Columbia Digital News System
An Environment for Briefing and Search over Multimedia Information

Alfred Aho†, Shih-Fu Chang*, Kathleen McKeown†,
Dragomir Radev†, John Smith*, and Kazi Zaman†

†Department of Computer Science
Columbia University
New York, NY 10027
{aho, kathy, radev, zkazi}@cs.columbia.edu

*Department of Electrical Engineering
and
Center for Telecommunications Research
Columbia University
New York, NY 10027
{sfchang, jrsmith}@ctr.columbia.edu

Abstract

In this paper we describe an ongoing research project called the Columbia Digital News System. The goal of this project is to develop a suite of effective interoperable tools with which people can find relevant information (text, images, video, and structured documents) from distributed sources and track it over a period of time. Our initial focus is on the development of a system with which researchers, journalists, and students can keep track of current news events in specific areas.

1 Introduction

Our research focuses on the development of technologies to aid people in finding and tracking the information they need to keep current in their jobs and lives. We are developing a system, the Columbia Digital News System (CDNS), that provides up-to-the-minute briefings on news of interest, linking the user into an integrated collection of related multimedia documents. Depending on the user's profile or query, events can be tracked over time with a summary given of the most recent developments. A representative set of images or videos can be incorporated into the summary. The user can follow up with multimedia queries to obtain more details and further information.

Our research is directed at producing a collection of interoperable multimedia tools with which users can manage knowledge, search for and track events, and summarize and present information. Our focus is on the development of efficient algorithms, modular components, and effective presentation methods. In this paper we outline our system architecture and describe the salient components of our system.

2 System Architecture

The architecture of the system that we are developing is shown in Figure 1. Using a combination of textual keywords and visual features, the user specifies the type of information he or she is interested in tracking. The request may be stored in a user profile for long-term gathering and tracking of information. The event-searching and tracking module looks for matching information on the requested topics, using pattern-matching techniques over multiple representations (e.g., text, metadata) and multimedia search techniques to find relevant text, images and multimedia documents. The documents thus retrieved can be then sent to the summarizer and the results can then be presented and viewed.

The summary information extracted from the documents can be stored in database templates. In addition, the user may search online for related databases (e.g., in the current news domain, the CIA World Fact Book[1] contains relevant information) from which additional information can be gleaned and merged with information from the set of articles for the summary. At the same time, representative images and videos illustrating new information or common content can be selected to be included in the summary. On presenting the summary and related images or video, the user can request to see additional information or specific sources of interest. This is accomplished through a viewing and manipulation component that may reinvoke multimedia search using image and/or textual features.

Our system also contains tools for knowledge management. The novel aspect of these tools is their ability to collect, categorize, and classify image and video information.

Several screens from a scenario of interaction with our existing system are shown below in Figure 2. After providing a few keywords specifying the area of interest, the user can produce a summary of several articles on the World Trade Center bombing, including

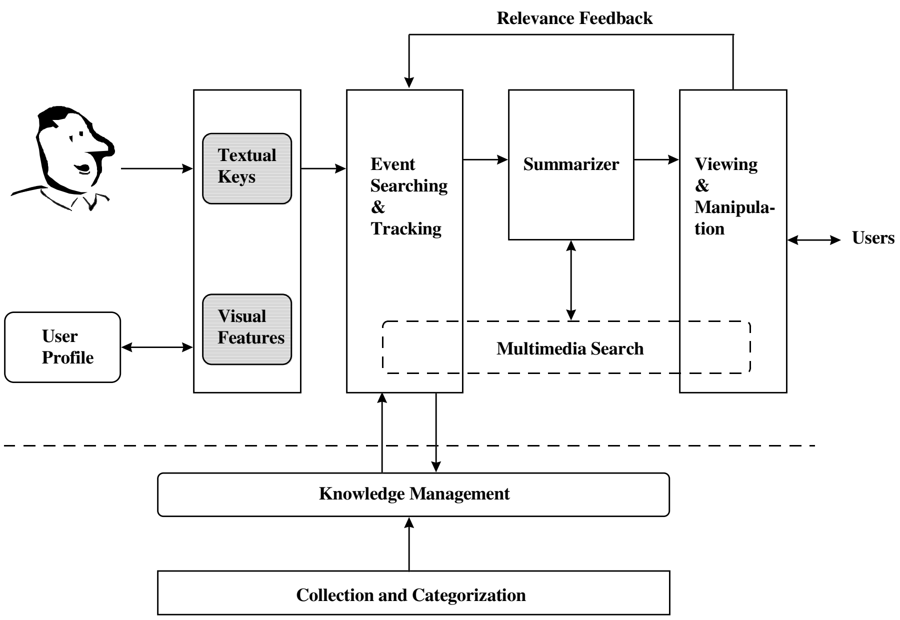

Figure 1: System Architecture

two representative images revealing what maps of the area and what photographs of the event are available. Currently, summarization is done using templates representing information extracted from the news article using a system developed for the DARPA message understanding conference (MUC) [2]. Since the MUC systems only handle South American terrorist news, we also store some hand-constructed templates in the same format for testing purposes. Following receipt of the summary, the user then can search for additional images, here choosing to search for images similar to the map. The image/video search engine uses content-based techniques to retrieve images with similar visual content (e.g., color and spatial structure in this case). This results in a set of 30 most relevant images (not shown) retrieved from our catalog of approximately 48,000 images. The user then refines the image search by adding in textual keywords to specify the topic, resulting in the 3 images shown in the second screen of Figure 2. Finally, the user can request additional textual documents relating to one of the retrieved images, receiving again a summary of the new documents retrieved.

Our current implementation is an early prototype illustrating our vision and therefore many open research issues remain. For example, we need to expand the kinds of media handled as well as interaction between media. By adding multimedia documents (news plus images), we can begin using the text within a document to aid in image searches. We can also use results from automated image subject classification to provide multimedia illustration in text summary. In order to facilitate scaling the system, our work includes development of tools for building resources that can be used for searching, tracking, and summarization. We are developing tools to annotate and catalog images using image features and associated text, tools to extract lexical resources from on-line corpora [3], tools to scan corpora to identify patterns in text not found by information extraction (e.g., role of a person), and are collecting domain ontologies and tools, such as the hierarchies of locations and companies developed for use in the DARPA (Defense Advanced Research Projects Agency) TIPSTER effort (e.g., [4]).

3 Knowledge Management

The Columbia Digital News System contains a number of tools to help collect, categorize, and classify information. In order to track events of interest, CDNS must be able to identify new information that is related to the user's interests. We have tools for automatic information categorization to aid in this task

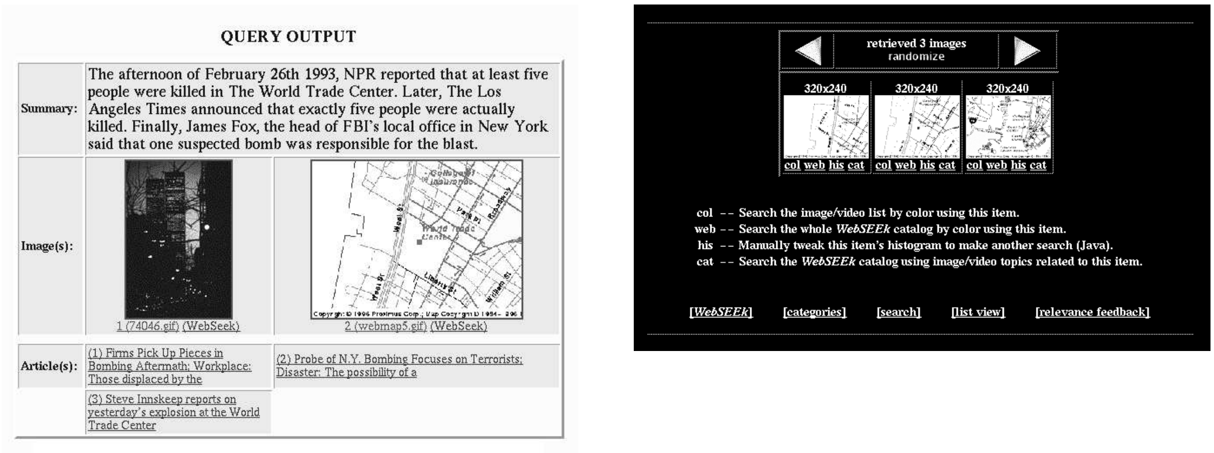

Figure 2: (a) Illustrated summary generated by prototype (b) Results of image search using both visual features and textual keys

as well as in improving the accuracy of any follow-up search and exploration that the user may wish to carry out. When a new image arrives, it can be automatically catalogued in the image taxonomy and the user can be notified if its category matches the user profile.

In addition to including an incoming image to the growing taxonomy, our work also features semi-automatic development of the image taxonomy using both image and textual features, active search of the World Wide Web to categorize images as part of our local collection, and use of the image taxonomy to improve the accuracy of follow-up search. While other groups have used image collections that were manually annotated with textual key words [5], our work focuses on semi-automatic cataloging through incremental classification into and extension of the taxonomy. This classification can then be subsequently used as an additional constraint on search to improve accuracy. In this section, we describe our current prototypes for image and video categorization.

3.1 Cataloging Images and Video

Aiming at a truly functional image/video search engine for online information, we have developed a tool called WebSeek [6] that issues a series of software agents (called spiders) to traverse the Web, automatically detect visual information, and collect it for automated cataloging and indexing. Taxonomies of knowledge are very useful for organizing, searching, and navigating through large collections of information. However, existing taxonomies are not well suited for handling dynamic online visual information such as images on the Web. We have developed a new working taxonomy for organizing image and video subject classes. Our WWW image/video cataloging system is unique in that it integrates both the visual features and text keys in visual material detection and classification. There are a few independently developed systems with similar goals [7, 5], but they do not use multimedia features to construct a comprehensive taxonomy like ours.

Our system uses both textual information and visual features to derive image/video type and subject classes. Type information indicates different forms of object content, such as "gif" and "jpeg" for images; "mpeg", "qt", and "avi" for video; and "htm" for html documents. Subject classes represent the semantic content of an image or video. They provide a valuable interpretation of the visual information, such as "astronomy/planets" and "sports/soccer". We have developed fully and semi-automatic procedures for type/subject mapping.

We use two methods for image type mapping. The first method examines the type of the hyperlink and the filename extensions of the URLs. Mappings between filename extensions and object type are given by the MIME content type labels. This method provides reliable mapping when correct filename extensions are given. The second method is to use the visual features of the images/video for type assessment. This automated procedure involves training on samples of the color histograms of images and videos. Fisher discriminant analysis is used to construct uncorrelated linear weightings of the color histograms that provide for maximum separation between training classes.

Mapping subject classes involves building an image/video subject taxonomy. Figure 3 shows a portion of a working taxonomy. It is built incrementally through the process of inspecting the key-terms associated with the visual information. For example, when new, descriptive terms such as "basketball" are automatically detected, a corresponding subject class is manually added to the taxonomy if it does not already exist, i.e., "sports/basketball".

Subject mapping for new image and video information involves a sequence of steps. First, key-terms are extracted automatically by examining textual information from hyperlinks (URLs and hyperlink tag) and directory names. Then, mapping between key-terms and the corresponding subject classes is done by using a key-term dictionary, which is derived in a semi-automatic process. Based on the term his-

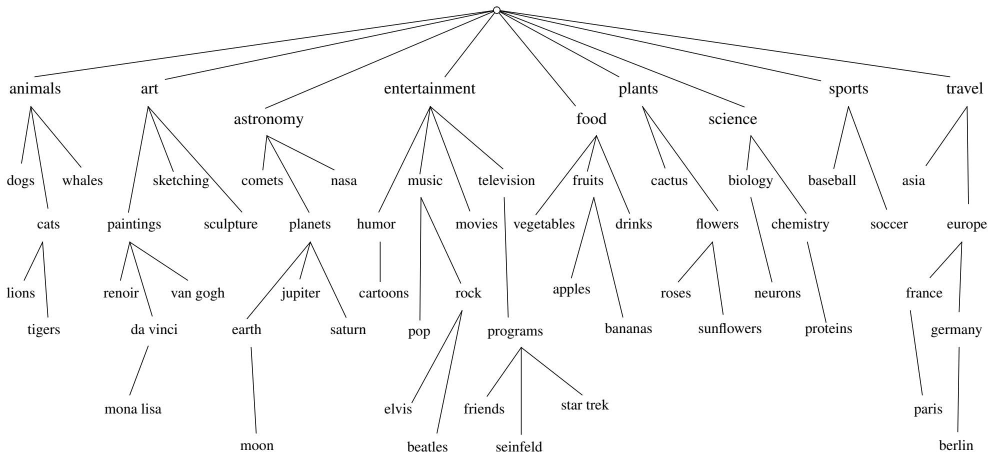

Figure 3: Image taxonomy

togram, terms with high frequency are presented for manual assessment. Terms are grouped into "descriptive" and "non-descriptive" classes. Only descriptive terms, such as "aircraft" and "texture" are added to the key-term dictionary. New images and videos associated with key-terms in the dictionary are automatically mapped to the corresponding subject classes.

3.2 Current Directions

In the area of image/video cataloging, currently we use only limited textual information from URLs, directory, and hyperlink tags. Visual features are used in automated type mapping, but not in subject mapping. We are investigating new techniques using sophisticated combination of visual features and text features. In [8], we used feature clustering to generate a hierarchical view of a series of video scenes and to detect video scenes with specific semantic content (e.g., anchorperson scenes). Several clustering techniques, such as self-organization maps, isodata, and k-mean, have been applied over various visual features (color, texture, and motion). We are augmenting our prior work with two new approaches. First, the working image taxonomy will be used to provide initial clusters of visual features. Consistency relations over various features within each cluster will be measured to assess the association strength of visual features and the semantic classes. Best representative features for different subject classes will be identified through a learning process [9]. Mapping of feature clusters to semantic objects will be derived through training and learning. For example, detection of salient objects (people, animals, logos, skylines) and activities may be achieved by optimal selection of features and their dynamic clustering based on the training data and user interaction.

While we have achieved remarkably good results with the techniques described here, accuracy of classification using textual keys depends on the quality of the URLs, directory names, and type extensions associated with the images. We are currently investigating how best to use text associated with images in multimedia documents or in related web pages to identify good textual keys. We are evaluating the use of statistical natural language tools we have developed for identifying key phrases in a textual documents, including tools for extracting technical terms [10], for extracting commonly occurring collocations [11, 12], and for identifying semantically related groups of words [13]. By extracting phrases from texts that repeatedly occur with the same image, we hope to find terms that reliably provide information about image topics.

4 Searching and Tracking

Two significant features of CDNS are its facilities for searching for visual and textual information and its capabilities for tracking changes and related information. The multimedia search facility in CDNS allows users to find specific information by specifying query keys including both textual and visual properties. Tracking modules allow a user to produce an initial set of multimedia information relevant to his or her interest. The summarizer can then be invoked to produce briefings, possibly containing representative images for illustration. Scanning these multimedia briefings, users can examine the source documents and pursue an active process of in-depth information exploration.

4.1 Content-Based Visual Query and Search

Content-based visual query (CBVQ) techniques provide for the automated assessment of salient visual features such as color, texture, shape, motion, spatial and temporal information contained within visual scenes [14]. By computing the similarities between images and videos using these extracted visual features, new powerful functionalities are added to the system:

- query based on features of visual information
- classification of images and video into meaningful semantic classes

- browsing and navigation by content through image and video archives [15]
- automated grouping of visual scenes into visually homogeneous clusters [8]

Much work in computer vision and pattern recognition has dealt with related problems, such as object segmentation and recognition. Some techniques are suited for generalization to tasks for visual query, such as texture analysis, shape comparison, and face recognition. However, CBVQ has new characteristics and requirements. Content analysis and feature extraction need to be fast for real-time database processing. Multiple features, including those from other media (e.g., text, audio), should be explored for object detection and indexing. Relevance feedback from user interaction is useful for adaptively choosing optimal features for different domains.

With these in mind, we are focusing on the following fundamental research issues in this area:

- fully automated extraction of visual features from both compressed and uncompressed images
- effective functions for measuring visual content similarity
- efficient indexing data structure enhancing fast feature matching
- fusion of multiple features and support of spatial queries
- linking low-level visual features to high-level semantics through intelligent clustering and learning

Our unique approach to processing compressed images is worth more extensive discussion. Due to their huge size, images and video are typically stored in compressed forms. A unique feature of our approach is the use of compressed data for feature extraction. Given compressed visual information (such as jpeg, wavelet images, or mpeg video), localized color regions, texture regions, scenes, or video objects are extracted without decoding the compressed data. Fundamental processes used in typical compression techniques, such as spatial-frequency decomposition for still image compression and motion analysis used in video compression, provide very useful clues for extracting visual features directly from the compressed images and video. Content-based queries are performed based on the extracted features. Uncompressed visual information is needed for display of final selected images or video only. The compressed-domain approach provides great benefits in reducing the resource requirements in computation, storage, and communication bandwidth. Data requirements are usually reduced by orders of magnitude (e.g., 10 times reduction in jpeg, 30 times reduction in mpeg). Required space and computation for decoding can be reduced as well.

Various visual similarity measurements have been investigated, including quadratic correlation distance and intersection of color histograms [14, 16], Mahalanobis distance of transformed vectors of texture [17, 18], and distance of absolute spatial orientations [19, 20]. Spatial indexing methods have been used to avoid exhaustive search of the entire collection and computing distance of each feature of every image in the collection [21]. Initial work has been reported for joint CBVQ search and spatial query [22]. We have developed a powerful technique, binary feature map, in combining feature representation and indexing. Color and texture image regions are represented by a unified binary vector. Efficient indexing methods (similar to file inversion) and fast distance comparison can be derived with binary feature vector to greatly reduce computation. We are developing techniques to combine this approach with spatial indexing structures like modified 2D strings and quadtrees for supporting fast processing of visual query including both localized content and spatial orientation.

Figure 4 shows the system architecture of a CBVQ system. Images stored in the archive are processed with automated content analysis tools to extract the prominent image regions with coherent features. The input query (either drawing or example image) is processed on the fly to obtain descriptions of its prominent regions. The associated visual features and their spatial information are compared against the image regions and features in the archive to find the matched results.

We have developed efficient algorithms and several CBVQ prototypes. Our early tool, VisualSEEk [15], is a fully automated image query system which supports both localized content query and spatial query. It allows users to specify visual features using a WWW/JAVA graphic interface. Users may specify arbitrarily-shaped regions with different features (e.g., color and texture) and their spatial relationship. Images matching the specified content are returned to users for viewing or further manipulation. The JAVA user interface for VisualSEEK is shown in Figure 5.

An example of visual query using VisualSEEK is as follows. A user wants to find all images containing *sunsets* in the archive. By specifying a golden yellow circle in the foreground and a light brown rectangle in the background, he finds several sunset images within just seconds. The matches in terms of color regions and their spatial layout can be exact or best. From the returned images, he may choose one or more and ask the system to return other images similar to the selected one(s). Through multiple iterations of these two types of queries (query by example vs. query by sketch), the user can navigate through the entire collection. Relevance feedback is provided through the selection of relevant images in the returned batch.

In a follow-up more advanced CBVQ tool called WebSEEk, we have integrated the search methods to encompass both visual features and text keys. To search for images and videos, the user issues a query by entering terms or selecting subjects directly. The tool then extracts matching items from the catalog.

The overall search process and model for user-interaction is depicted in Figure 6. As illustrated, a query for images and videos produces a search results

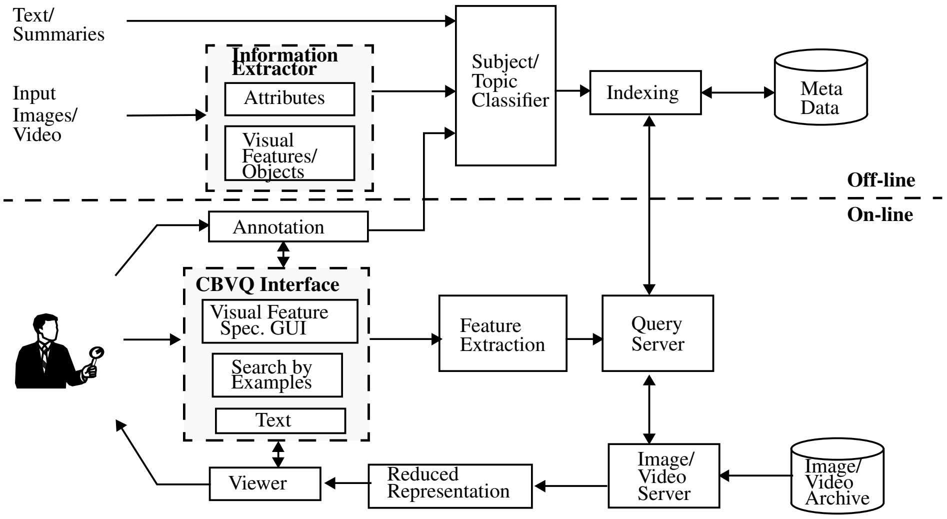

Figure 4: CBVQ architecture

list, list **A**, which is presented to the user. For example, Figure 7(a) illustrates the search results list for a query for images and videos related to "nature", that is, $\mathbf{A} = \text{Query}(\texttt{SUBJECT} = \text{"nature"})$. The user may manipulate, search or view **A**.

After possibly searching and/or viewing the search results list, the user can feed back to the manipulation module the list as another list **C**, as illustrated in Figure 6. The user manipulates **C** by adding or removing records. This is done by issuing a new query that creates a second, intermediate list from which the user can generate a new search results list **B** using various list-manipulation operations, such as union and intersection.

The user may browse and search the list **B** using both content-based and text-based tools. In the case of content-based searching, the output is a list **C**, where $\mathbf{C} \subseteq \mathbf{B}$ gives an ordered subset of **B**, and **C** is ordered by highest similarity to the user's selected item. In the current system, list **C** is truncated to $\mathcal{N} = 60$ records, where $\mathcal{N}$ may be adjusted by the user.

For example, $\mathbf{C} = \mathbf{B} \simeq \mathbf{B}^{\text{sel}}$, where $\simeq$ means visual similarity, ranks list **B** in order of highest similarity to a selected item from **B**. The following content-based visual query

$$\mathbf{C} = \text{Query}(\texttt{SUBJECT} = \text{"nature"}) \simeq$$

$$\mathbf{B}^{\text{sel}}(\text{"mountain scene image"}),$$

ranks the "nature" images and videos in order of highest visual similarity to the selected "mountain scene image." Alternatively, the user can select one of the items in the current search results list **B** to search the entire catalog for similar items. For example, $\mathbf{C} = \mathbf{A} \simeq \mathbf{B}^{\text{sel}}$ ranks list **A**, where **A** is the full catalog, in order of highest visual similarity to the selected item from **B**. In the example illustrated in Figure 7(b), the query $\mathbf{C} = \mathbf{A} \simeq \mathbf{B}^{\text{sel}}(\text{"red race car"})$, retrieves the images and videos from the full catalog that are most similar to the selected image of a "red race car."

4.2 Tracking changes

Changes and follow-ons to a document may be of even more interest than the original document itself. A news story, for example, may have subsequent follow-up stories containing additional information (e.g., suspect was now caught) or corrections to previous stories (e.g., additional victims were found). Ideally, the user would like to track the changes in an ongoing story and have the results reported in a meaningful and understandable way.

Our approach to this problem is to develop tools that identify and track changes in multimedia documents. The changes can then be fed into the summarization modules for presentation to the user. Since our current application is journalism, we are investigating difference measures that work well on news stories. A number of statistical and structural differencing approaches have been recently reported, so we are interested in evaluating the relative efficacy of these measures on news stories. We have developed a WebDiff tool that uses combined statistical and structural techniques to measure and display the differences among Web documents.

We use statistical natural-language-processing techniques to focus on semantic differences between two documents. In information retrieval various vector-based techniques have been developed to estimate the similarity of the content of documents. Since news stories are typically shorter in length and we do not wish to go through the extensive computations required for training, we find simpler metrics preferable. We characterize a document by its content words, that is, by its nouns, adjectives, and verbs; other parts of speech, such as articles, adverbs, and prepositions, are ignored. We use a part-of-speech tagger to identify the content words in the documents under consideration. The statistical metrics we currently use are based on the number of content words common to the docu-

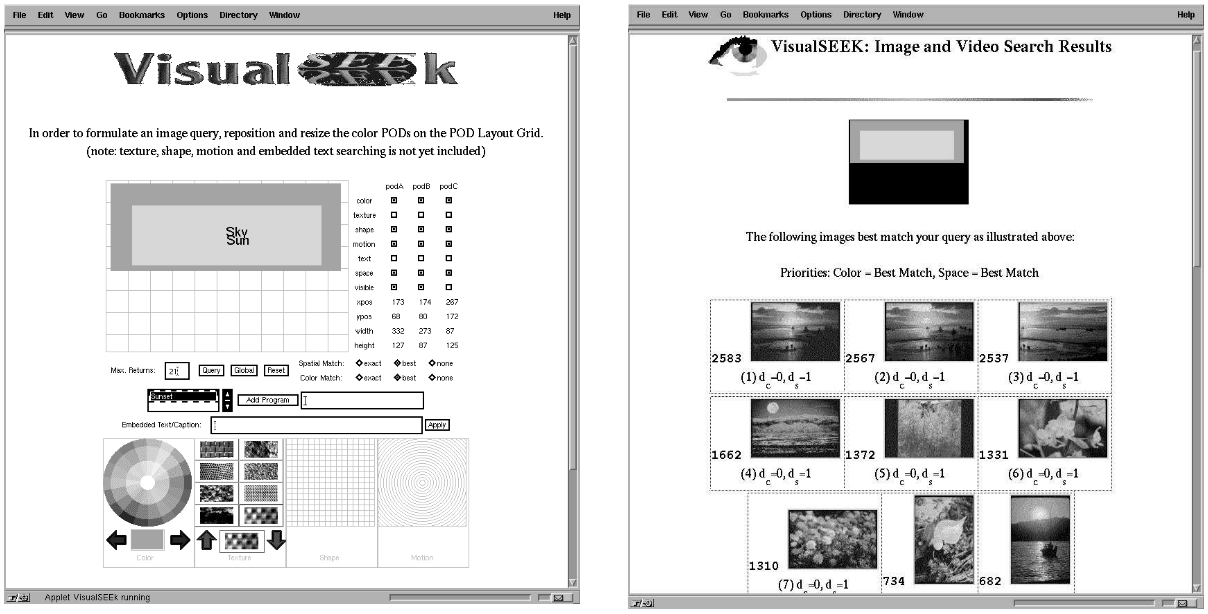

Figure 5: VisualSEEk user interface: (a) Users specify color regions to find sunsets images (b) Results

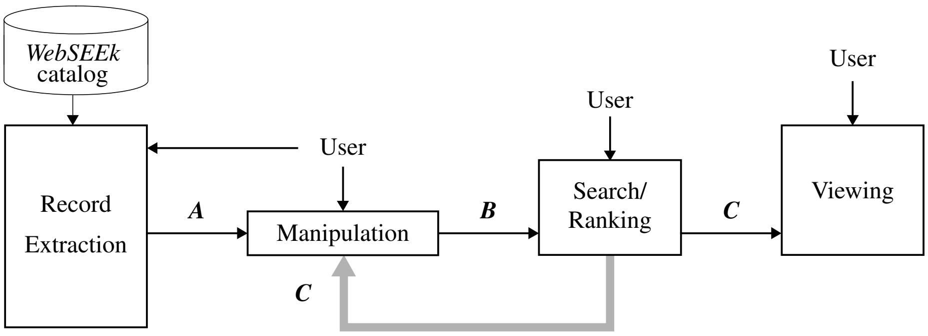

Figure 6: Search, retrieval and search results manipulation processes.

ments.

For structural differences, we are building on the recent work done at Stanford [23] for measuring change detection in hierarchies of objects. The structural properties we employ are based on the grammatical structure of an HTML page as well as other kinds of metadata associated with a document. Although we describe the specific problem of dealing with HTML pages, it should be noted that our techniques are applicable to other domains and formats as well.

Our approach consists of making use of the inherent hierarchical structure of an HTML document. We parse the HTML documents and construct a formal internal representation for each document. This representation is an annotated tree consisting of nodes of different types including lists (both ordered and unordered), text strings, and anchors. We then find the minimal sequence of transformations required to map one tree into another. Typical transformations are insertions, deletions, and updates of nodes. Often, there are restrictions on some of the transformations – for example, it may not be possible to change one type of node to another. Once we have the minimal sequence of transformations, the differences between web pages can be highlighted in a variety of ways. One approach is to display the changes at the level of the source web pages with insertions, deletions, and changes represented in different colors. Alternative methods include using appropriate symbols to mark up the changes in the original document.

4.3 Current Directions

Our differencing algorithms are based on dynamic programming techniques, which allow various kinds of transformations to be weighted differently. Our goal is to find the appropriate weights for tracking meaningful differences in news stories. The research challenge is to find metrics that work well on documents containing images as well as text.

We are also advancing content-based visual query

(a) (b)

Figure 7: (a) Search results for `SUBJECT` = "nature", (b) content-based visual query results for images/videos ≃ "red race car".

by developing effective query interfaces that support flexible combinations of visual features. For example, we are integrating image features (e.g., color, texture, and spatial relationships) with video features (e.g., motion, trajectory, and temporal relationships). In future applications, we can link our image search engines to video browsing/editing systems, which we believe will result in a more powerful environment for visual information access and manipulation.

5 Summarization

The goal of summarization within CDNS is to brief the user on information within the documents retrieved by tracking. In many cases, the summary may provide enough information for the user to skip reading the original document. In others, the user may want to check the original documents to verify information contained within the summary, to follow up on an item of interest, or to resolve conflicting information between sources. In a user focus group, journalists indicated that the summary can provide adequate information to determine reliability of the information presented and whether it is worth following up on the original sources.

Summary generation is strongly shaped by the characteristics of the tracking environment. Tracking of documents on the same event happens over time. At any particular point in time, the summarization component will be given a set of documents on the same event from a specific time period and at later points, will receive additional later breaking news on the same event. To meet the demands of this environment, our work on summarization includes the following key features:

- Summaries are generated over *multiple articles*, merging information from all relevant sources into a concise statement of the facts. This is in contrast to most previous work that summarizes single articles [24, 25, 26, 27].
- Summaries must identify how perception of the event changes over time, distinguishing between *accounts* of the event and the event itself [28].
- Given access to live news, the summarizer must provide an update since the last generated summary, identifying new information and linking its presentation to earlier summaries.

Given the multimedia nature of the tracking environment, presentation of tracked information must include more than just textual summaries. While our work does not include methods for summarizing a set of images returned from search, we do make use of our categorization tools to select a representative sample of retrieved images that are relevant to, or part of, the textual news. If images are an item of interest, journalists can follow up on the summary by requesting images similar to one or more of the representative images.

In the remainder of this section, we present the main phases of summary generation in CDNS. Unlike previous work on summarization, CDNS does not extract sentences from the input articles to serve as the summary. Instead, we use natural language processing techniques to extract structured information from the document and language generation techniques [29] to merge information extracted from different documents and form the language of the summary. Our research focus is on problems in the language generation stage. In the following sections, we describe the tools we have developed and use for information extraction, selection of summary content (conceptual summarization), and determination of summary wording and phrasing (linguistic summarization).

5.1 Information Extraction

Input to summarization is the set of articles and images related to a specific event found by event tracking. Our approach to information extraction is to use and extend existing natural language tools, allowing us to focus on research issues in the language generation component. We describe three tools we are using: a system for extracting information specific to terrorist events, a tool for extracting names and their descriptions, and tools for connecting into online structured data.

Currently we are using an information extraction system for the domain of terrorist news articles developed under the DARPA human language technology program, NYU's `Proteus` system [2]. `Proteus` filters irrelevant sentences from the input articles, and uses parsing and pattern matching techniques to identify event type, location, perpetrator, and victim, among others, building a template representation of the event as shown in Table 1. Coverage of `Proteus` is limited to South American terrorist activities. In order to handle common events in current news (e.g., terrorism in the Mid East or the US), we are enhancing this approach using finite-state techniques [30] for extraction of the fields in the templates.

We have also developed separate pattern matching techniques to extract specific types of information from input text, building *domain knowledge sources* for use in text generation. We have built a prototype tool, called PROFILE, which uses news corpora available on the Internet to identify company and organization names, person names along with descriptions (e.g., *South Africa's main black opposition leader, Mangosuthu Buthelezi* or *Sinn Fein, the political arm of the Irish Republican Army*), building a large-scale lexicon and knowledge source. Descriptions of entities are collected over a period of time to ensure large lexical coverage. Our database contains automatically retrieved descriptions of 2,100 entities at this moment.

Once the descriptions of entities have been extracted, they are converted automatically into Functional Descriptions (FD) [31] which are fed to the natural language generation component. FDs generated in this way can be reused in summaries and can be manipulated using linguistic techniques. Descriptions can be used to augment the output produced by the information extraction component in several ways; it can be used to provide more accurate descriptions of entities found in the input articles or it can be used to find descriptions of entities where the input news does not contain adequate descriptions. By searching through profiles from past news, a description suitable for the summary can be selected. Using language generation techniques such as aggregation (e.g., [32, 33, 34]), the exact phrasing of the description may be modified to produce more concise wordings for the summary. For example, two descriptions may merged into a shorter description conveying the same meaning (e.g., "presidents Clinton and Chirac" instead of "president Clinton and president Chirac").

In order to tap into nontextual sources, we are building facilitators using KQML as a communication language [35] that monitor on-line databases that are available through the World-Wide Web, providing an implementation independent view of the source. These will provide notification when new information arrives and will extract information from remote sites for summarization. CDNS currently includes tools for access to the CIA Fact Book [1] and the George Mason database of terrorist organizations [36], both of which provide information relevant to terrorism and current affairs in general.

5.2 Conceptual Summarization

Conceptual summarization requires determining which information from the set of possible extracted information to include in the summary, how to combine information from multiple sources, and how to organize it coherently.

Our work builds on our prototype system, SUMMONS [28], which generates summaries of a series of news articles using the set of templates produced by DARPA message understanding systems as input (SUMMONS architecture is shown in Figure 10). Output is a paragraph describing how perception of an event changes over time, using multiple points of view over the same event or series of events. We collected a corpus of newswire articles, containing *threads* of articles on the same event where later articles often contain summaries of earlier articles. Analysis of the corpus informed development of both conceptual and linguistic summarization.

Conceptual summarization uses planning operators, identified through corpus analysis, to combine information from separate templates into a single template. The summarizer includes seven planning operators, which identify differences and parallels between templates such as contradictions or agreement between sources, addition of new information, or refinement of existing information. Each planning operator is written as a rule, containing preconditions which must be met for the rule to apply, and actions which extract or modify information from the input templates creating a new template.

An example of a planning operator is **`generalization`**. It uses an ontology developed for the DARPA information extraction systems to determine when values contained in the same slot of different templates fall under the same node in the ontology. The precondition looks for attributes of the same name in different

0.	MESSAGE: ID	TST2-MUC4-0048
1.	MESSAGE: TEMPLATE	1
2.	INCIDENT: DATE	19 APR 89
3.	INCIDENT: LOCATION	EL SALVADOR: SAN SALVADOR (CITY)
4.	INCIDENT: TYPE	ATTACK
5.	INCIDENT: STAGE OF EXECUTION	ACCOMPLISHED
6.	INCIDENT: INSTRUMENT ID	-
7.	INCIDENT: INSTRUMENT TYPE	-
8.	PERP: INCIDENT CATEGORY	TERRORIST ACT
9.	PERP: INDIVIDUAL ID	-
10.	PERP: ORGANIZATION ID	"FMLN"
11.	PERP: ORGANIZATION CONFIDENCE	SUSPECTED OR ACCUSED: "FMLN"
12.	PHYS TGT: ID	-
13.	PHYS TGT: TYPE	-
14.	PHYS TGT: NUMBER	-
15.	PHYS TGT: FOREIGN NATION	-
16.	PHYS TGT: EFFECT OF INCIDENT	-
17.	PHYS TGT: TOTAL NUMBER	-
18.	HUM TGT: NAME	"ROBERTO GARCIA ALVARADO"
19.	HUM TGT: DESCRIPTION	"ATTORNEY" "ATTORNEY": "ROBERTO GARCIA ALVARADO"
20.	HUM TGT: TYPE	ACTIVE MILITARY: "ATTORNEY" LEGAL OR JUDICIAL: "ROBERTO GARCIA ALVARADO"
21.	HUM TGT: NUMBER	1: "ATTORNEY" 1: "ROBERTO GARCIA ALVARADO"
22.	HUM TGT: FOREIGN NATION	-
23.	HUM TGT: EFFECT OF INCIDENT	DEATH: "ATTORNEY" DEATH: "ROBERTO GARCIA ALVARADO"
24.	HUM TGT: TOTAL NUMBER	-

Table 1: Sample template.

templates, with values that can be generalized. All fields that cannot be generalized must not have conflicting values. The action merges the input templates and creates a new template with the generalized values for the attributes. In the example shown, three generalizations are performed. Bogotá and Medellin are both identified as cities in the same country, Colombia. In addition, both *hijacking* and *bombing* appear in the ontology as hyponyms of *terrorist act*. *Tuesday* and *Wednesday* are also generalized to *week*. These more general values are used in a new template. Figure 8 shows excerpts from two templates to which the *generalization* operator can be applied. The combined template will be used as the input to the summarizer to generate the summary sentence shown in (Figure 9).

5.3 Linguistic Summarization

Linguistic summarization is concerned with expressing selected information in the most concise way possible. Input to linguistic summarization is one or more templates created by merging the templates for each article. Each template (both these extracted by the MUC system and those generated by the summarizer using planning operators) will be used as input to generate a sentence. This requires first creating a case frame representation for each template, selecting which field will be realized as the verb of the sentence and mapping other fields to the arguments of the verb. During this process, words are selected to realize the values of the fields. The resulting case frame is passed through syntactic generation, where a full syntactic tree of the sentence is created, grammatical constraints are enforced, and morphological agreement is carried out. We use the FUF/SURGE package [31, 37], a robust grammar of English (SURGE) along with a unification interpreter (FUF) and some text manipulation tools written in Perl as the basis for both word choice and syntactic generation.

Since the FUF/SURGE package provided a complete domain independent module for syntactic generation, our focus was in creating a lexicon representing constraints on how words are to be selected. We identified commonly occurring phrases in the corpus that are used to mark summarized material. For example, if two messages contain information about two separate bombings, then the summary will refer to *a total of two bombings*. In another example, an *update* sentence will be started by *later reports show that ...*.

We also are examining using and modifying phrasing from the original article to aid in developing a robust lexical chooser. By creating functional descriptions for extracted phrases, they can be modified and re-used in novel ways in the output.

5.4 Current Directions

Our current system does not yet incorporate information extracted by PROFILE or from remote databases. We are developing new planning operators that determine when such information is needed and retrieve it from one of these sources. Such information can be used to fill gaps, either in the user's own knowledge or in the template information provided to the generator. It may also be used to connect people, organizations, events or companies mentioned in different articles by providing the relation between them. For example, we might identify the "FMLN" as the "Farabundo Marti National Liberation Front" and include its leader, if the user has not seen any previous summaries about the FMLN. We will use corpus analysis to identify when and how background information is provided in a summary. Yet another issue in this task will be how to combine information at the

template fields	message 1	message 2
MESSAGE:ID	TST-S03-0014	TST-S02-0011
SECSCOURCE:SRC	UPI	Reuters
INCIDENT:DATE	Tuesday	Wednesday
INCIDENT:LOC	Bogotá	Medellin
PERPETRATOR:NUM	1	2
INCIDENT:TYPE	"bombing"	"hijacking"

Figure 8: Two templates to be combined using the *generalization* operator.

A total of three terrorist acts were reported in Colombia last week.

Figure 9: Generated sentence using the *generalization* operator.

semantic level (e.g., from the templates) with phrasal and lexical information (e.g., collocations, technical terms). At one level, collocations and technical terms could augment the HTML links to the original articles (see Figure 2) by serving as subject indices. Alternatively, full phrases could be used to refer to a slot in a template or a relation between slots. For example, if one template has "Garcia Alvarado" as victim and another template has "Attorney General" as the victim, then an extracted phrase such as "Garcia Alvarado, Attorney General" may be used to merge the templates and refer to the slots of both.

As we move to live news feeds, we must modify our summarizer so that it can generate a summary that updates a user on the latest breaking news, without unnecessarily repeating information provided earlier. In order to provide an update, the content planner must compare new information against its model of past sessions with the same user, identifying contradictions and additions. The planning operators will be extended to identify links, similarities and differences between new incoming information and information in the discourse model. In addition to combining information, the planning operators must also be able to filter and order information according to importance. We will experiment with a variety of techniques for this task, ranging from repetition across sources, user ranking of information importance, and empirically derived strategies for ordering information based on a corpus of summaries.

6 Conclusions

In the CDNS project, we have developed tools and a system architecture with which users can manage knowledge, summarize information, and search for and track events. Our work on summarization is mature and unique in several aspects — we have shown that it is possible to obtain highly readable summaries from multiple sources but in relatively restricted domains.

In the area of content-based visual query, we have developed effective and efficient techniques for searching unconstrained images based on selected visual features. Our Web image/video search and cataloging engine is among the first of its kind and demonstrates innovative ways of integrating multimedia features. We have built large-scale online prototypes for evaluation, and we are integrating our content-based search tools with a real-time Web video editing/browsing system which can be used as an interactive visual information system. We have also developed a platform for evaluating methods for measuring differences in Web documents.

As we move to the future, we will focus on enhancing access to live data, integrating multiple media, and incorporating feedback from various users. One of our next steps is to track new information from a variety of sources including live feeds. Although the tools that we have developed can facilitate access to live news sources, CDNS currently runs primarily over stored local collections. One of our current efforts is to incorporate the individual tools we have developed for the separate components into an integrated system that supports live feeds.

A primary focus in CDNS is the integration of multiple media at all levels of the system. In our initial research, we have developed an architecture that provides a model for interaction between components that summarize and track text and components that primarily search image data. This model provides a paradigm for user interaction with multiple media and illustrates how feedback between components enhances overall interaction with the system. We have begun integrating textual and image processing techniques within individual components of the system as well, as demonstrated by our use of both textual features provided by URLs and directory names and image features to constrain search over images. We are continuing in this direction by developing algorithms that utilize more sophisticated natural language processing techniques for identifying textual features with further image processing. We are also exploring methods for incorporating multimedia processing of video, using both image clues and textual clues from a transcript accompanying the video.

Given our focus on the domain of news, journalists are target users of CDNS. We are interacting with the faculty and students at the Graduate School of Journalism at Columbia University to collaborate both in content development and in developing technology that is tailored to journalists' needs. In an initial focus group, journalists provided feedback on current summarization and search techniques, noting that if a summary clearly attributes information to sources, identifies contradictions and agreements between underlying articles, and points out differences between

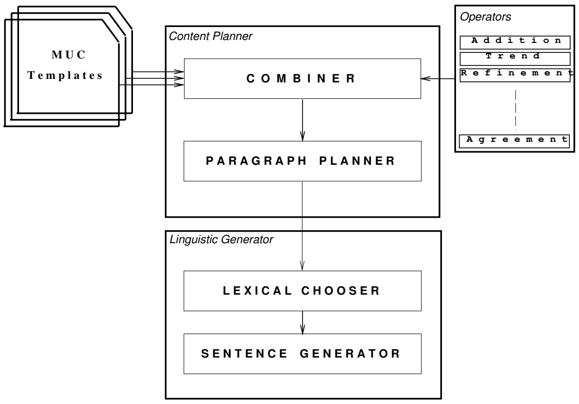

Figure 10: (a) SUMMONS System Architecture.

articles being summarized, this will aid the journalist in determining whether the original sources are important. Our future work includes planned user studies for formative and summative evaluation, and using the real-time video editing/browsing prototype described above as a testbed of an interactive visual information system. These interactions will help us tune our system and algorithms to this important domain of application.

References

[1] CIA. The CIA world factbook. URL: http://www.odci.gov/cia/publications/95fact, 1995.

[2] Ralph Grishman, Catherine Macleod, and John Sterling. New York University: Description of the PROTEUS system as used for MUC-4. In *Proceedings of the Fourth Message Understanding Conference*, June 1992.

[3] Dragomir R. Radev and Kathleen R. McKeown. Building a generation knowledge source using internet-accessible newswire. In *Proceedings of the 5th Conference on Applied Natural Language Processing*, Washington, DC, April 1997.

[4] New Mexico State University CRL. Products and prototypes. URL: http://www.nmsu.edu/offer.html, 1996.

[5] Interpix. Image Surfer. Technical Report http://www.interpix.com/.

[6] J. R. Smith and S.-F. Chang. Searching for images and videos on the World-Wide Web. submitted to IEEE multimedia magazine, also cu-ctr technical report 459-96-25, 1996. Demo accessible from URL `http://www.ctr.columbia.edu/webseek`.

[7] C. Frankel, M. Swain, and V. Athitsos. Web-Seer: An image search engine for the World Wide Web. Technical Report TR-96-14, University of Chicago, July 1996.

[8] D. Zhong, H. Zhang, and S.-F. Chang. Clustering methods for video browsing and annotation. In *IS&T/SPIE Symposium on Electronic Imaging: Science and Technology – Storage & Retrieval for Image and Video Databases IV*, San Jose, CA, February 1996.

[9] T. P. Minka and R. W. Picard. An image database browser that learns from user interaction. Technical Report 365, MIT Media Laboratory and Modeling Group Technical Report, 1996.

[10] John S. Justeson and Slava M. Katz. Technical terminology: some linguistic properties and an algorithm for identification in text. *Natural Language Engineering*, 1:9–27, 1995.

[11] Frank Smadja and Kathleen R. McKeown. Using collocations for language generation. *Computational Intelligence*, 7(4), December 1991.

[12] Frank Smadja. Retrieving collocations from text: Xtract. *Computational Linguistics*, 19(1):143–177, March 1993.

[13] Vasileios Hatzivassiloglou and Kathleen R. McKeown. Towards the automatic identification of adjectival scales: Clustering adjectives according to meaning. In *Proceedings of the 31st Annual Meeting of the ACL*, pages 172–182, Columbus, Ohio, June 1993. Association for Computational Linguistics.

[14] W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, and C. Faloutsos. The QBIC project: Querying images by content using color, texture, and shape. In *Storage and Retrieval for Image and Video Databases*, volume SPIE Vol. 1908, February 1993.

[15] J. R. Smith and S.-F. Chang. VisualSEEk: a fully automated content-based image query system. In *Proc. ACM Intern. Conf. Multimedia*, Boston, MA, May 1996. ACM. Demo accessible from URL `http://www.ctr.columbia.edu/VisualSEEk`.

[16] M. J. Swain and D. H. Ballard. Color indexing. *International Journal of Computer Vision*, 7:1 1991.

[17] F. Liu and R. W. Picard. Periodicity, directionality, and randomness: Wold features for image modeling and retrieval. Technical Report 320, MIT Media Laboratory and Modeling Group Technical Report, 1994.

[18] T. Chang and C.-C. Kuo. Texture analysis and classification with tree-structured wavelet transform. *IEEE Trans. Image Processing*, 3(4), October 1993.

[19] H. Samet. The quadtree and related hierarchical data structures. *ACM Computing Surveys*, 16(2):187 – 260, 1984.

[20] S.-K. Chang, Q. Y. Shi, and C. Y. Yan. Iconic indexing by 2-D strings. *IEEE Trans. Pattern Anal. Machine Intell.*, 9(3):413 – 428, May 1987.

[21] C. Faloutsos and K.-I. Lin. *FastMap*: A fast algorithm for indexing, data mining and visualization of traditional and multimedia datasets. In *ACM Proc. Int. Conf. Manag. Data (SIGMOD)*, pages 163 – 174, 1995.

[22] E. G. M. Petrakis and C. Faloutsos. Similarity searching in large image databases. Technical Report 3388, Department of Computer Science, University of Maryland, 1995.

[23] S. Chawathe, S. Rajaraman, H. Garcia-Molina, and Widom J. Change detection in hierarchically structured information. In *Proceedings ACM SIGMOD Symposium*. ACM, 1996.

[24] Lisa F. Rau, R. Brandow, and K. Mitze. Domain-independent summarization of news. In *Summarizing Text for Intelligent Communication*, pages 71–75, Dagstuhl, Germany, 1994.

[25] Julian M. Kupiec, Jan Pedersen, and Francine Chen. A trainable document summarizer. In Edward A. Fox, Peter Ingwersen, and Raya Fidel, editors, *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 68–73, Seattle, Washington, July 1995.

[26] Chris D. Paice and Paul A. Jones. The identification of important concepts in highly structured technical papers. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 69–78, 1993.

[27] Hans P. Luhn. The automatic creation of literature abstracts. *IBM Journal*, pages 159–165, 1958.

[28] Kathleen R. McKeown and Dragomir R. Radev. Generating summaries of multiple news articles. In Edward A. Fox, Peter Ingwersen, and Raya Fidel, editors, *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 74–82, Seattle, Washington, July 1995.

[29] Kathleen R. McKeown. *Text Generation: Using Discourse Strategies and Focus Constraints to Generate Natural Language Text*. Studies in Natural Language Processing. Cambridge University Press, Cambridge, England, 1985.

[30] Darrin Duford. CREP: a regular expression-matching textual corpus tool. Technical Report CUCS-005-93, Columbia University, 1993.

[31] Michael Elhadad. *Using Argumentation to Control Lexical Choice: A Functional Unification Implementation*. PhD thesis, Department of Computer Science, Columbia University, New York, 1993.

[32] James Shaw. Conciseness through aggregation in text generation. In *Proceedings of the 33rd Annual Meeting of the ACL (Student Session)*, pages 329–331, Cambridge, Massachusetts, June 1995. Association for Computational Linguistics.

[33] Kathleen McKeown, Jacques Robin, and Karen Kukich. Generating concise natural language summaries. *Information Processing and Management, Special Issue on Summarization*, 31(5):703–733, September 1995.

[34] Jacques Robin and Kathleen McKeown. Empirically designing and evaluating a new revision-based model for summary generation. *Artificial Intelligence*, 85, August 1996. Special Issue on Empirical Methods.

[35] Tim Finin, Rich Fritzson, Don McKay, and Robin McEntire. KQML — a language and protocol for knowledge and information exchange. Technical Report CS-94-02, Computer Science Department, University of Maryland and Valley Forge Engineering Center, Unisys Corporation, Computer Science Department, University of Maryland, UMBC, Baltimore, MD 21228, 1994. Accessible from `ftp://gopher.cs.umbc.edu/-pub/ARPA/kqml/papers/kbks.ps`.

[36] The Terrorist Profile Weekly. The TPW terrorist group index. URL: http://www.site.gmu.edu/~cdibona/-grpindex.html, 1996.

[37] Jacques Robin. *Revision-Based Generation of Natural Language Summaries Providing Historical Background*. PhD thesis, Department of Computer Science, Columbia University, New York, 1994.