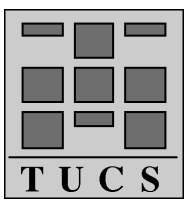
A prototype-matching system for scientific abstract collection semantic clustering

Antonina Kloptchenko
University of Turku, Department of Computer Science,
Lemminkäisenkatu 14, FIN-20520 Turku, Finland
Barbro Back
Åbo Akademi University, Department of Computer Science,
Lemminkäisenkatu 14, FIN-20520 Turku, Finland
Ari Visa
Tampere University of Technology, Department of
Information Technology, Tampere, Finland
Jarmo Toivonen
Tampere University of Technology, Department of
Information Technology, Tampere, Finland
Hannu Vanharanta
Pori School of Technology and Economics, Pori, Finland



Turku Centre for Computer Science
TUCS Technical Report No 465
December 2002
ISBN 952-12-1019-2
ISSN 1239-1891

Abstract

The growth of digitally available text information has created a need for effective text processing tools. Document clustering aims at solving some of the text processing problems, such as text categorization, topic discovery, text browsing and searching, retrieval by content and organizing retrieval results on the Web. We have used an information retrieval by content method built on prototype matching clustering of a scientific text collection, which in our case are the abstracts from The Hawaii International Conference on System Science 2001. Our aim is to retrieve the documents from a conference paper collection according to similarities in their contents and semantic structures. Our prototype-matching information retrieval method consists of document pre-processing, "smart" document encoding on different syntactic levels, clustering document histograms using a vector quantization algorithm, and matching those histograms for every document against a prototype. In the report, we position our methods among the existing document clustering methods, explain the motivation behind the clustering of scientific conference papers, and give an example of using our prototype tool for information retrieval by content on the scientific abstract collection. The method offers a promising alternative for task of information retrieval by content from scientific text collections.

Keywords: text clustering, information retrieval by content, scientific text collection

TUCS Laboratory
Data Mining and Knowledge Management Laboratory

1. Introduction

The Internet, digital libraries, data warehouses, and information organizations generate and carry far more available text information than it is possible for anyone to process manually (Aslam, et al. 1999). Text is the most common form of written communication that carries different meaning to different users. Quality, quantity and ambiguous structure of available text create a number of certain difficulties for working with it. Searching, organizing, browsing, analysing and retrieving valuable information from textual databases have become time consuming and costly procedures. The methods of knowledge discovery in textual databases also known as text mining (TM) methods strive to assist user information needs by accomplishing various text-associated tasks at less expense. TM aims at looking for patterns in text and can be defined, according to (Witten, et al. 1998), as the process of analysing text to extract previously unknown information that is useful for particular purposes. TM is a multidisciplinary field that includes information retrieval, text analysis, clustering, visualization, and categorization (Tan, 1999). One of the TM subpart is information retrieval (IR), which refers to a process of locating the subset of the documents that are deemed to be relevant to the query (Rijsbergen, 1979). On the one hand TM methods help readers to rediscover what the author had implied in texts, on the other hand, discover some valuable to the reader knowledge that an author did not explicitly stated. IR methods offer to retrieve information, which according to the user assumptions already exists in the data set and will satisfy his/her information needs.

Very often users of textual information face the hardships in anticipation of their own information needs. Users tend to articulate their information needs very vaguely and have poor motivation to construct smart queries for IR tools. It is easy to imagine situations where the user might not be fully acquainted with established terminology in a field, or not fully sure about the content of the documents he/she needs to retrieve. Most users, as was noticed in (Anick and Vaithyanathan, 1997), prefer to answer questions about the relevancy of information already presented to them by retrieval system, rather than to describe explicitly what they are looking for. This behaviour requires some sophisticated text analysis tools that could help users to deal with different text corpora. Text clustering helps users to deal with information overload in response to their information needs by offering effective text exploration techniques. Clustering methods help to organize text collection and contribute to various text analysis tasks, e.g. topic discovery, text searching, organizing retrieval results on the Web, and text retrieval by content.

During the last years, applied science has become more and more interdisciplinary. The task of how to sort out the papers submitted to a scientific conference in the proposed categories and tracks has turned out to be a nontrivial task. The taxonomy of scientific conferences has grown very complicated, due to the blurred borders of applied research fields. The authors and the readers of the scientific articles frequently represent the same semantics using different words or describe different meanings using words that have various meanings. This phenomenon is called word ambiguities in IR literature. Authors use similar keywords for identifying the content of the presented papers, which can belong to either the same or different tracks. On the one hand, even experienced readers, such as track chairmen, encounter certain difficulties with the determination of what particular track the paper belongs to. On the other hand,

a conference attendee who wants to read the papers similar to his/her research interests needs to browse the whole conference proceeding, or rely on the keyword search, considering keywords to be a reflection of the paper content.

In this report, we offer a prototype matching text-clustering system for retrieval by content. The prototype matching system is an IR system with embedded text mining capabilities, because it aims at retrieving relevant documents, which is by definition is a purpose of IR system, and at the same time, the retrieved document should be semantically similar to each other, which becomes a subtask of TM when the similarity is not stated explicitly. According to (Hand, 2001), text retrieval by content is one of the most important tasks in data mining from textual databases. Therefore, we classify our system as an IR system with TM capabilities. We illustrate the system using a scientific conference abstract collection from The Hawaii International Conference on System Science 2001. The system is based on document preprocessing, "smart" document encoding and collection clustering, and document retrieval phases. It aims to help the conference organizers and attendees to retrieve the papers from the conference proceeding based on their semantic content similarities. We suggest that the user take an abstract from an interesting conference paper, and use it as a prototype query.

The material presented in the remainder of the report is organized as follows. In Section 2, we review the related work in using clustering for information retrieval and other text processing purposes and explain what makes our approach different. In Section 3, we describe the prototype-matching methodology based on document encoding, creating histograms of documents on different syntactic levels, and matching and retrieving them. In Section 4, we provide our motivation and way to accomplish a task of the prototype matching clustering on a chosen scientific conference text collection. In Section 5, we give a brief description of our scientific abstract collection. Section 6 contains an exposition of the experiments we have conducted. In Section 7 we provide a discussion about the results. Finally, in Section 8, we provide some conclusions and suggestions for future work.

2. Background

Document clustering has been extensively explored for information retrieval and text mining domains for learning about text collections. Clustering techniques strive to create a subset from a collection of documents, so that a cluster represents a group of documents having features that are similar, compared to the features of other groups (Hand, 2001). Clustering does not require any predefined categories for grouping the documents (Jain, et al. 1999). The central assumption proposed by Van Rijsbergen in 1979, and known as Cluster Hypothesis, had made document clustering a powerful method for IR (van Rijsbergen, 1979). It states that a document relevant to a request is more likely to be similar to one another than to non-relevant documents. This hypothesis has received an experimental validation in the context Scatter/Gather system that uses document clustering as its primitive operation (Cutting, et al. 1992). Hierarchical, K-means and Relational Clustering are the most popular and known text clustering methods[1] (Karanikas, 2000).

[1] A good overview of clustering methods for IR is presented by Willet (Willett, P. (1988). "Recent Trends in Hierarchic Document Clustering: A Critical Review." Information Processing and Management **24**(5): 577-597.

Hierarchical clustering in form of agglomerative or divisive clustering often portrayed as the better quality clustering approaches, because they present textual information in intuitively understandable forms of hierarchies. Hierarchical clustering assumes a similarity function for determining the similarity of instances in a cluster. Agglomerative probabilistic clustering based on Generalizable Gaussian Mixture model was successfully tested for segmentation of e-mails by (Szymkowiak, 2001). Hierarchical document clustering using Ward's method based upon a series of nearest neighbour searches was addressed in (El-Hamdouchi and Willett, 1986). (Cutting, et al. 1992), (Schutze and Silverstein, 1997) studied the ways to improve clustering algorithms to make them computationally feasible in order to implement them in real-time.

(Steinbach, et al. 2000) argues that the quadratic time complexity of hierarchical algorithms makes them less appealing than k-means clustering, which has a time complexity linear in the number of documents. K-means is direct clustering method that uses specified number of clusters *k*, *centroids* as attributes of clusters' description, and assigned clustering evaluation function. The author shows that "bisecting" K-means technique produces results that are as good or better than tested hierarchical approaches.

IBM implemented hierarchical and binary relational clustering in Intelligent Miner for Text. The vocabulary analysis and determination of important pairs of terms can be archived by hierarchical clustering, and finding hidden in document topics and establishing relationship between them can be done using binary relationship clustering.

In addition to organizing text corpora for retrieval by content (Anick and Vaithyanathan, 1997), (Merkl and Schweighofer, 1997), text collection clustering helps to accomplish a number of other TM tasks, such as

a) topic discovery (Larsen, 1999; Zaiane, 1999),
b) completing automatic overviews (El-Hamdouchi and Willett, 1986),
c) browsing and searching (Cutting, et al. 1992),
d) organizing retrieval results (Lee and Yang, 1999), (Zamir and Etzioni, 1998),
e) text categorization (Aslam, et al. 1999),

(Larsen, 1999) discovered topic hierarchies and organized the search results by the topic similarity using the unsupervised clustering algorithm. The algorithm is tied to the feature extraction and the grouping of the points based on a proximity measure in a feature space. (El-Hamdouchi and Willett, 1986) used a tree-like document conceptual clustering to complete a quick overview of a large financial news collection, by characterization of document groups. Clustering for browsing and searching was done in (Cutting, et al. 1992), using a technique that supports an iterative browsing interface by dynamically scattering a document collection into smaller clusters. The user then selects and gathers relevant groups among the clusters to group these results again. Herein, the user navigates the document search space. In (Lee and Yang, 1999), a SOM-based clustering method based on word co-occurrences was presented for retrieval on a Chinese corpus from the web. Clustering for organizing the retrieval results on the Web using snippets, not a full text, was studied in (Zamir and Etzioni, 1998). Text categorization according to natural topic structure using dense subgraph structure and star algorithm was accomplished by (Aslam, et al. 1999) classified Business letters into corresponding types by extracting and weigthening of index terms, employing language frequency statistics and morphological knowledge. (Anick and Vaithyanathan, 1997)) studied a document clustering approach for retrieval by content. The main point of this approach was to exploit clustering and paraphrases of term occurrence. (Merkl and

Schweighofer, 1997) used another clustering approach for retrieving by content and organizing legal text corpora. It was based on SOM as a clustering mechanism, and aimed at the detection of similarities between documents. WebSom system is based on SOM clustering and allows browsing and retrieving the resulted matching list to perform multi-level search of text collection with increasing navigating role of a user (Kohonen, 1998). In a majority of algorithms mentioned above, the user participates actively in the whole clustering process, controlling the fulfilment of his/her information needs.

There are a number of primary challenges in textual data clustering for retrieval by content, such as effective representation of text, the determination of similarity, and the high dimensionality of document collections. The effective solutions for those challenges are discussed in (Schutze and Silverstein, 1997), (Salton and McGill, 1983), and (Anick and Vaithyanathan, 1997).

There are a number of another approaches to organize scientific text collection for retrieval by content that are based on indexing. (Lawrence, et al. 1999) attempted to create a digital library of scientific literature on the web, that will include efficient location of articles, full-text indexing of the articles, autonomous citation indexing, information extraction, similar document detection, user profiling and more. The full-text indexing, with analogy to popular scientific CiteSeer website, was built as a usual hash table of words (inverted index) including stop-words.

We designed our prototype-matching clustering system for a purpose of text retrieval by content that can operate without any specifically known morphological, lexical knowledge, predefined or chosen indexes. It differs from the methods mentioned above because it does not focus on word co-occurrences (Lee and Yang, 1999), or on feature extraction (Larsen, 1999), and does not create a high dimensional vector space to represent the whole collection (Cutting, et al. 1992). It takes into consideration that sentence and paragraph structure, and word order carry just as much important semantic information to a reader as word appearances.

3. Methodology

Currently, the prototype-matching clustering methodology for text analysis on different syntactic levels consists of the phases described below.

3.1 Document collection pre-processing and encoding

a. Pre-processing takes place before text documents are presented to the text clustering system. We do a basic filtering so that every sentence occupies its own line. Compiling the abbreviation file performs synonym and compound word filtering. We round numbers, separate punctuation marks by spaces, and exclude extra carriage returns, mathematical signs, and dashes. We do not remove stop words to keep our method language independent.
b. After basic filtering of the text, we encode the document. A word *w* is transformed into a number according to the following formula:

$$y = \sum_{i=0}^{L-1} k^i \times c_{L-i} \qquad (1)$$

where L is the length of the word character string, c_i is the ACSII value of a character within a word *w* and k is a constant. Every word and single punctuation mark in the documents is encoded to an individual feature word vector in the files corresponding to every document. Each word is analyzed character by character, so

that a key entry in a code table is calculated. This approach is accurate and sustainable for statistical analysis, although it is sensitive to capital letters and conjugations.

3.2 Document processing and matching

We have used a text clustering methodology and a vector quantization algorithm for document processing and matching (Visa, et al. 2000; Toivonen, et al. 2001). The document-processing phase of a prototype-matching clustering methodology consists of the following steps:

a. We set the minimal and maximal values (a and b) for the word codes, and look at their distribution (a set of word code numbers from 3.1b) for the entire document collection. In the training phase, we divide the range between the minimal and maximal values of words' code numbers into N_w logarithmically equal bins. First, we calculate the frequency of words belonging to each bin. We normalize the bins' counts according to the quantity of all words in the text. For estimation of the word codes' distribution, we chose the Weibull distribution. The Weibull distribution - one of the most widely used lifetime distributions in reliability engineering[2] - is a versatile distribution that can take on the characteristics of other types of distributions based on the value of the shape parameter. A number of Weibull distributions is calculated with various possible values for a and b using a selected precision. The best fitting Weibull distribution is to be compared with the code distribution in a sense of the smallest square sum by calculating the Cumulative Distribution Function according to:

$$CDF = 1 - e^{(((-2.6\times\log(y/y_{\max}))^b)\times a)} \quad (2)$$

where a and b are the parameters to be adjusted in Weibull distribution. The size of every bin is $1/N_w$.

Hereby, we have created a common word histogram for the entire document collection. Every word in it belongs to a bin that can be found using the code number and the parameters of the best fitting Weibull distribution. The quantization is the best where the words are the most typical to a text (usually 2-5 symbol length words). The encoding algorithm produces a unique number for each word and only the same word can get an equal number.

b. Similarly to the word level, we convert every sentence into a number on the sentence level. First, every word in a sentence is changed to a bin number (bn_i) in the same way as we did for words. The whole sentence is considered as a sampled signal. Since the sentences in the text contain different numbers of words, the sentence vector's lengths vary. To overcome this fact we apply Discrete Fourier Transformation (DFT) to every sentence vector in a collection. In the transformation we do not consider all of the coefficients, however, we transform bn_i = bin number of the word i into output coefficients from B_0 to B_n to create a cumulative distribution like the one on the word level. The range between the minimal and maximal values of the sentence code numbers is divided into N_s equally sized bins. We calculate the frequency of sentences belonging to each bin. Then we divide the bins' counts with the total number of

[2] http://www.weibull.com (1998)

sentences in a collection. Finally, we find the best Weibull distribution corresponding to both cumulative distributions. A graphical representation of a sentence quantization process is given in (Toivonen, et al. 2001).

c. Furthermore, we examine every document in a collection by creating the histograms of the documents' word and sentence code numbers (levels), according to the corresponding values of quantization. We encode the filtered document from a collection word by word on the word level. Each word code number is quantified using word quantization created with all the words in the database. The histogram consists of N_w bins and is normalized by the total number of words in the document. We create similar histograms for every document in the database for the sentence level.
d. We convert the paragraphs of the documents into vectors using the code numbers of the sentences. The vectors are Fourier transformed as well, and the coefficient B_i represents the paragraph. We find the best Weibull distribution corresponding to the paragraph data and do the paragraph quantization.
e. We examine every document in a collection by creating the histograms of the documents' word, sentence and paragraph code numbers (levels), according to the corresponding value of quantization. On the word level the filtered text from the document is encoded word by word. Each word code number is quantified using word quantization created with all the words in the database. The histogram consists of N_w bins and is normalized by the total number of words in the document. The aim is to create similar histograms for every semantically similar document in the database for the sentence and paragraph levels.

3.3 Document retrieval

Using the histograms of all the documents in the collection, we analyse the single documents' text on the word, sentence and paragraph levels. Although theoretically we can compare the histograms using any distance measures, Euclidian is proved to be the best choice in document retrieval phase. The closest documents in terms of the smallest Euclidian distance between them form a cluster. To complete the retrieval part we choose the documents with the smallest distances to the prototype. The system creates a distance proximity table of all distances among the documents in a collection. We retrieve the documents from the top of proximity table to every prototype document presented to the system.

4. Description of Task

As it was mentioned earlier, one of the distinct features of the conferences on applied science is cross-topic and interdisciplinary research. This feature creates certain obstacles within decision-making concerning what track a particular paper belongs to. Authors, conference organizers and attendees can experience difficulties in the conference setting with choosing an appropriate track to submit or assign a paper to, or to attend.

The conference organizers have noticed some similarities in the submitted papers that run across the traditional conference track division. To save the efforts of the experts to process submitted papers manually looking for links and the common topics, the conference organizers often rely on an authors' presentation of the keywords as the reflection of the main topic of a paper or on the track leaders who decide whether a

particular paper is relevant to a track stream. Both approaches risk the occurrence of bounded rationality, which can lower the paper classification and decrease the conference attendees' satisfaction.

We offer our user the opportunity to determine the content of the scientific paper more objectively. As stated earlier, the user of the system can save the efforts and time on constructing a smart query, using instead entire interesting abstract. The system aims at retrieving the documents that contain the same meaning from a document collection. A prototype-matching system is a simple content-based information retrieval system. The system is able to retrieve the documents that contain the same meaning from the entire data collection. The system includes all three key components of an information retrieval system (van Rijsbergen, 1979):

1. Query presentation - a part of the scientific article, representing the users information needs;
2. Document representation - smartly constructed sentence level histograms for representing a text collection;
3. Matching function - the Euclidian distance measure between sentence level histograms for ranking the documents according to their similarity.

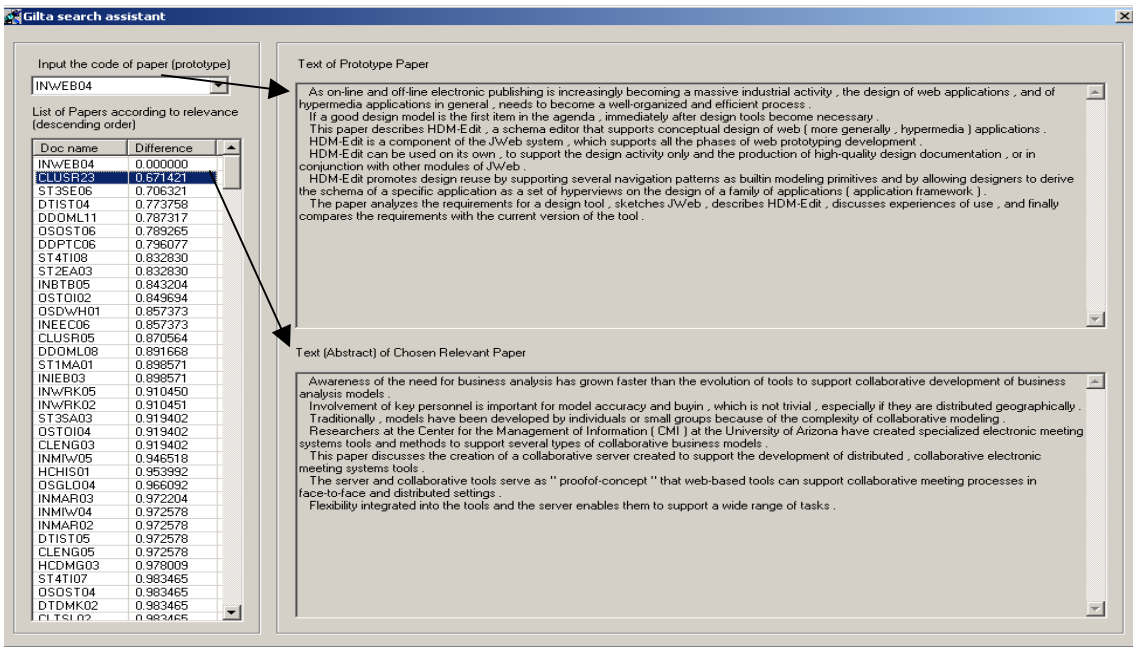


Figure 1. User Interface.

The interface of our running prototype based on unfiltered abstracts is depicted in Figure 1, using which he/she can retrieve the papers that are semantically similar to the interesting abstract. We present the conference codes of the submitted papers as a pull-down list in the upper left panel and the text abstracts in the right panel. So that the text of a chosen prototype-abstract (e.g. "Supporting Reusable Web Design with HDM-Edit" with a corresponding conference code INWEB04) is on the upper panel and a text of a chosen abstract-match (e.g. "Experiences with Collaborative Applications that Support Distributed Modeling" with a corresponding conference code CLUSR23) is on the lower panel from the top of the distance proximity table of the prototype. The top of distance proximity table is situated in the left panel and contains the list of codes of the abstracts that are semantically close to a prototype abstract.

5. Description of Data Collection

We have chosen the scientific abstracts from the entire HICSS-34 conference proceeding database for our pilot study because abstracts are designed to project research for the public eyes by offering a preliminary overview of the research in brief form (dos Santos 1996).

HICSS 34 is a general-purpose conference that has served a computer society for over three decades. HICSS addresses a wide range of issues from computer science, computer engineering, and information systems. The objective of HICSS is to provide a unique environment in which researchers, academicians and practitioners in the information, computer and system sciences can exchange ideas, techniques and applications (Sprague 2001). Thus HICSS organizing committee tries to schedule all the sessions carefully to create a high degree of interaction and discussion among the conference participants to establish a workshop-like setting at the conference. The scientific papers at HICSS-34 were arranged into nine major tracks, which were further divided into seventy-eight minitracks. The organizers made an effort to identify six themes that run across the tracks based on the similarities and expansion of the scientific fields besides the traditional track division. This particular distribution of papers into non-traditional themes made a HICSS-34 conference proceeding an interesting data collection for our investigation. Table 1 presents the taxonomy of the HICSS-34 conference, where the outlined six *cross-track themes* are listed on the right hand side. The themes cover 168 papers in the conference from thirty different minitracks.

№	Track Title /№ of papers/№ of Minitracks	№	Theme Title /Number of papers
1	Collaboration Systems and Technology /66 /9	1	Knowledge Management/20
2	Complex Systems /29 /5	2	Data Warehousing-Data Mining/24
3	Decision Technologies for Management /47 /7	3	Collaborative Learning/22
4	Digital Documents /40 /6	4	Workflow/12
5	Emerging Technology /30 /4	5	E-commerce Development/54
6	Information Technology in Health Care /26 /5	6	E-commerce Application/36
7	Internet and Digital Economy /68 /12		
8	Organizational Systems and Technology /63 /14		
9	Software Technology /75 /13		

Table1. Taxonomy of Tracks and Themes of HICSS-34.

6. Experiments

We conducted several separate experiments to test the ability to retrieve the most similar in meaning of the proposed prototype-matching system on the scientific conference corpus. The most significant ones are presented here. In our experiments, we have used the methodology described in Section 2. We left out the paragraph level analysis because the abstracts as the short informative-consistent representation of the scientific papers often consist of only one paragraph. From every abstract we have omitted the abstract titles, and author listing as irrelevant and keywords as redundant information for our system. We tried different sizes of recall window. We did not consider an order within a recall window, only paper co-occurrence.

The first experiment was on the larger text collection - the entire abstract collection from the conference proceeding. We examined the system's ability to retrieve the most similar abstracts from the entire conference abstract collection using any

chosen abstract as a prototype query to cluster the collection. We inspected the abstracts from the top of the proximity table for every prototype-abstract. Because conference tracks are meant to unite the papers from the same research filed, the majority of the closest matches to every prototype should be from the same track in the first experiment.

The second experiment was with a slightly different scope – to analyze the consistency of the cross-track themes proposed by the conference organizers. Because themes are supposed to unite the most semantically close papers from different track we have expected that among the closest matches to and abstracts from a certain theme would appear abstracts of the papers from the same theme and different tracks.

6.1 Experimental Settings

6.1.1 Creating the word level histograms

After abstract filtering every word from every abstract that previously was in a form a string of ACSII characters was converted into a number according to Formula (1) from Section 3.1, so that the unique number corresponds to the unique word. After coding we composed a common text containing all words in their numeric forms. In the teaching phase, the range between the minimum and maximum word values was divided into logarithmically equal bins to calculate the count of the words belonging to every bin. A set of the cumulative Weibull distributions restricted by the suitable minimum and maximum values was computed to find the best fitting Weibull distribution and to divide it into the number of areas that is equal to the number of the words in the common text in the testing phase. Hereupon, we created a common word histogram for the entire text collection. Every word was assigned to a bin that is found using the code number and best fitting Weibull curve. The unique number calculated for every word has to appear somewhere in the common word histogram, which consists of 2080 bins. We created 444 individual word histograms based on the parameters from a Weibull-distributed common word histogram. From now on, we could match abstract word histograms against each other simply by calculating the Euclidian distances between them. Thereby, we established the similarities between the abstracts on the word level.

As an illustration of word level analysis, we have chosen the abstract from the paper "Supporting Reusable Web Design with HDM-Edit" (with the conference code INWEB04). After text preprocessing, we have coded the words from it, according to formula (1) from the methodology section, e.g. for word *design* the result is 6472:

$$y=k^5\,acsii\,(\mathbf{d}) + k^4\,acsii\,(\mathbf{e}) + k^3 acsii\,(\mathbf{s}) + k^2 acsii\,(\mathbf{i}) + k\,acsii\,(\mathbf{g}) + acsii\,(\mathbf{n})$$

Words	Word Codes
The	645
paper	6770
system	7311
method	3332
in	320
design	6472

Figure 2. Word codes (k=2).

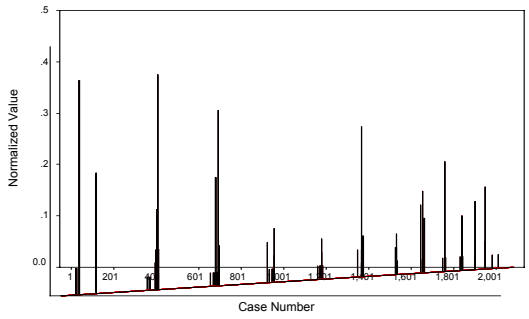
Some examples of the word codes form the filtered abstract to paper INWEB 04 is presented in Figure 2, for $k=2$.

After Weibull-curve quantization of a common word histogram, we created the word histogram for INWEB04, a fragment of its word digital array after normalization is presented in Figure 3 with 73 nonempty bins in it. The minimum value of bin 0.0268221 in the normalized digital array of INWEB04 word histogram corresponds to one hit in a common word histogram. After simple calculation we have established that 210 out of 218 words in INWEB04 match the common word histogram making 78 hits, e.g. a comma sign has

appeared 16 times, and was classified into one bin with value of 0.429153. Because different words are clustered into different ranges in the quantization, and some of them occur several times in the same range, the histogram in Figure 3 consists of higher and lower bars. The Euclidian distances between the histograms of some other words and the INWEB04 word histogram shows the similarity in vocabulary used in INWEB04 and other abstracts.

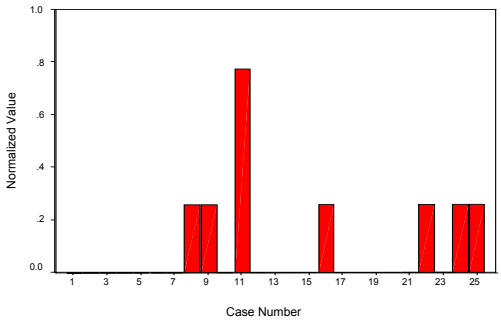
6.1. 2 Creating the sentence level histograms

The further analysis was carried out on the sentence level in a belief that sentence structure carries more sophisticated semantic meaning than word usage. Converting every sentence into a number using the word bin numbers from the previous phase we built up a file where a unique number was assigned to every sentence. The encoded sentences were altered according to Fourier transformation to create a cumulative distribution of the sentences from the whole data set. We divided the range between the minimum and maximum values of sentence codes into the numbers of the bins that is equal to the number of sentences in our common text to be able to pick the matching Weibull distribution to use in sentence quantization. Using the parameters from the Weibull distribution, we built sentence histograms of the size of 25 for every abstract in our scientific text corpus. The closest sentence histograms in terms of the smallest Euclidian distances between them illustrate the semantic similarities between the corresponding abstracts.



0 0 0 0 0 0 0 0 0.0268221 0 0 0 0 0 0 0 0 0 0 0 0.0268221 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0.0536442 0.0804663 0 0 0 0 0 0 0 0 0 0 0 0 0.160933 0 0 0 0 0

Figure 3. INWEB 04 Word Level Histogram (Bar histogram and a fragment of a digital normalized array).



0 0 0 0 0 0 0 0 0.258199 0.258199 0 0.774597 0 0 0 0 0.2581990 0 0 0 0 0.258199 0 0.258199 0.258199

Figure 4. INWEB 04 Sentence Level Histogram (Bar histogram and a digital a digital normalized array).

As an illustration of the sentence level analysis we present in Figure 4 the resulted sentence level histogram for INWEB04. There are 9 sentences in the INWEB04 abstract; no less than 4 of them match the common sentence histogram, because 0.258199 represents one hit and 0.774597 represents 3 hits on a common sentence histogram. Because similar sentence structures are clustered into the same range in a quantization, we obtain higher and lower bars in the sentence histogram. Figure 4 contains the normalized sentence histogram for INWEB 04, and a corresponding to it normalized digital array. Calculating Euclidian distances between INWEB 04 sentence

histogram and other sentence histograms in the data set shows the semantic similarities between INWEB 04 and other abstracts.

INWEB04	0
CLUSR23	0.671421
ST3SE06	0.706321
DTIST04	0.773758
DDOML11	0.787317
OSOST06	0.789265
DDPTC06	0.796077
ST2EA03	0.83283
ST4TI08	0.83283
INBTB05	0.843204
OSTOI02	0.849694
INEEC06	0.857373
OSDWH01	0.857373
CLUSR05	0.870564
DDOML08	0.891668
INIEB03	0.898571
ST1MA01	0.898571
INWRK05	0.91045
INWRK02	0.910451

Table 2. A fragment of the proximity table for INWEB04
Recall window = 47.

6.3 Conducing the experiments

In the first experiment, we studied every abstract from the conference collection after accomplishing all the procedures described above in order to allocate all the closest matches from the conference collection to the abstracts. In other words, any chosen abstract was used as a prototype query in attempt to retrieve the abstracts of papers that are the most semantically similar to a prototype from the collection. We expected the retrieval results to be from the same tracks, since tracks are the subsets of thematically similar research papers. We report our results for the recall window 47, which is equal to the average number of papers in the tracks.

The second experiment focus was on the 168 abstracts arranged by the conference organizers into the cross-track themes and their closest matches expecting them to be from the same cross-track theme. We performed clustering by calculating the Euclidian distance between the sentence histograms of an abstract-prototype and other abstracts, concentrating our attention on the abstract appearance in our clusters and in conference theme or track division. We set the recall window at 25, which is equal to the average numbers of papers in the cross-track themes.

6.4 Results

6.4.1 Results from the First ("Track") Experiment

The results obtained from our system along with a line of our reasoning can be explained from an example of the paper "Supporting Reusable Web Design with HDM-Edit" (INWEB 04) from "Web Engineering" minitrack in "Internet and the Digital Economy" track (IN - the first letter in the code of the papers from the track). The paper analyzes the requirements and a design of a web-publishing tool. It sketches and describes HDM-editor, discusses the experiences of it use, and finally compares the requirements of the current version of the tool. The text of an abstract from INWEB 04 is presented in top window of Figure 1.

The conference organizers had classified INWEB 04 into "Web Engineering" minitrack from "Internet and Digital Economy" track. Table 2 contains a fragment of a proximity table for INWEB 04 with the distances between our prototype and the abstracts that are similar to it. The left column contains the codes of the papers that are the first 18 matches out of 443 possible ones in a recall window 47. The right column contains the distances. From the INWEB04 proximity table we have learned, that from 47 papers in a recall window our system classified 13 papers from the same track as closest matches to INWEB 04 on the sentence level, and 11 papers on the word level. We used the white bold font on a gray background to outline the papers that belong to the same track as INWEB04. Table 3 represents the taxonomy of the track to which INWEB 04 belongs.

Name of Minitrack	Paper codes	Number of papers
Communities in the Digital economy: Concepts, Models and Platforms	(INCDE01–08)	8
E-commerce in the Finance Industry	(INFIN01-04)	3
Business to Business Electronic Commerce	(INBTB01-06)	6
E-commerce Customer Relationship Management	(INCRM01-06)	6
Infrastructure for E-business on the Internet	(INIEB01-06)	6
Virtual and Knowledge-Based Organizations	(INVKO02-03)	2
Economics and E-commerce	(INEEC01-06)	6
Marketing and E-commerce	(INMAR01-06)	6
Internet and Workflow Automation: Technical and Managerial Issues	(INWRK01-06)	6
Web Engineering	(INWEB01-09)	9
E-commerce Systems Development Methodologies	(INSDM01-03)	3
Managing Information on the Web	(INMIW01-07)	7

Table 3. The Taxonomy of Internet and Digital Economy Track

After careful reading of every abstract from the top of a distance proximity table we noticed, that the first nearest abstracts to INWEB04 discuss the problems related to collaboration support tools for web-based cooperation ("Experiences with Collaborative Applications that Support Distributed Modeling" (CLUSR23) from Collaboration Systems and Technology Track), coordination of shared software space ("Lost and Found Software Space" (ST3SE06) from the Software Engineering Tools Track). Those papers coincide with some of the ideas from INWEB04, such as a need for a support tool, its development, design and reuse. The closest matches are from the different fields of management information systems, namely software engineering (ST3SE06), groupware (CLUSR23) and business modeling ("Operations Centers for Logistics: General Concepts and the Deutsche Post Case" (DTIST04)), but they address the same problems of collaboration and tool reuse, either in software design or organizational structures.

Name of Minitrack	Codes	Number of papers in it
Market/ Economics	(CSMAE01–10)	10
Information Management	(CSIMG01-04)	4
Security, Reliability and Control	(CSSAR01-08)	7
Hybrid Dynamic Systems	(CSHDS01-03)	3
Self Organized Criticality	(CSSOC01-05)	5

Table 4.The Taxonomy of Complex System Track

After that we have looked at Complex System Track (CS – the first letter in the code of the papers from the track) - the track with the smallest number of papers in it. We present the taxonomy of the track in Table 4. We reason as follows, if paper A is

close in meaning to paper C, and paper B is close to the same paper C, then paper A and B are semantically close, we induced the sustainability of our retrieval results.

Table 5 contains a fragment of a proximity table for 5 papers: Impact of Renewable "Distributed Generation on Power Systems" (CSSAR01), "Multi-Area Probabilistic Reliability Assessment" (CSSAR02), "Min-max Transfer Capability: A New Concept" (CSSAR04), "Network Control as a Distributed, Dynamic Game" (CSSAR05), "Power System State Estimation: Modeling Error Effects and Impact on System Operation" (CSSAR06). All of them belong to "Security, Reliability and Control" minitrack of "Complex Systems" track (CSSAR01-06).

After a detailed inspection of the distance proximity table for those papers, we discovered that some of the papers, being from different tracks, have a tendency to fire as the closest matches to the papers from this minitrack. For instance, the paper "Empirical Norms as a Lever for On-line Support of General Practice" (HCDMG08) being from "Information Technology in Health Care" track discusses problems of complex system model building, its sustainability and usage. Mentioned above issues are semantically similar to the problems addressed in CSSAR01-08 papers. We highlighted the cross-referring papers by italic underlined font in Table 4. Using light gray background we outlined one specific example: the papers "Collective Memory Support in Negotiation: A Theoretical Framework" (CLNSS05) and "Multi-level Web Surfing" (ETWFW05) that make the semantic similarity between CSSAR05 and CSSAR06 stronger. We highlighted the papers from the same "Complex Systems" track by white bold font on dark gray background. We reasoned similarly for analyzing the retrieval by content results for every track.

CSSAR01	CSSAR02	CSSAR04	CSSAR05	CSSAR06
DDUAC06	OSKBE03	*DTUML06*	*ST3DS03*	DDTEC02
HCIST03	OSCIS01	HCTMD04	*CLUSR04*	*HCDMG08*
ST3SE03	CLUSR09	HCTMD05	INMIW05	OSSCI01
ST2EA04	*DTABS01*	ST2CP03	*OSOST09*	*CLALN02*
CLUSR16	DTIST02	DDOML06	*OSPMT06*	CSMAE02
DTMKI05	ST3SE02	DTDMK01	*ST2EA04*	INCRM04
CSMAE02	CLUSR19	INBTB04	CLALN05	*CLNSS05*
INIEB04	HCDMG01	DTABS03	*HCDMG08*	*ETWFW05*
CSIMG04	ST1MA02	HCTMD01	ST2CP04	CLUSR11
DDPTC08	ST3SA01	*INCRM04*	*DTUML06*	*DTIST01*
OSINF05	OSTTA07	*DTABS04*	ST2EA05	*CLNGL01*
ST4TI05	CSHDS02	CLDGS02	*ETSIT06*	*ST2WS01*
CLUSR02	INCRM03	CLENG01	CSSAR06	HCHIS01
INCRM05	ST1QS02	INCDE06	*CLNSS05*	CLTSL03
ST2CP01	ST3SE01	INMAR04	*ETWFW05*	DDOML12
ST4NI03	HCDAM03	INMIW07	*CLALN02*	*DTUML06*
CLENG02	CLUSR23	CSIMG01	CSSAR04	ST2IM01
CLUSR08	OSETH03	*DDUAC04*	OSINF04	ST3SA06
ST2WS01	CSMAE07	*OSINF05*	CLUSR12	*ST2EA04*
ST3SA02	ETWFW03	ST4TI06	*INEEC03*	CSSAR08
DTDMK04	OSTOI05	*ETSIT06*	INWEB05	CSSAR05
INWEB01	ST1MA04	*INMIW05*	*OSPMT04*	INCDE04
OSRMA02	ST2IM05	*OSPMT06*	ST3SE04	*INEEC03*
ST2CP07	OSMTO03	ST3SA04	CSIMG01	CSHDS03
DTDMK02	*OSOST09*	CSSAR05	CSIMG04	*ST3DS03*
DTABS01	CLUSR13	ST2WS05	CSSOC03	*DTABS04*
HCDMG08	OSPMT04	CLNGL01	*DDUAC04*	*INIEB04*

Table 5. A Fragment of a Proximity Table for 5 papers from "Complex Systems" Track (CSSAR01-06).

Table 5 contains a fragment of a proximity table for 5 papers: Impact of Renewable "Distributed Generation on Power Systems" (CSSAR01), "Multi-Area Probabilistic Reliability Assessment" (CSSAR02), "Min-max Transfer Capability: A New Concept" (CSSAR04), "Network Control as a Distributed, Dynamic Game"

(CSSAR05), "Power System State Estimation: Modeling Error Effects and Impact on System Operation" (CSSAR06). All of them belong to "Security, Reliability and Control" minitrack of "Complex Systems" track with codes respectively CSSAR01 to CSSAR 08.

Table 6 contains hit ratios per track (hit ratio1 and hit ratio 2), that reflect how many abstracts from the same track were retrieved among 47 or 25 closest matches respectively on the sentence level. We believe that sentence level convey more semantics than word usage. Recall window 25 was chosen to make the hit ratio values from track and theme experiment comparable.

№	Track Title	Number of papers	Hit ratio 1 (recall window 47)	Hit ratio 2 (recall window 25)
1	Collaboration Systems and Technology	66	25.8%	18.2%
2	Complex Systems	29	27.6%	17.2%
3	Decision Technologies for Management	47	25.5%	19.1%
4	Digital Documents	40	25%	15%
5	Emerging Technology	30	30%	20%
6	Information Technology in Health Care	26	23.1%	19.2%
7	Internet and Digital Economy	68	23.5%	14.7%
8	Organizational Systems and Technology	63	22.2%	15.8%
9	Software Technology	75	21.3%	16%

Table 6. The results from "Track" experiment.

6.4.2 Results from the Second ("Theme") Experiment

The conference organizers had put the paper "Supporting Reusable Web Design with HDM-Edit" (INWEB 04) into the largest cross-track Theme "E-commerce Development". The theme unites the abstracts from three tracks: "Software Technology", "Emerging technologies" and "Internet and Digital Economy", divided into a total number of nine minitracks. Table 7 contains the taxonomy of this cross-track theme.

Name of the Track	Name of the Minitrack within a Theme (code of papers within it)
Internet and Digital Economy	Managing Information on the Web (INMIW01-07)
	Infrastructure for E-business on the Internet (INIEB01-06)
	Web Engineering (INWEB01-09)
	E-commerce Systems Development Methodologies (INSDM01-03)
Emerging Technologies	Waiting for the Web (ETWFW01-07)
Software Technology	Mobile-Commerce: A New Frontier for E-business (ST1MC01-05)
	Novel Information Systems for Business to Business Electronic Commerce (ST4NI01-05)
	Trading Intangible Goods (ST4TI01-08)
	Quality of Service in Web Services (ST2WS01-05)

Table 7. Taxonomy of E-commerce Development Theme

We carefully read every abstract from a top of a distance proximity table once again. The closest matches are from the different fields of management information systems, namely, business modeling (DTIST04), groupware (CLUSR23) and software engineering (ST3SE06), but they address the same problems of collaboration and tool reuse, either in software design or organizational structures. Notably, the paper coded ST4TI08 (form the Software Technology Track) being from the same cross-track theme as INWEB04 discusses regulatory and fiscal aspects of electronic goods. This establishes more ambiguous similarity between INWEB04 and ST4TI08 than between INWEB04 and its first closest matches. It makes us believe that our clustering results are robust even though they are different from cross-track theme division.

Name of the Track	**Name of the Minitrack within a Theme (code of papers within it)**
Information Technology in Health Care Track	Health Care Data Management (HCDAM01-06)
	Data Mining for health Care Quality, Efficiency, and Practice Support (HCDMG01-08)
Decision Technologies for Management Track	Data Mining, Knowledge Discovery, and Information Retrieval (DTDMK01-07)
Organizational Systems and Technology Track	Data warehousing (OSDWH01-03)

Table 8. Taxonomy of Data Warehousing/Data Mining Theme.

Reading the papers from the smallest in number of papers cross-track theme "Data Warehousing and Data Mining" after the second experiment, and analyzing them on a semantic level has revealed that 26% of papers that fire among the closest ones to the papers from the data mining theme discuss the data/text mining methods applicability and some theoretical problems. We present the taxonomy of the cross-track theme that unites papers from three tracks: "Organizational System and Technology Track", "Information Technology Track" and " Decision Technologies for Management Track" in Table 8. The results have showed that some papers from the theme are further down on the proximity table than abstracts from papers belonging to different theme, than theme 2. Although those 26% of papers were not included in this theme, they form a stable cluster of papers on the same theme. As another observation, we have noticed

The Codes from Some Papers from Theme 2				
DTESM01	**DTESM02**	**OSDWH03**	**DTDMK05**	**DTDMK06**
ST3DS02	*INWEB05*	*ST2CP05*	OSSCI04	ETWFW07
HCDMG07	ST4TI04	*INIEB06*	*ST2CP05*	HCDMG07
ST3SA02	*CLGSS02*	*CSMAE09*	HCTMD06	ST4NI02
DDUAC06	*OSPMT05*	INSDM01	ST4TI02	*CSSOC01*
DTDMK06	DTUML09	DTUML02	*INSDM01*	CLALN06
DDPTC10	*CSSOC01*	DTDMK05	ETNON02	*CLCDV08*
ST4NI05	OSSCI01	DTESM01	*OSOST08*	ETWFW01
CLUSR01	DTMKI04	*INWEB05*	OSTTA07	DTESM01
DDUAC08	*OSOST07*	*INWRK03*	*CLDGS03*	OSTTA05
OSDWH03	CLUSR15	ST2CP06	*ST2EA02*	CLUSR20
OSKBE01	OSERP02	*CLCDV08*	CLNSS03	DDVUE02
OSTTA05	ST2EA07	ETNON13	DDVUE06	*ETNON03*
INIEB06	CLALN02	ST3SV03	OSDWH03	DTMKI03
CLCDV08	CLUSR04	DDUAC06	OSKBE01	ETEGV06

15 Closest Matches

Table 9. A Fragment from a Proximity Table for 5 papers from "Data Warehousing and Data Mining" theme.

that some abstracts fire as the closest match to only one abstract in the Data Warehousing and Data Mining theme. As a tendency, this pointed to lower relevancy of data mining papers to this theme, because majority of the closest matches to this cross-track theme had fired repeatedly as the closest ones to several abstracts in the theme. The results from the second experiment for the Data Warehousing and Data Mining theme are depicted in Table 9. We present a fragment of a proximity table for 5 abstracts from the "Data Warehousing and Data Mining" and 15 closest matches to them. Using white bold font on the light gray background we highlighted the abstracts from the theme. White font on very dark gray backgrounds shows those 26% of papers that fired repeatedly and did not belong to the minitracks from the theme, but belonged to another minitracks from the tracks that form "Data Warehousing and Data Mining" theme. Using italic underlined font we highlighted the papers that fired several times in the presented fragment of a proximity table. For instance, paper with code CLCDV08 has fired as the closest match to DTDMK05, DTDMK06, and DTESM01 in our fragment of proximity table, thus CLCDV08 forms the cluster with DTDMK05, DTDMK06, and DTESM01.

Another types of results from the second experiment are presented in Table 10. We answer the question how many papers within a certain theme (their names and sizes are presented in the left columns) have fired as the closest matches to the papers from the same theme on the sentence levels. The hit ratio values have showed so that, for example, our clustering method and the conference organizers clustered in the same theme 22.2% of papers from E-commerce development cross-track theme with a recall window 47 and 18.5% with a recall window 25.

№	Theme Title	Number of Papers	Hit ratio 3 (recall window 47)	Hit ratio 4 (recall window 25)
1	Knowledge Management	20	20%	20%
2	Data Warehousing/Data Mining	24	20.8%	16.7%
3	Collaborative Learning	22	40.9%	27.3%
4	Workflow	12	25%	16.7%
5	E-commerce Development	54	22.2%	18.5%
6	E-commerce Application	36	25 %	19.4%

Table 10. The results from "Theme" experiment

7. Discussions

The hit ratios, that show how often the papers from the same track have fired on the top of a distance proximity table to a prototype from the same track, are presented in Tables 6 and 10 for a recall window 47 and 25. Before warning, that the values of hit ratios are rather low one should understand the nature of comparison that we made between automatic retrieval results and conference track division while calculating hit ratio values. The hit ratio values are calculated in the assumptions that tracks unite semantically close paper. Track division is subjective and makes a weak reference point for calculating hit ration values very relative. As was noticed in (Yarowsky and Florian, 1999), there are a number of different issues except topic of a paper, e.g. conflict of interest, to be considered while routing an article to a particular track in conference settings. Sometimes the semantic similarity of short text documents is not obvious for

the reader and can be determined only after very careful linguistic analysis, for example, the text of the abstracts in Figure 1.

While analyzing the results, we noticed that word usage and some peculiarities of the written style of the scientific abstracts have a significant impact on the clustering ability of our methodology. All abstracts from research articles consist of the same components: introduction, method, results and discussion (dos Santos, 1996). Therefore the ranges of distance measures on word and sentence level were so narrow. The closeness on a word level of all examined abstracts (the Euclidian distance range is [0.484344...1.246202]) can be explained by the nature of textual content. The majority of abstracts contain words such as *paper, analysis, discusses, present, the, result, system, model, process, information*, which makes abstract vocabulary very specific and versatile. The meaning of the text plays an important role in the clustering results as well. The evidence to this conclusion is strong on the sentence level analysis. The peculiarity of the proximity table on the sentence level is that our system calculated only 116 unique distance metrics for 444 different abstracts. For instance, there are 43 out 443 abstracts are distant from INWEB 04 at 1.412314 and there are 8 out of 443 abstracts are distant from INWEB 04 at 1.414215. The closeness of all abstracts on the sentence level (the Euclidian distance range is [0.38517...1.414215]) can be explained by a particular academic writing style with specific sentence structure, since authors used the same words and word order to describe their achievements in information system research, e.g. *we present, our paper discusses, this paper describes*. We discovered that our prototype-matching clustering of the scientific text corpus is somewhat different from the theme division proposed by the organizing committee. However, the evaluation that is available in Table 10 has proved the methodology results promising.

As for the limitations of our study, we can consider the critique toward the scalability of the methodology, limited experimental data collection and result evaluation. However, the methodology evaluation was offered in (Visa, et al. 2002) by examining the similarities in different translation of the books of Bible. The scalability of the method has been examined on TREC data (Visa, et al. 2001).

8. Conclusion and Future Work

In this report we described the clustering of the scientific text corpus from the Hawaii International Conference on System Science-34 using to the prototype-matching clustering method. We aimed at establishing the semantic similarities among the conference papers by clustering the abstracts from them. The conference organizers of HICSS-34 had offered nontraditional cross-track theme classification of the submitted papers to help the conference attendees to visit all sessions relevant to their research needs. Our prototype-matching clustering method consists of text filtering, “smart” document encoding on word and sentence levels, creating word and sentence level histograms, and prototype matching phases. We formed the clusters according to the Euclidian distances between the text of a prototype and the rest of a document collection.

In the paper we have presented two experiments from the clustering sessions on a scientific abstract collection. In our first experiment, we tested the system ability to

retrieve the closest abstract according to content from the whole document collection. In the second experiment, we examined the semantic closeness of the papers from the same cross-track themes and their closest matches. Even though our clustering results turned out to be somewhat different from the cross-track division offered by the conference organizers, our method was able to capture some semantic similarities between the scientific abstracts. The specific limited vocabulary and conservative academic style of the abstracts had a strong impact on our clustering results.

As future work, we plan to consider trying out the method on the full-text articles from the HICSS-34 document collection.

9. Acknowledgement

We gratefully acknowledge Tomas Eklund for his valuable comments, Pr. Pirkko Walden for the initial task and the financial support of TEKES (grant number 40887/97) and the Academy of Finland.

10. Reference

Anick, P., J., Pelekhov, K., and Rus, D. (1999). A Practical Clustering Algorithms for Static and Dynamic Information Organization. ACM-SIAM Symposium on Discrete Algorithms, ACM Press.

Anick, P. and S. Vaithyanathan (1997). Exploiting Clustering and Phrases for Context-Based Information Retrieval. SIGIR 97, Philadelphia, USA, ACM Press.

Aslam, J., K. Pelekhov, et al. (1999). A Practical Clustering Algorithms for Static and Dynamic Information Organization. ACM-SIAM Symposium on Discrete Algorithms, ACM Press.

Cutting, D., D. Karger, et al. (1992). Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections. 15th Annual International SIGIR'92, Denmark, ACM Press, NY, USA.

dos Santos, M. (1996). "The textual organization of research paper abstracts in applied linguistics." Text **16**(4): 481-499.

El-Hamdouchi, A. and P. Willett (1986). Hierarchic Document Clustering Using Ward's Mehtod. ACM Conference on Research and Dvelopment in Information Retrieval, ACM Press.

Hand D., M. H., and Smyth P. (2001). Principles of Data Mining. Boston, USA, A Bradford Book, The MIT Press, 2001.

Jain, A., M. Murty, et al. (1999). "Data Clustering: A Review." ACM Computing Surveys **31**(3): 265-323.

Karanikas, H., Tjortjis, C., and Theodoulidis (2000). An Approach to Text Mining using Information Extraction. Principles and Practice of Knowledge Discovery in Databases (PKDD-2000), Springer-Verlag Publisher.

Kohonen, T. (1998). Self-Organization of Very Large Document Collections: State of the Art. Proceedings of ICANN98, the 8th International Conference on Artificial Neural Networks, Springer, London.

Larsen, B., and Aone, A. (1999). Fast and Effective Text Mining Using Linear-time Document Clustering. KDD-99, San Dieago, CA, USA, ACM.

Lawrence, S., K. Bollacker, et al. (1999). Indexing and Retrieval of Scientific Literature. 8th International Conference on Information and Knowledge Management, CIKM 99, Kansas City, Missouri, USA, ACM Press.

Lee, C. and H. Yang (1999). A Web Text Mining Approach Based on Self- Organizing Map. WIDM-99, Kansas City, MO, USA, ACM.

Merkl, D. and Schweighofer (1997). En Route to Data Mining in Legal Text Corpora: Clustering Neural Computation, and International Treaties. 8th International Workshop on database and Expert Systems Applications (DEXA'97), Toulouse, France, IEEE.

Schutze, H. and C. Silverstein (1997). Projection for Efficient Document Clustering. SIGIR 97, Philadelphia, PA, USA, ACM Press New York, NY, USA.

Sprague, R. H., Jr. (2001). Preface to The Hawaii International Conference on System Science 2001. The Hawaii International Conference on System Science 2001, Maui, Hawaii, University of Hawaii at Manoa.

Steinbach, M., G. Karypis, et al. (2000). A Comparison of Document Clustering Techniques. TextMining Workshop (KDD).

Szymkowiak, A., Larsen, J. and Hansen, L.K. (2001). Hierarchical Clustering for Datamining. KES-2001 Fifth International Conference on Knowledge-Based Intelligent Information Engineering Systems & Allied Technologies, Osaka and Nara, Japan.

Tan, A. (1999). Text Mining: The state of the art and the challenges. PAKDD-99, Workshop on Knowledge Discovery from Advanced Databases (KDAD'99), Beijing, China.

Toivonen, J., A. Visa, et al. (2001). Validation of Text Clustering Based on Document Contents. Machine Learning and Data Mining in Pattern Recognition (MLDM 2001), Leipzig, Germany, Springer-Verlag.

van Rijsbergen, C. (1979). Information Retrieval (Second Edition). London:, Butterworths.

Visa, A., J. Toivonen, et al. (2000). Toward Text Understanding - Classification of Text Documents by Word Map. Proceedings of AeroSence 2000, SPIE 14th Annual International Symposium on Aerospace/Defense Sensing, Simulating and Controls., Orlando, USA.

Visa, A., J. Toivonen, et al. (2002). "Contents Matching Defined by Prototypes: Methodology Verification with Books of the Bible." Journal of Management Information Systems **18**(4): 87-100.

Visa, A., J. Toivonen, et al. (2001). Prototype-matching - Finding Meaning in the Books of the Bible. Hawaii International Conference on System Science, HICSS-34, Maui, Hawaii, USA.

Willett, P. (1988). "Recent Trends in Hierarchic Document Clustering: A Critical Review." Information Processing and Management **24**(5): 577-597.

Witten, I., B. Z., et al. (1998). Text mining: A new frontier for lossless compression. Data Compression Conference '98, IEEE.

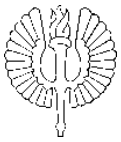
Yarowsky, D. and R. Florian (1999). Taking the load off the conference chairs: towards a digital paper-routing assistant. Joint SIGDAT Conference on Empirical Methods in NLP and Very LargeCorpora.

Zaiane, O. (1999). Resource and Knowledge Discovery from The Internet and Multimedia Repositories. School of Computing Science, Simon Frazer University.

Zamir, O. and O. Etzioni (1998). Web Document Clustering: A Feasibility Demonstration. SIGIR'98, Melbourne, Australia, ACM Press.

Turku Centre for Computer Science
Lemminkäisenkatu 14
FIN-20520 Turku
Finland

http://www.tucs.fi/



University of Turku
• Department of Information Technology
• Department of Mathematics

Åbo Akademi University
• Department of Computer Science
• Institute for Advanced Management Systems Research

Turku School of Economics and Business Administration
• Institute of Information Systems Science