empty) regions, as shown in Figure 9.

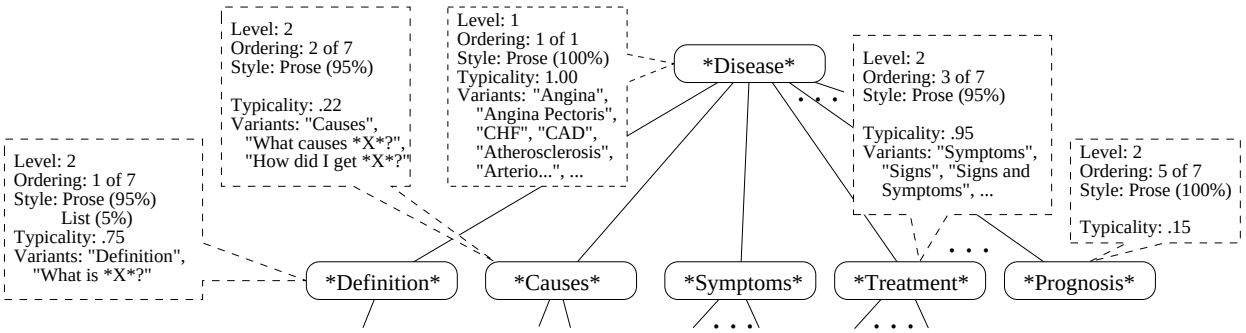

Fig. 8. An automatically constructed composite topic tree for the consumer healthcare domain. Details have been omitted for clarity

Varying k, the depth in the tree, changes the ratio of *relevant* to *intricate* topics, and thus changes the amount of fine-grained detail that appears in the summaries. Through some initial experimentation, we found that a value of k=2 worked well, but note that summarization for other genres, summary lengths, audience and other factors may influence the choice of k.

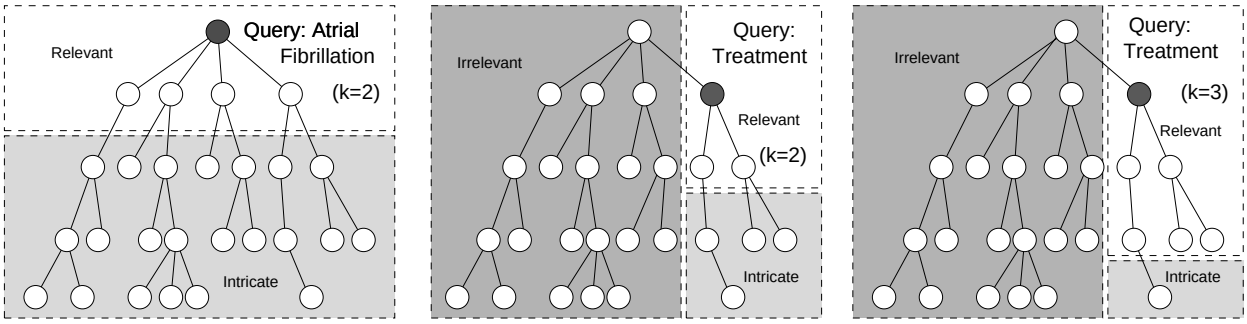

Fig. 9. A pictorial representation of how *relevant*, *irrelevant*, *intricate* topic types are defined by the interaction of a particular topic tree and two different queries

6.3 Output Characteristics

The output of Centrifuser is a tripartite summary which blends standard content-based relevance with structural information. Each component is specifically designed to cover an aspect of a layperson's information need. The summary consists of the following components, also labeled in Figure 10:

(1) **Navigation links.** PERSIVAL uses the composite tree to locate topics related to the user's query, shown as navigation links to activate alternative queries.
(2) **Synopsis of commonalities.** The lay user summarization module also provides a high-level synopsis of the information that is repeated across documents. The topic queried as well as its specific subtopics are summarized using sentence extraction techniques. Such an overview is most useful for browsers with broad information needs.
(3) **Differences across documents.** Searchers need to be provided with descriptions that differentiate documents from each other in content and in form; these are *indicative* differences.

By highlighting the facets of documents that are unique yet relevant to the user's query, we filter out commonality already represented by the synopsis and present the

Overview summary of **Atrial Fibrillation**

We found **9** documents about atrial fibrillation.

(1) **Here are some links to more specific subtopics of atrial fibrillation**: [definition][symptoms][causes][diagnosis][treatment][complications][prevention][prognosis][for more information]

(2) **Here's some general information on atrial fibrillation put together from the 9 documents**: Atrial fibrillation is a common type of palpitation, where you experience an irregular and often rapid beating of the upper chambers of the heart, known as the atria. There are a number of treatment options for AF. The first line of treatment usually involves medications, but there are other treatments which may be appropriate. Symptoms of AF may include one or more of the following: heart palpitations, lack of energy or feeling over-tired, etc. The causes of atrial fibrillation include: rheumatic heart disease, ischaemic heart disease, etc. Treatment varies from case to case and your general outlook will depend on the severity of your underlying heart condition. Medications are prescribed in the management of atrial fibrillation depending on the overall treatment goal: Heart surgery Patients at "low risk" may be given aspirin 325 mg/d to prevent stroke. See also: Anticoagulants, Arrhythmias, etc.

(3) **We've grouped the 9 documents by their topics to guide you in your search for information on atrial fibrillation. Click on a document's title to see it in another window**:

- Atrial Fibrillation Study, Atrial Fibrillation Foundation and The Atrial Fibrillation Page examine key topics. Some of their topics are "controlling heart s irregular rhythm provides no benefit over controlling heart rate", "what is atrial fibrillation af", "atrial fibrillation foundations" and "study findings suggest revised approach to therapy for atrial fibrillation" . The first 2 documents are longer than usual.
- There are 5 documents, including Atrial Fibrillation, that are generally related to your question. These documents discuss topics such as "news and upcoming events", "patient education event atrial fibrillation foundation", "atrial fibrillation foundations" and "boston atrial fibrillation symposium" and are longer than usual. Management of Patients With Atrial Fibrillation contains more figures and tables than usual.
- Primary atrial fibrillation - HUM-MOLGEN doesn't seem to be related to the main sense of your question and contains more material than average.

Fig. 10. Generated summary of documents retrieved for the query "*What is atrial fibrillation?*". Underlined portions represent links to related queries or documents in the result set

distinguishing characteristics of the documents. This assists a user in choosing appropriate documents and drilling down on a subtopic of interest.

6.4 Architecture

Centrifuser detects commonality and uniqueness by comparing topic trees of documents retrieved by PERSIVAL's search component against the composite topic tree. The process of creating a summary is illustrated in Figure 11, which uses three pieces of information – i) the input documents' topic tree, ii) the composite topic tree, and iii) the user's query.

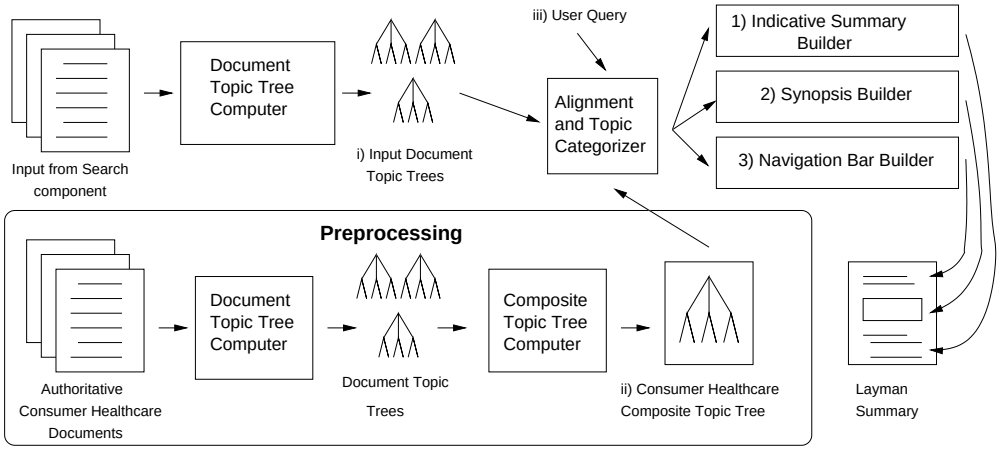

Fig. 11. Centrifuser architecture

We will now illustrate the steps in producing the example text summary, by tracing how

Centrifuser creates each component.

Navigation Links. To allow the layperson to refine or broaden their initial query, Centrifuser creates navigation links for topics related to the query. Clicking on a link causes the entire PERSIVAL system to be reinvoked using the new topic as the search query.

The construction of these links proceeds from the alignment of the user's query to the composite topic tree. A browsing scope centered on the query topic is defined and each in-scope topic is instantiated as a link. This is shown by the dotted outline in Figure 12. This implements the ability to move "up" to a parent topic for generalization to a broader topic; to move "down" to a child topic for specialization and to move "across" to a sibling topic. In this fashion, it is possible to navigate to all topics within the composite topic tree through the navigation interface.

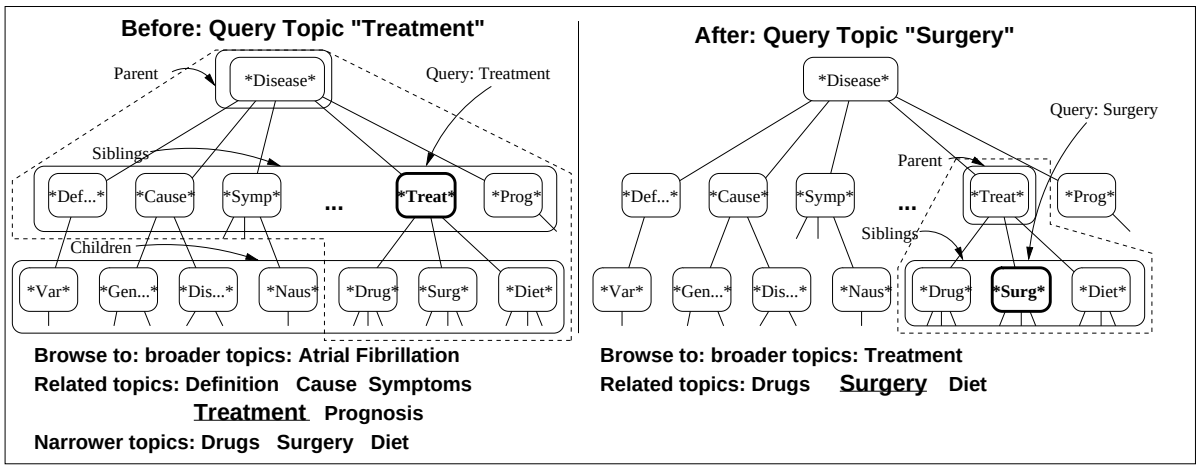

Fig. 12. Navigation browsing scope, indicated by the dashed outline, as illustrated before and after browsing to the "Surgery" topic. Resulting navigation links shown below the tree

Synopsis Based on Similarities. In addition to the navigation links to help locate and navigate the information space, browsers need a high-level synopsis. The idea here is that common information is of higher importance, an assumption often made in multi-document summarization [16,18,62].

The relevant topics in the composite topic tree are the starting point for building the summary since these topics are of the proper granularity. To build the summary, appropriate text is iteratively selected from the input documents by sentence extraction [10] until the summary length specified by the user is met.

The system divides the summary length evenly among all relevant topics. For each topic, the system identifies sections among the input documents that contain information on the topic by using the alignment process discussed previously. Representative sentences are selected from these sections using the SimFinder tool [63], which identifies sets of similar sentences. From each set, the highest ranking sentence is selected using a ranking that combines features that encode each sentence's type (*e.g.*, body, fragment and header), position and length.

The selected sentences are formed into a summary by first ordering the selected topics and then internally ordering each topic's sentences. Topics are organized by their most prominent ordering found in the composite topic tree (*e.g.*, "Symptoms" before "Diagnosis" before "Treatment"). Within a topic, sentences are ordered by their physical position in its respective document.

This process results in an ordered extract focusing on the commonalities across documents. The text's sentences are extracted from multiple input documents and can be viewed as an instantiation of the composite topic tree. By choosing breadth over depth in allotting sentences to aligned topics, the system creates an overview that surveys as many topics as possible. By clustering similar sentences and using only a single sentence per cluster, the system attempts to eliminate redundant information.

Indicative Summaries Based on Differences. Some documents will be more relevant for specific searches and less relevant for others. For example, a document that specializes in AF treatments will be useful for a patient looking for the side effects of certain drugs used to treat AF, but may be useless for another patient who is unsure what type of AF he might have, and is interested in ways to diagnose the different variants of the general condition.

This type of query interaction is modeled by the relevant, irrelevant and intricate regions of the individual document topic trees, as explained earlier in Section 6.2. Each individual document's ratio of topics in these three regions can help us assess the document's importance. In the example, topics in the document which address treatments for AF would be relevant to the former treatment query but irrelevant to the latter diagnosis query.

Relevant topics will also be more interesting than others, by virtue of being more typical or rare. For example, if "Prognosis" is a rare topic to find in a document on AF, it may be worthwhile to report it to the searcher in case he is looking specifically for this hard-to-find topic. Centrifuser models this using the typicality values for the topic stored in the composite topic tree. We use smoothing to assign a very low (e.g, $\frac{1}{|N|}$) typicality value to topics which are not represented in the composite topic tree. For convenience, we set a threshold α, for which a relevant region topic is sub-classed as *typical* if its typicality is higher than a threshold α and sub-classed as *rare* otherwise. Thus, a document that has many rare topics, such as "Prognosis", can be reported to the searcher as it meets criteria for retrieval. Our research employs these four topic categories – *rare*, *typical*, *irrelevant* and *intricate* – for document classification. Each document is assigned to one of seven document categories based on the relative ratio of its topic categories. We explain the categories and give specific details on the categorization criteria in Table 1. The example in Figure 10 illustrates three of these document categories.

The document categories enable PERSIVAL to both cluster and prioritize the input documents with respect to their distribution of information. The final presentation of differences among documents first organizes the documents into separate text bullets (one for each document category), and then uses text generation to create one or more fluent sentences that describes the category, its primary topics and any distinguishing salient metadata (*e.g.*, content type "does the document contain pictures or tables" or audience "does it target medical students"). PERSIVAL uses a nondeterministic grammar of about 40 rules to dy-

Document Type	Topic Distribution	Description
Prototypical	At least 50% of the in-scope topics are *typical* **and** at least 50% of all topics are *typical*	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. The first bullet in the Figure 10 illustrates such documents.
Comprehensive	At least 50% of the in-scope topics are *typical*	Has typical content but also contains other information. These documents cover more topics than usual, and are usually longer. An example of a comprehensive document could be a chapter of a medical text on AF.
Specialized	At least 50% of all topics are *typical*	The document's topics are mostly typical, but the topics in the area relevant to the user query are of mixed type. These documents treat only specific typical topics of interest to the current user query. The second bullet in Figure 10 illustrates some specialized documents.
Atypical	At least 50% of all topics are *rare*	Contains information that may contains information on special cases or exceptions. If the topic "Prognosis" is rare, then a document about life expectancy of patients who experience atrial fibrillation would be an example.
Deep	At least 50% of all topics are *intricate*	Often relevant to the query topic but having much underlying about particular subtopics of the query. An example would be an entire document on "Treatments for atrial fibrillation".
Irrelevant	At least 50% of all topics are *irrelevant*	Contains information outside the scope of interest for the query. An example is the last bullet in Figure 10, which is a bulletin board post and does not match the structure of consumer healthcare documents well.
Generic	*n/a*	These documents display no particular distribution of information.

Table 1
Topic type distribution used in document classification. The examples in the list below pertain to a general query of "atrial fibrillation"

namically generate the category and optional metadata descriptions, resulting in varying sentence and phrase structure. The content selection and ordering strategies used are based on an empirical study of a corpus of indicative summaries from the Library of Congress [64].

As with the extract portion of the summary, the length of this description text is controlled by the user. Since the number of categories are limited to seven and since a single description is generated per category (regardless of its cardinality), many documents can be represented in a single summary.

6.5 Evaluation

In the design and development of the Centrifuser module of PERSIVAL, we have also employed an iterative paradigm. In the first iteration, which involved a limited pilot evaluation, confusing terminology and descriptions were identified and corrected.

We report the results of the second round of evaluation here. Along with a team of evaluation specialists, we assessed Centrifuser in comparison to three currently available Web-based search engines: Google, Yahoo! and About.com. Since About.com is professionally edited, it was selected as one possible upper bound on performance. The goal of this second round of evaluation was to assess Centrifuser's usability and suitability of its summary content.

Search Frameworks:	Centrifuser (fully automatic)		Yahoo (human-edited)		Google (fully automatic)		About.com (professionally edited)	
Content and usability	+	-	+	-	+	-	+	-
Content - Usefulness	10	1	2	1	1	4	3	1
Content - Overall understanding	9	3	1	1	1	1	4	1
Organization of information	2			5	1		5	4
Understandability of labels			5		2		3	7
Navigational ability	3		1	1		1	1	2
Effort to find information	1	1		1		5	1	
Relevance of links		1			1		9	
Amount of information		2		6		2	1	4
Number of links available	2	1	1			4	3	
Range of information available	4		1		1		10	
Search capability		1					1	

Table 2
Frequency of positive (+) and negative (-) coded verbal comments made by subjects regarding the usability and content of the four systems tested

Medical professionals were first consulted to select three widely applicable medical conditions that were used in evaluating the interfaces: diabetes, hypertension (high blood pressure) and angina (chest pain). A total of 13 subjects were interviewed and taped as part of the study. All subjects were recruited from the waiting room at the cardiac and surgical intensive care and were either friends or relatives of patients undergoing treatment at the hospital for one of the three conditions described above. Subjects were presented with their selected query results as displayed by the four systems in random order. While we could not control for the output in the commercial systems, Centrifuser used the documents returned by Google as its input for this experiment rather than documents from PERSIVAL's consumer health collection. In this manner, Centrifuser could be thought of as an alternative interface to Google.

Subjects were allowed to examine the initial screen returned by each search engine. Because of the controlled nature of the experiment, we executed the searches on all four search engines in preparation for the experiment and stored the anonymized results for the experiment to display. This was necessary to remove variables that were not directly associated with the usability of the interface and to control for brand recognition. Subjects were asked to verbalize their thoughts about the content (not layout) as they examined each of the interfaces. We use this "think aloud" protocol as it captures user comments in an unrestricted natural setting. The recordings were transcribed and comments were categorized according to an adapted usability coding scheme [65] by two reviewers, independently.

Data on which features of the systems were identified as being useful or problematic are summarized in Table 2. For example, while subjects liked About.com for its clarity of labeling and its range of linkages to broad resources (see category "Range of Information Available"), they were critical of the relevance of links that Google provided (several subjects commented that they felt Google did not filter their information request very well, providing links to many irrelevant sites). The majority of positive comments regarding the content of information provided (in terms of its usefulness and understandability) were made about Centrifuser. Specific comments were made by several of the subjects regarding the

perceived usefulness of having a synopsis and differences made available to them in response to their queries. Our study shows that the completely-automated Centrifuser module enables lay people to quickly identify high-level information while directing their further knowledge discovery with indicative differences and navigation links to related queries and representative documents. Centrifuser also largely outperforms Yahoo!, which employs an extensive amount of manual editing.

7 Conclusion

This article presents a unified approach to personalization of search results of medical information to varied users. From our observations and formative feedback on our system, we have formalized a user model that reflects the distribution of users' needs in our medical digital library. This coarse-grained user model comprises three dimensions, encompassing users types, their access task, and the patient record. In the implementation of PERSIVAL, we focus on the most salient combinations of these dimensions and introduce fine-grained, individualized customization when appropriate and necessary.

When implementing the PERSIVAL summarizer we rely on a different summarization strategy for each user type. TAS creates summaries for physicians, while Centrifuser creates summaries for lay people. Both summarizers perform multi-document automatic summarization, using categorization, information extraction, and language generation to cull specific facts and passages which can help the user determine relevance.

In the case of the physician user type, our model summarizes clinical studies based on open-ended research questions. Such summaries need to be relevant to the patient at hand, and as such, TAS employs individualized customization based on the patient record to filter out irrelevant information. Regularities observed in the sentential structure of clinical texts motivated our template-based fact extraction approach. To handle the volume of potentially relevant facts, TAS further filters the identified relationships using the patient record. The final, filtered relations allow TAS to use text generation techniques to report sources that reinforce each other's findings as well as point out contradictions between articles.

In the case of the lay user type, supporting information seeking at a broader level was more important than individualized customization. As such, the implementation for lay users emphasizes the simultaneous support of both browsing and searching. As lay articles are written with more variability than technical ones, Centrifuser captures regularities at a coarse-grained level, in the form of composite topic trees which formalize expected information for consumer health texts. This allows for an efficient comparison to discover both what sections are common and what sections are novel in new unseen texts. This module, Centrifuser, acts as a way to filter the voluminous quantity of healthcare information and provides specific information in the summary to help the user determine relevance. This act of information filtering can help address the problem of quality control in consumer information [66] by selecting only information that is stated across many documents.

Our design limits the range of personalization in summarization, by controlling the type of input data, by restricting the user model, and by narrowing the number of access tasks. These limitations have permitted the PERSIVAL project to develop a practical approach

for generating complex summaries to different user groups. We will build on this approach in future work, by looking at different points along the spectrum from domain expert to lay user and by adding individualized modeling based on the patient record to lay user summarization.

Future work in PERSIVAL is also motivated by our evaluations. From our evaluation of PERSIVAL's lay user summarization module, we have found that lay users may be searching for answers to specific questions that can only be found in the technical literature. We are currently working on strategies to summarize clinical studies for lay users by "translating" technical terminology to lay terms. For PERSIVAL's technical summarizer, we plan to refine our implementation and evaluation metrics using iterative design techniques. The culmination will be a full-scale task-based evaluation. In the latest development cycle, we have learned that physicians sometimes want a factual answer to specific queries, rather than summaries. To address this need, we plan to design a new summarization module to extract answers from medical textbooks, which will complement the TAS and Centrifuser components. Thanks to our unified approach, and our modular design of PERSIVAL, incorporating a new summarization strategy is straightforward.

8 Acknowledgments

Much of the material in this article is based upon work support by the National Science Foundation under grants No. IRI 96-19124 and IRI 96-18797 (STIMULATE) and IIS-9817434 (DLI-2). Any opinions, findings, and conclusions or recommendations expressed in this dissertation are those of the author and do not necessarily reflect the views of the National Science Foundation. Some parts of the work were also supported by a grant from Columbia University's Strategic Initiative Fund sponsored by the Provost's Office.

References

[1] K. McKeown, S.-F. Chang, J. Cimino, S. Feiner, C. Friedman, L. Gravano, V. Hatzivassiloglou, S. Johnson, D. Jordan, J. Klavans, A. Kushniruk, V. Patel, and S. Teufel. PERSIVAL, a system for personalized search and summarization over multimedia healtcare information. In *Proc. of the Joint Conf. on Digital Libraries*, 2001.

[2] P Clayton, R. Sideli, and S. Sengupta. Open architecture and integrated information at Columbia-Presbyterian Medical Center. *M.D. Computing*, **9**(5):297–303, 1992.

[3] G. Hripcsak, J. Cimino, and S. Sengupta. WebCIS: Large scale deployment of a web-based clinical information system. In *Proc. of the AMIA Symposium*, 1999.

[4] E. Mendonca, J. Cimino, S. Johnson, and Y. Seol. Accessing heterogeneous sources of evidence to answer clinical questions. *Journal of Biomedical Informatics*, 34, 2001.

[5] N. Green, P. Ipeirotis, and L. Gravano. SDLIP + STARTS = SDARTS: A protocol and toolkit for metasearching. In *Proc. of JCDL*, 2001.

[6] S. Teufel, V. Hatzivassiloglou, K. McKeown, K. Dunn, D. Jordan, S. Sigelman, and A. Kushniruk. Personalized medical article selection using patient record information. In *Proc. of the AMIA Symposium*, 2001.

[7] S. Ebadollahi, S.-F. Chang, H. Wu, and S. Takoma. Indexing and summarization of echocardiogram videos. In *American College of Cardiology*, 2001.

[8] S. Lok and S. Feiner. The AIL automated interface layout system. In *Proc. of the Int'l Conf. on Intelligent User Interfaces*, 2002.

[9] R. Kittredge, T. Korelsky, and O. Rambow. On the need for domain communication knowledge. *Computational Intelligence*, 7(4), 1991.

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

[11] J. Kupiec, J. Pedersen, and F. Chen. A trainable document summarizer. In *ACM SIGIR*, pages 68–73, 1995.

[12] C.-Y. Lin and E. Hovy. The potential and limitations of sentence extraction for summarization. In *Proc. of the NAACL Workshop on Automatic Summarization*, 2003.

[13] J. Carbonell and J. Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *SIGIR*, 1998.

[14] D. Radev, H. Jing, and M. Budzikowska. Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies. In *Proc. of the NAACL Workshop on Summarization*, 2000.

[15] C.-Y. Lin and E. Hovy. From single to multi-document summarization: A prototype system and its evaluation. In *Proc. of the ACL Conf.*, 2002.

[16] R. Barzilay, K. McKeown, and M. Elhadad. Information fusion in the context of multi-document summarization. In *Proc. of the ACL Conf.*, 1999.

[17] D. Radev and K. McKeown. Generating natural language summaries from multiple on-line sources. *Computational Linguistics*, 24(3):469–500, September 1998.

[18] I. Mani and E. Bloedorn. Summarizing similarities and differences among related documents. *Information Retrieval*, 1(1-2):35–67, 1999.

[19] K. McKeown, J. Klavans, V. Hatzivassiloglou, R. Barzilay, and E. Eskin. Towards multidocument summarization by reformulation: Progress and prospects. In *Proc. of the AAAI Conf.*, 1999.

[20] D. Marcu. Discourse trees are good indicators of importance in text. In Inderjeet Mani and Mark Maybury, editors, *Advances in Automatic Text Summarization*, pages 123–136. MIT Press, 1999.

[21] D. Radev. A common theory of information fusions from multiple text sources, step one: Cross-document structure theory. In *Proc. of 1st Workshop on Discourse and Dialogue at ACL 2000*, pages 74–83, 2000.

[22] W. Pratt and I. Sim. Physician's information customizer (PIC): Using a shareable user model to filter the medical literature. In *Proc. of the Int'l Conf. on Medical Informatics (MEDINFO'95)*, 1995.

[23] M. Becher, B. Endres-Niggemeyer, and G. Fichtner. Scenario forms for web information seeking and summarizing in bone marrow transplantation. In *Proc. of Workshop on Multilingual Summarization and Question Answering*, 2002.

[24] L. Osman, M. Adballa, J. Beattie, S. Ross, I. Russell, J. Friend, J. Legge, and J. Grahama Douglas. Reducing hospital admission through computer supported education for asthma patients. *British Medical Journal*, 308(6928):568–571, 1994.

[25] G. Carenini, V. Mittal, and J. Moore. Generating patient specific interactive explanations. In *Proc. of Symposium on Computer Applications in Medical Care*, 1994.

[26] K. Binsted, A. Cawsey, and R. Jones. Generating personalised patient information using the medical record. In *Proc. of Artificial Intelligence in Medicine Europe*, 1995.

[27] E. Reiter and L. Osman. Tailored patient information: Some issues and questions. In *Proc. of ACL Workshop on From Research to Commercial Applications*, 1997.

[28] D. Bental, A. Cawsey, R. Jones, and J. Pearson. Adapting web-based information to the needs of patients with cancer. In *Proc. of Adaptive Hypermedia and Adaptive Web-based Systems*, 2000.

[29] A. Waern, M. Tierney, A. Rudstrom, and J. Laaksolahti. Concall: Edited and adaptive information filtering. In *Proc. of the Int'l Conf. on Intelligent User Interfaces*, 1999.

[30] K. McKeown, D. Jordan, and V. Hatzivassiloglou. Generating patient-specific summaries of online literature. In *Proc. of Intelligent Text Summarization, AAAI Spring Symposium*, 1998.

[31] C. Borgman. Why are online catalogs hard to use? lessons learned from information-retrieval studies. *Journal of the American Society for Information Science*, 37(6):387–400, 1986.

[32] N. Belkin, R. Oddy, and H. Brooks. ASK for information retrieval: Part I. background and theory. *Journal of Documentation*, 38(2):61–71, 1982.

[33] M. Hearst. *User Interfaces and Visualization*, chapter 10, pages 257–324. ACM Press, 1999.

[34] R. Kass and T. Finin. Modeling the user in natural language systems. *Computation Linguistics*, 14(3):5–22, 1988.

[35] K. Sparck-Jones. Automatic summarizing: Factors and directions. In Inderjeet Mani and Mark Maybury, editors, *Advances in Automatic Text Summarization*. The MIT Press, 1999.

[36] E. Reiter and R. Dale. *Building Natural Language Generation Systems*. Cambridge University Press, 2000.

[37] B. Endres-Niggemeyer. A grounded theory approach to expert summarizing. In *Proc. of Intelligent Text Summarization, AAAI Spring Symposium*, 1998.

[38] J. Nielsen. Iterative user interface design. *IEEE Computer*, 26(11):32–41, 1993.

[39] F. Brooks. *The Mythical Man-Month*. Addison Wesley, 1995.

[40] National Library of Medicine, Bethesda, Maryland. *Unified Medical Language System (UMLS) Knowledge Sources*, 1995. `http://www.nlm.nih.gov/research/umls/`.

[41] S. Teufel and N. Elhadad. Collection and linguistic processing of a large-scale corpus of medical articles. In *Proc. of the Language Resources and Evaluation Conf.*, 2002.

[42] C. Friedman, P. Alderson, J. Austin, J. Cimino, and S. Johnson. A general natural-language textprocessor for clinical radiology. *Journal of American Medical Informatics Association*, 1(2):161–174, 1994.

[43] G. Hripcsak, C. Friedman, P. Alderson, W. DuMouchel, S. Johnson, and P. Clayton. Unlocking clinical data from narrative reports. *Annals of Internal Medicine*, 122(9):681–688, 1995.

[44] N. Elhadad and K. McKeown. Towards generating patient specific summaries of medical articles. In *Proc. of NAACL Workshop on Automatic Summarization*, 2001.

[45] A. Bagga. Analysis of the MUC-7 information extraction task. In *Proc. of the Message Understanding Conf.*, 1998.

[46] S. Abney. Partial parsing via finite-state cascades. *Journal of Natural Language Engineering*, 2(4):337–344, 1996.

[47] P. Jacobs. PHRED: a generator for natural language interfaces. *Computational Linguistics*, 11(4):219–242, 1985.

[48] K. Sparck-Jones. Factorial summary evaluation. In *Proc. of the Document Understanding Conf.*, 2001.

[49] A. Cawsey, R. Jones, and J. Pearson. The evaluation of a personalised information system for patients with cancer. *User Modeling and User-Adapted Interaction*, 10(1), 2000.

[50] A. Lennox, L. Osman, E. Reiter, R. Robertson, J. Friend, I. MacCann, D. Skatun, and P. Donnan. Cost effectiveness of computer tailored and non-tailored smoking cessation letters in general practice.

[51] J. Carroll. *Making Use: Scenario-Based Design of Human-Computer Interactions*. MIT Press, 2000.

[52] D. Radev. Learning correlations between linguistic indicators and semantic constraints: Reuse of context-dependent descriptions of entities. In *Proc. of the COLING-ACL Conf.*, 1998.

[53] G. Berland, M. Eilliot, L. Morales, J. Algazy, R. Kravitz, M. Broder, D. Kanouse, J. Mu noz, J.-A. Puyol, M. Lara, K. Watkins, H. Yang, and E. McGlynn. Health information on the internet: Accessibility, quality and readability in english and spanish. *American Medical Association*, 285(20):2612–2621, 2001.

[54] K. Summers. *Automatic discovery of logical document structure*. PhD thesis, Cornell University, 1998.

[55] M. Zizi and M. Beaudouin-Lafon. Hypermedia exploration with interactive dynamic maps. *Int'l Journal on Human Computer Interaction Studies*, 43:441–464, 1995.

[56] J. Hu, R. Kashi, and G. Wilfong. Document image layout comparison and classification. In *Proc. of the Conf. on Document Analysis and Recognition*, 1999.

[57] D. Niyogi. *A Knowledge-Based Approach to Deriving Logical Structure From Document Images*. PhD thesis, State University of New York at Buffalo, 1994.

[58] M. Hearst. TileBars: Visualization of term distribution information in full text information access. In *Proc. of the ACM SIGCHI Conf. on Human Factors in Computing Systems*, 1995.

[59] J. Morris and G. Hirst. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational Linguistics*, 17(1):21–48, 1991.

[60] M.-Y. Kan. Combining visual layout and lexical cohesion features for text segmentation. Technical Report CUCS-002-01, Columbia University, 2001.

[61] M.-Y. Kan. *Automatic text summarization as applied to information retrieval: Using indicative and informative summaries*. PhD thesis, Columbia University, 2003. Chapters 3-4.

[62] C. Monz. Document fusion for comprehensive event description. In *Proc. of ACL-EACL Workshop on Human Language Technology and Knowledge Management*, 2001.

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

[64] M.-Y. Kan, K. McKeown, and J. Klavans. Applying natural language generation to indicative summarization. In *Proc. of the EACL Workshop on Natural Language Generation*, 2001.

[65] A. Kushniruk, V. Patel, and J. Cimino. Usability testing in medical informatics: Cognitive approaches to the evaluation of information systems and user interfaces. In *Proc. of the AMIA Annual Fall Symposium*, 1997.

[66] H. McClung, R. Murray, and L. Heitlinger. The internet as a source for current patient information. *Pediatrics*, 101(6):1065, 1998.