An overview of Emerging Trend Detection (ETD)

1. INTRODUCTION

What is an emerging trend? An emerging trend is a topic area that is growing in interest and utility over time. For example, Extensible Markup Language (XML) emerged as a trend in the mid 1990's. Table 1 shows the results of a INSPEC®[Inspec] database search on the keyword 'XML' from 1994 to 1999 (no records appeared before 1994). As you can see from this table, XML was emerging from 1994 to 1997; by 1998 it was well represented as a topic area in computer science.

Year	Number of documents
1994	3
1995	1
1996	8
1997	10
1998	170
1999	371

Table 1: Emergence of XML in the mid-1990s

Knowledge of emerging trends is particularly important to individuals and companies who are charged with monitoring a particular field or company. For example, a market analyst specializing in biotech companies may want to review technical and news-related literature for recent trends that will impact the companies he is watching and reporting on. Manual review of all the available data is simply not feasible. Human experts who are tasked with identifying emerging trends need to rely on automated systems as the amount of information available in digital format increases.

An Emerging Trend Detection (ETD) application takes as input a large collection of textual data and identifies topic areas that are previously unseen or are growing in importance within the corpus. Current applications in ETD fall generally into two categories: fully-automatic and semi-automatic. The fully-automatic systems take in a corpus and develop a list of emergent topics. A

human reviewer would then peruse these topics and the supporting evidence found by the system to determine which are truly emerging trends. These systems often include a visual component that allows the user to track the topic in an intuitive manner [Davison][Swan]. Semi-automatic systems rely on user input as a first step in detecting an emerging trend [Porter][Roy]. These systems will then provide the user with evidence to indicate if the input topic is really emerging, usually in the form of user-friendly reports and screens that summarize the evidence available on the topic.

We begin with a detailed description of several semi-automatic and fully-automatic ETD systems in section 2. We discuss the components of an ETD system including: linguistic and statistical features, learning algorithms, training and test set generation, visualization and evaluation. In section 3 we review the ETD capabilities in commercial products. Our conclusions are presented in section 4. In section 5 Dan Phelps, from Kodak, describes the role of ETD systems in modern corporate decision making processes.

2. ETD SYSTEMS

As mentioned above, ETD systems can be classified as either fully automatic or semi-automatic. Semi-automatic systems require user input as a first step in detecting the emerging trends in a topic area. We have developed both fully and semi automatic systems which have successfully identified emerging trends. In this section we provide an overview of the components that are included in most ETD systems (input data sets, attributes used for processing, learning algorithms, visualization, evaluation), followed by a detailed description of several ETD systems

We begin with a discussion on the data that is used in ETD systems. The most commonly used data repository for ETD emerged from the Topic Detection and Tracking (TDT) project [TDT] that began in 1997. TDT research develops algorithms for discovering and threading together topically related material in streams of data, such as newswire and broadcast news, in both English and Mandarin Chinese. The TDT project, while not directly focused on emerging trend detection, has nonetheless encouraged the development of various fully automated systems that track topic changes through time. Several of those algorithms will be described in this section.

As part of the TDT initiative several data sets have been created. The TDT data sets are sets of news stories and event descriptors. Each story/event pair is assigned a relevance judgement. A relevance judgement is an indicator about the relevance of the given story to an event. See table 2 for several examples of the relevance judgement assignment to a story/event pair. Thus, the TDT data sets can be used as both training and test sets for ETD algorithms. The Linguistic Data Consortium (LDC) [LDC] currently has three TDT corpora available for system development, the TDT Pilot study (TDT-Pilot), the TDT Phase 2 (TDT2), the TDT Phase 3 (TDT3), as well as the TDT3 Arabic supplement.

Story Description	Event	Relevance Judgement
Story describes survivor's reaction after Oklahoma City Bombing	Oklahoma City Bombing	Yes
Story describes survivor's reaction after Oklahoma City Bombing	US Terrorism Response	No
Story describes FBI's increased use of survellance in government buildings as a result of the Oklahoma City Bombing	Oklahoma City Bombing	Yes
Story describes FBI's increased use of survellance in government buildings as a result of the Oklahoma City Bombing	US Terrorism Response	Yes

Table 2: Story/event pairs

Not all of the systems we describe rely on the TDT data sets. Other approaches to creation of test data have been used, such as manually assigning relevance judgements to the input data and comparing the system results to the results produced by a human reviewer. This approach is tedious and necessarily limits the size of the data set. Some of the systems we present use online databases such as the INSPEC®[Inspec] database, which contains engineering abstracts, or the US Patent database [USPTO], which allows searching of all published US Patents. The input data set, along with the selection of appropriate attributes that describe the input, is a critical component of each ETD system. Attribute selection is at the core of the tracking process, since it is the attributes that describe each input record and ultimately determine the trends.

The attributes obtained from the corpus data are input to the methods/techniques employed by each ETD system we describe below. As you will see, some research groups use traditional IR methodologies to detect

emerging trends, while others have focused on more traditional machine learning approaches such as those used in data mining applications.

Work in the areas of visualization-supported trend detection has explored techniques for identifying topics. When a user is trying to understand a large amount of data, a system that allows an overview, at multiple levels of detail and from multiple perspectives, is particularly helpful. One of the simplest approaches is a histogram, where bars indicate discrete values of actual data at some discrete time value. Information visualization is meant to complement machine learning approaches for trend detection. Plotting the patterns along a timeline allows us to see the rate of change of a pattern over time. For each algorithm described below, we will discuss the visualization component, showing how the component enhances the trend detection capabilities of the system.

The evaluation of an emerging trend detection system can be based on formal metrics, such as precision (the percentage of selected items that the system got right) and recall (the proportion of the target items that the system found), or by less formal, subjective means (e.g., answers to questions such as: Is the visualization understandable?). The particulars of an evaluation are related to the goals of the method and thus can vary greatly, but some justification and interpretation of the results should always exist to validate a given system.

2.1 Technology Opportunities Analysis (TOA)

Alan L. Porter and Michael J. Detampel describe a semi-automatic trend detection system for technology opportunities analysis in [Porter]. The first step of the process is the extraction of documents from the knowledge area to be studied from a database of abstracts, such as INSPEC®[Inspec]. The extraction process requires the development of a list of potential keywords by a domain expert. These keywords are then combined into queries using appropriate Boolean operators to generate comprehensive and accurate searches. The target databases are also identified in this phase (ex. INSPEC®[Inspec], COMPENDEX® [Compendex], US Patents [USPTO], etc.).

The queries are then input to the Technology Opportunities Analysis Knowbot (TOAK), a customer software package also referred to as TOAS (Technology Opportunities Analysis System). TOAK extracts the relevant documents (abstracts) and provides bibliometric analysis of the data. Bibliometrics uses information such as word counts, date information, word co-occurrence information, citation information and publication information to track activity in a subject area. TOAK facilitates the analysis of the data available within the documents. For example, lists of frequently occurring keywords can be quickly generated, as can lists of author affiliations, countries, or states.

In [Porter], the authors present an example of how the TOAK system can be used to track trends in the multichip module sub field of electronic and manufacturing and assembly. Figure 1 (from [Porter]) shows a list of keywords that appear frequently with 'multichip module' in the INSPEC®[Inspec] database. The authors observed that multichip modules and integrated circuits (particularly hybrid integrated circuits) co-occurred very frequently. An additional search using the US Patent database showed that many patents had been issued in the area of multichip modules. Furthermore, the integrated circuits activity was more likely to be US based, while large scale integration activity was more likely to be based in Japan.

TOAK is meant to be used by a human expert in an interactive and iterative fashion. The user generates initial queries, reviews the results and is able to revise the searches based on his/her domain knowledge. TOA represents an alternative approach to the time-consuming literature search and review tasks necessary for market analysis, technology planning, strategic planning or research.

TABLE 2
Multichip Module Keywords and Frequencies [INSPEC Database]

Keyword	Number of articles	Keyword	Number of articles
Multichip modules	842	Circuit layout CAD	69
Packaging	480	Tape automated bonding	68
Hybrid integrated circuits	317	Printed circuit manufacture	66
Module	271	Printed circuit design	65
Integrated circuit technology	248	Thin film circuit	62
Integrated circuit testing	127	CMOS integrated circuits	56
Substrates	101	Soldering	50
VLSI	98	Optical interconnections	48
Surface mount technology	93	Lead bonding	44
Flip-chip devices	93	Integrated optoelectronics	43
Integrated circuit manufacture	88	Printed circuits	42
Ceramics	85	Production testing	41
Circuit reliability	80	Reliability	41
Polymer films	79	Microassembling	38
Cooling	70	Circuit CAD	35
Metallisation	69	Microprocessor chips	35

Figure 1: Sample output from TOAK showing keywords that co-occur with multichip modules

Input Data and Attributes

The INSPEC®[Inspec] database served as the corpus for TOA and its related software, TOAK. (Major national and international publication databases, along with a major U.S. Patent database [USPTO] are other corpus possibilities.) Two opportunities exist for attribute selection. First (Table 3), a list of keywords (a

single word or multiple words) and their possible combinations (using Boolean operators) are supplied to TOAK, which retrieves all relevant items. The number of keyword occurrences and keyword co-occurrences – the appearance of two keywords in the same item – are calculated per year and over all years. A second pass (Table 4) involves selecting all phrases (single- and multi-word) from a specific field and calculating the number of items that contain each phrase. For example, every phrase in the keyword field of all items may be counted, or every phrase in the affiliation field. [Porter]

Attribute	**Detail**	**Generation**
n-grams[1]	Examples: multichip modules, ball grid array	Manual
Frequency	Number of each *n*-grams occurrence	Automatic
Frequency	Number of each *n*-grams co-occurrence (see text)	Automatic
Date	Given by year	Automatic

Table 3: TOA First Pass Attributes

Attribute	**Detail**	**Generation**
Field	A section of an item. Examples: keyword, city	Manual
Frequency	Number of each n-gram in a field	Automatic

Table 4: TOA Second Pass Attributes

Learning Algorithms

Like most of the systems to facilitate trend detection in textual collections, TOA relies on the expertise of the user who is researching a given area. TOAK provides access to many different data sources, including INSPEC®[Inspec], Engineering Index [COMPENDEX], US Patents [USPTO] and others, but is necessarily limited as not all R&D work is patented or published. The power of TOAK resides in the visual interface and easy access to different views of the data. There are no inherent learning algorithms present in the system; the user is solely responsible for trend detection.

[1] an n-gram is a sequence of n words. For example, the phrase "stock market" is a bigram (or 2-gram).

Visualization

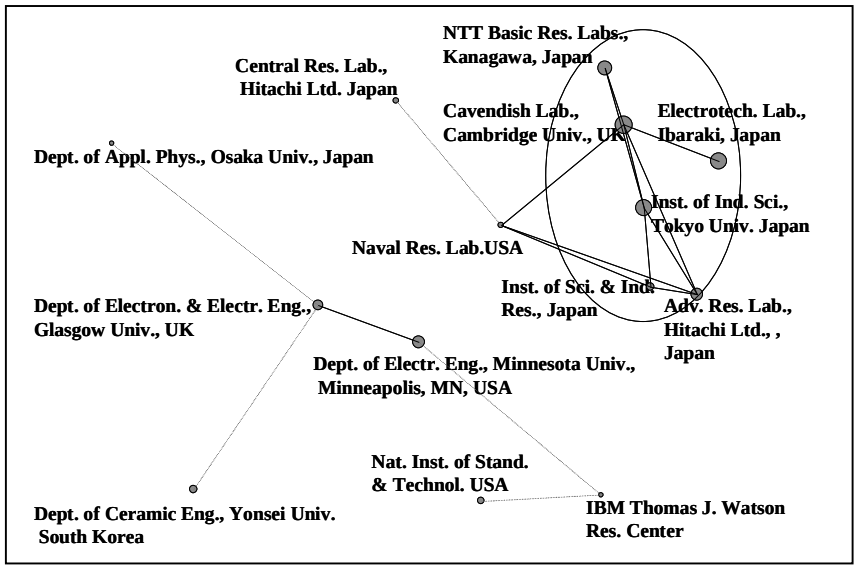

Figure 2: Research organizations that are afflilated with research in nanotechnology

Visualizations in TOA include frequency tables, histograms, weighted ratios, log-log graphs, Fisher-Pry curves, and technology maps [Porter]. These tools present information graphically using various linking and clustering approaches such as multi-dimensional scaling. In multi-dimensional scaling the goal is to reduce an n dimensional space to 2 or 3 dimensions. For example, figure 2 (taken from [Porter and Jhu]) shows a mapping of the organizations that are affiliated with research in nanotechnology. In this case thre are 40 affiliations, and the map will have 40 nodes. A path-erasing algorithm is used to build a proximity matrix to connect the nodes. This handles the problem of distortion, a common problem that occurs when mapping a high dimensional spatial relation to a 2D or 3D space. TOA can also present maps based on other attributes that are available in the data. Attributes such as source, country of origin or author are commonly used. Similar techniques are used to generate keywords maps that represent relationships among frequently occurring index terms, and principal components maps that represent relationships among conceptual clusters. These maps represent co-occurrence and correlative information gathered from within the dataset.

Evaluation

Identification of trends is left to the user in this semi-automatic method. But TOA can be evaluated on how well it presents information to the user who must make emerging trend judgments. Visualizations are meant to significantly increase understanding of the data, and intuitively do. But there is no support for the efficacy of these tools, apart from the authors' claims. Solutions exist for evaluating this type of method. For example, independent sources or focus groups could strengthen the argument that the visualizations are indeed helpful. ThemeRiver [section 2.5] takes this usability approach for evaluation. Formal metrics, even with a semi-automatic method, can also be utilized as in CIMEL [section 2.2].

2.2 Constructive, collaborative inquiry-based multimedia e-learning(CIMEL)

CIMEL is a multi-media framework for constructive and collaborative inquiry-based learning [Blank]. The semi-automatic trend detection methodology described in [Roy] has been integrated into the CIMEL system in order to enhance computer science education. A multimedia tutorial has been developed to guide students through the process of emerging trend detection. Through the detection of incipient emerging trends, the students see the role that current topics play in course-related research areas. Early studies of this methodology, using students in an upper-level computer science course, show that use of the methodology improves the number of incipient emerging trends identified.

Our semi-automatic algorithm employs a more robust methodology than TOA because the user base is assumed to be individuals who are learning a particular area, as opposed to domain experts. The methodology relies on web resources to identify candidate emerging trends. Classroom knowledge, along with automated 'assistants', is then used to evaluate the identified candidates. This methodology is particularly focused on incipient trends (those that are occur for the first time).

Identify a main topic area for research (ex. Object databases)
Identify recent conferences and workshops in this area (ex. OOPSLA for Object Oriented Programming)
Review content and create a list of candidate emerging trends
Evaluate each emerging trend identified in step 3, using general web research tools (ex. Google™ search)
For each emerging trend remaining after step 4, verify the trend using an INSPEC®[Inspec] database search

Table 5: Methodology for detecting emerging trends

The methodology is outlined in table 5. In the step 2 of this methodology (after a main topic area has been identified) the user is directed to a list of recent conferences and workshops and instructed to review the content to identify a list of candidate emerging trends. Next, the user is directed to a general-purpose web search engine to find other references to candidate emerging trends identified in step 3. Searches on the emerging trend phase, along with terms such as 'recent research,' 'new approach,' etc. are employed to improve the precision of the search results. The user is provided with a detailed algorithm that includes parameters for evaluation of the pages returned from the search engine. The candidate emerging trend may be rejected as a result of this search. In addition, other candidate emerging trends may be identified in this step.

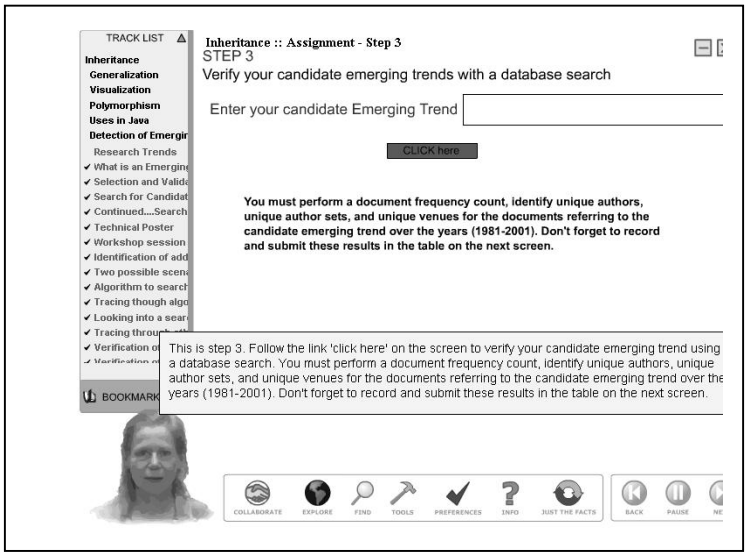

Figure 3: CIMEL Tutorial

Finally the student is asked to verify candidate emerging trends using document count, author and venue spread using an INSPEC®[Inspec] database search. Again, to make the trend detection process easier, this step has been automated [Gevry]. Students are only required to enter the candidate emerging trend (Figure 3) which they have already identified in steps 3 and 4 while the database search tool automatically generates document count, unique author sets, unique co-author sets, a list of unique venues, etc. pertaining to the chosen candidate emerging trend. The tool also provides a link to the corresponding abstracts, which can be accessed by clicking on individual document titles. This feature of the tool is important, as the student still has to make his or her own

decision, considering the information provided by the tool and using the heuristics provided in the tutorial, to validate a candidate emerging trend. Again, the user is given specific parameters for determining if the candidate emerging trend is truly an incipient emerging trend.

For example, the students in an upper-level Object Oriented Software Engineering course would be asked to find an emerging trend in the field of Object Databases. Several conference web sites would be provided, including the Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) website. A manual review of the content of papers presented at OOPSLA '01 should lead the student to the candidate emerging trend 'XML Databases.' A search of the web using Google™ would detect additional papers related to XML databases, providing further evidence that this is an emerging trend. Finally, the student would be directed to the INSPEC®[Inspec] database. A search on XML (<and>) Databases (<and>) Object-oriented will reveal the information shown in table 6. Further inspection shows multiple author sets and publication venues, confirming that XML Databases is an incipient emerging trend in the field of Object Databases.

Year	Number of documents
Prior to 1999	0
1999	5
2000	11
2001	5

Table 6: Evidence of XML Databases as an incipient emerging trend

Input Data and Attributes

The corpus for this semi-automatic methodology can be any web resource. A description of the main topic is chosen, which can consist of any text. An initial search of recent conferences and workshops is performed to identify candidate emerging trends. Phrases associated with emerging trends[2] are used in conjunction with either the main topic or the candidate emerging trends to locate candidate emerging trends using a web search engine. Several attributes guide this initial decision-making process (Table 7), including the current year, the number of times either the main topic or candidate emerging trend appears on the page, the number of supporting terms on the page, and the line or paragraph containing the main topic/candidate emerging trend and supporting term. [Roy] The validation step (Table 8) involves automatically calculating

[2] The list of current supporting terms: most recent contribution, recent research, a new paradigm, hot topics, emergent, newest entry, cutting edge strategies, first public review, future, recent trend, next generation, novel, new approach, proposed, current issues.

four frequencies across time: the number of unique documents, unique authors, unique author sets, and unique venues [Gevry]. These frequencies help the user make a final emerging trend determination. For example, an increase in the number of documents that reference the main topic and candidate emerging trend over time indicates a true emerging trend. On the other hand, if one or two documents appear in different years by the same author, the trend is not emerging. [Roy].

Attribute	**Detail**	**Generation**
n-gram	Main topic, e.g., object databases	Manual
n-gram	Candidate trend, e.g., XML Databases	Manual
n-gram	Supporting terms	Automatic
n-gram	Search item – any Boolean <and> combination of the previous attributes, e.g., XML <and> novel	Automatic
Date	Given by year	Automatic
Frequency	Number of times main topic/candidate trend occurs on page	Automatic
Frequency	Number of times helper term occurs on page	Automatic
n-gram	Line or paragraph containing the main topic/candidate trend and helper term in a given document	Manual

Table 7: CIMEL Initial Step Attributes

Attribute	**Detail**	**Generation**
Frequency	Number of unique authors, per year	Automatic
Frequency	Number of unique documents, per year	Automatic
Frequency	Number of unique author sets, per year	Automatic
Frequency	Number of unique venues, per year	Automatic

Table 8: CIMEL Validation Step Attributes

Learning Algorithms

Like TOA, the CIMEL system relies on the user to detect emerging trends. No machine learning component is included. Instead CIMEL relies on a precisely defined manual process. Like TOA, this system is restricted by the

electronic availability of documentation in a given subject area. Furthermore, the INSPEC®[Inspec] query tool is currently based on abstracts that are downloaded to a local database, which must be periodically refreshed. Unlike TOA, CIMEL provides specific parameters for identifying an emerging trend, rather than solely relying on the domain expertise of the user.

Visualization

At the current time the visualization component for trend detection CIMEL is under development.

Evaluation

Two groups of students in a Programming Languages class were asked to identify emerging trends in the area of Inheritance. Group B (experimental) viewed a multimedia tutorial on the methodology and case study; Group A (control) did not. Hypothesis testing was performed on the standard precision metric of the groups. Precision for a student was found by dividing the number of actual emerging trends found (one or two for this experiment) by the number of total trends found (two, if the student completed the assignment). Recall was not determined since a complete list of emerging trends was not available. A lower tail *t*-test concluded with 95% confidence that the mean precision of students that used the methodology (Group B) was significantly greater than the mean precision of students that did not use the methodology (Group A). These results suggest that detecting emerging trends is much more likely when using the methodology [Roy Thesis].

2.3 TimeMines

The TimeMines system [Swan] takes free text data, with explicit date tags, and develops an overview timeline of the most important topics covered by the corpus. Figure 4 (taken from [Swan]) presents the sample output from TimeMines. TimeMines relies on Information Extraction (IE) and Natural Language Processing (NLP) techniques to gather the data. The system uses hypothesis testing techniques to determine the most relevant topics in a given timeframe. Only the 'most significant and important information' (as determined by the program) is presented to the user.

TimeMines begins its processing with a default model which assumes the occurrence of a feature depends only on its base rate of occurrence, and does not vary with time. Each feature in a document is compared to the default model. A statistical test is used to determine if the feature being tested is significantly different than what the model would expect. If so, the feature is kept for future processing, if the feature is not significantly different than the baseline, it is ignored.

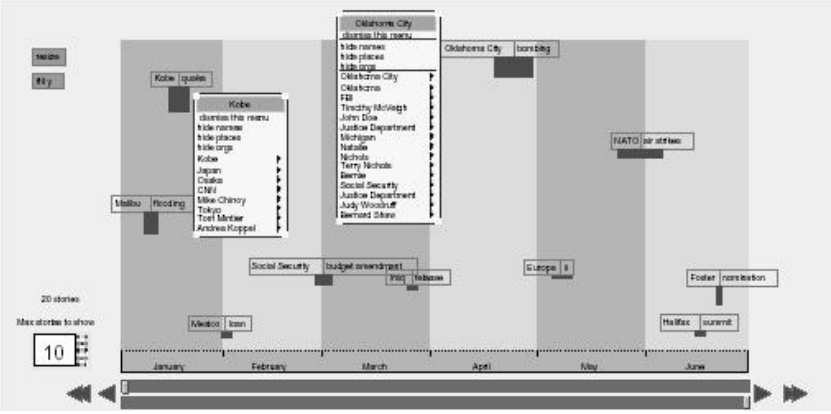

Figure 4: TimeMines sample output

The reduced set of features that is developed using the first round of hypothesis testing is then input into a second processing phase which groups related features together. The grouping again relies on probabilistic techniques that combine terms that tend to appear in the same timeframes into a single topic. Finally, a threshold is used to determine which topics are most important and these are displayed via the timeline interface (figure 4). The threshold is set manually, and was determined empirically.

Like TOA, TimeMines present a model of the data, without drawing any specific conclusions about whether or not a topic is emergent. It simply presents the most important topics to the user, and relies on the user's domain knowledge for evaluation of the topics.

Input Data and Attributes

The TDT and TDT-2 corpora were date tagged and part-of-speech tagged with JTAG [Xu] for this method (TDT-2 was preliminarily tagged with Nymble [Bikel]). In the TimeMines system, an initial attribute list of all "named entities" and certain noun phrases is generated. A named entity is defined as a specified person, location, or organization (found by the Badger IE system [Fisher] in the given example). Noun phrases match the regular expression (N|J)*N for up to five words, where N is a noun, J is an adjective, | indicates union, and * indicates zero or more occurrences. The date-specified documents are thus represented as a "bag of attributes", where each attribute is true or false (i.e., whether the named entity or noun phrase is contained in the document or not). The attributes are shown in table 9.

Attributes	**Detail**	**Generation**
n-gram	Person, location, or organization	Automatic

Regular expression	Follows (N|J)*N pattern for up to five words, e.g., hot topic research	Automatic
Presence	"True" if the *n*-gram or regular expression occurs in the document, else "False". Each document has a presence attribute for every *n*-gram and regular expression.	Automatic
Date	Given by day	Automatic

Table 9: TimeMines Attributes

Learning Algorithms

There are two separate machine learning aspects present in the TimeMines application. First, TimeMines must select the 'most significant and important information' to display. To do this, TimeMines must extract the 'most significant' features from the input documents.

TimeMines uses a statistical model based on hypothesis testing to choose the most relevant features. The system assumes a stationary random model for all features (noun phrases and named entities) extracted from the corpus. The stationary random model assumes that all features are stationary (meaning they do not vary over time) and the random processes generating any pair of features are independent. Features whose actual distribution matches this model are considered to contain no new information and are discarded. Features that vary greatly from the model are kept for further processing. The hypothesis testing is time dependent. In other words, for a specific block of time, a feature either matches the model (at a given threshold) or violates the model. Thus the phrase 'Oklahoma City Bombing' may be significant for one time slice, but not significant for another.

After the feature set has been condensed, TimeMines uses another learning algorithm, again based on hypothesis testing. Using the reduced feature set, TimeMines checks for features within a given time period that have similar distributions. These features are grouped into a single 'topic.' Thus each time period may be assigned a small number of topic areas, represented by a larger number of features.

One potential drawback of ranking the general topics derived from the significant attributes was discussed [Swan]. The occurrence of an attribute is measured against all other occurrences of it in the corpus, thus a consistently heavily used attribute may not distinguish itself properly. The Kenneth Starr-President Clinton investigation is unquestionably the most covered story in the TDT-2 corpus, yet ranked 12th because it is so prevalent throughout. Against a longer time period, including time after coverage had died down, the story probably would have ranked 1st.

Like all of the algorithms we present here, the final determination of whether or not a topic is emerging is left to the user, but unlike CIMEL [section 2.2] and TOA [section 2.1], the user does not direct the TimeMines system. This system is completely automated; given a time-tagged corpus and it responds with a graphical representation of the topics that dominate the corpus during specific time periods.

Visualization

TimeMines generates timelines automatically for visualization of temporal locality of topics and the identification of new information within a topic. The x–axis represents time, while the y-axis represents the relative importance of a topic. The most statistically significant topic appears at the top (Figure 4). Each block in the visualization interface includes all the terms used to describe a topic and thus indicates the coverage within the corpus. Clicking on a term (named entity or noun phrase) pops up a menu of all the associated features of that type within the topic, and a sub-menu option allows the user to choose this feature as the label, or to obtain more information about the feature. However no effort is made to infer any hierarchical structure in the appearance of the feature in the timeline.

Evaluation

Two hypotheses are evaluated: Do term occurrence and co-occurrence measures properly group documents into logical time-dependent stories, and, are the stories themselves meaningful to people? [Swan] A randomization test [Edgington] was conducted to support the first hypothesis. The documents were shuffled and assigned an alternate date, but were otherwise left intact. From an information retrieval (IR) standpoint the corpus looked the same, since term frequency and inverse document frequency were preserved. The authors concluded results from the test overwhelming suggest the groupings are logical and not random.

The second hypothesis was explored with two methods of evaluation but results were inconclusive. The first evaluation method used precision and recall metrics from IR. The January 1996 *Facts on File* [Facts] listed 24 major stories, which were used as the "truth" set to compare with the TimeMines-generated major stories. Precision was defined as the number of *Facts on File* major stories found by TimeMines divided by the total number of *Facts on File* major stories. Recall was defined as the number of *Facts on File* major stories found by TimeMines divided by the total number of TimeMines-generated major stories. A precision of 0.25, a recall of 0.29, and an examination of the stories listed in the truth and generated sets suggested an incompatibility between the two sets. Whether the underlying problem was the specific truth set or the use of precision and recall was not addressed.

The second evaluation method attempted to tune the χ^2 threshold. Four students manually determined whether the automatic groupings related to zero,

one, or multiple topics. But based on a pair wise Kappa statistic, the manual results could not be distinguished from random results. [Swan, Allan]

2.4 New Event Detection

New event detection, also referred to as first story detection, is specifically included as a subtask in the TDT initiative. New event detection requires identifying those news stories that discuss an event that has not already been reported in earlier stories. New event detection operates without a predefined query. Typically algorithms look for keywords in a news story and compare the story with earlier stories. New event detection insists that the input be processed sequentially in date order. Only past stories can be used for evaluation, not the entire corpus.

A new event detection algorithm based on a single pass clustering algorithm is presented in [Allan]. The content of each story is represented as a query. When a new story is processed, all the existing queries (previous stories) are run against it. If the 'match' exceeds a predefined threshold (discussed below) the new story is assumed to be a continuation of the query story. Otherwise it is marked as a new story.

An interesting characteristic of news stories is that events often occur in bursts. Figure 5, taken from [Yang] shows a temporal histogram of an event where the X axis represents time in terms of days (1 through 365) and the Y axis is the story count per day.

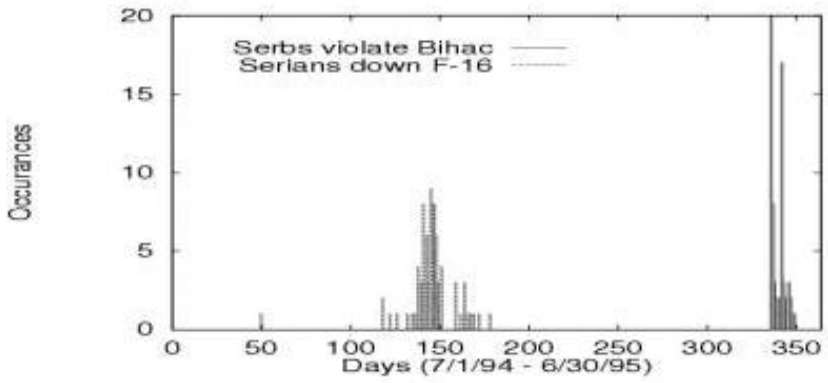

Figure 5: temporal histogram of news data

News stories discussing the same event tend to be in temporal proximity and hence lexical similarity and temporal proximity are considered to be two criteria

for document clustering. Also, a time gap between the bursts as indicated in the graph discriminates between two distinct events. New event detection uses a classifier-based approach to identify new stories. The system incorporates the use of temporal proximity and is more likely to match stories that appear in the same timeframe.

With proper tuning the algorithm was able to separate news stories related to the Oklahoma City Bombing from those about the World Trade Center bombing. However, some stories could not be detected. For example, the crash of Flight 427 could not be distinguished from other airplane accidents, and the OJ Simpson trial could not be separated from other court cases.

Input Data and Attributes

All stories in the TDT corpus deemed relevant to 25 selected "events" were processed. For new event detection, each story is represented by a query and threshold. Table 10 lists all the attributes required for the query. The *N* most frequent single words comprise the query, and are weighted and assigned a "belief" value by the Inquery system [Allan, Ballesteros, et al.], indicating the relevance of the word in the story to the query. Belief is calculated using term frequency and inverse document frequency. Term frequency is defined by the number of times the word occurs in the story, the length of the story, and the average length of a story in the collection. Inverse document frequency is the log of the number of stories in the collection divided by the number of stories that contain the word.

Attribute	**Detail**	**Generation**
Unigram	A single word	Automatic
Frequency	Number of times unigram occurs, per story	Automatic
Count	Total number of unigrams, per story	Automatic
Mean	Average number of unigrams per story	Automatic
Frequency	Number of stories in which unigram occurs	Automatic
Count	Number of stories	Automatic
Date	Given by available granularities	Automatic

Table 10: New Event Detection Attributes

Learning Algorithms

The algorithm presented in [Allan] is based on a single-pass clustering algorithm that detects new stories by comparing each story processed to all of the previous stories detected, which are saved as queries in the system. As each

incoming story is processed, all previous 'queries' are run against it. If a story does not match any of the existing queries, the story is considered a new event.

The system relies on a threshold to match the queries to the incoming stories. The initial threshold for a query is set by evaluating the query with the story from which it originated. If a subsequent story meets or exceeds this initial threshold for the query, the story is considered a match. The threshold function uses the evaluation function of the Inquery system [Allan, Ballesteros, et al]. Since new event detection demands that documents be processed in order, traditional IR metrics such as such as document frequency (number of documents containing the term) and average document length are not readily available. To overcome this problem, an auxiliary collection is used. The thresholding function was adapted further to take advantage of the time dependent nature of the news story collection. A time penalty was added to the threshold, increasing the value required to 'match' a story, as the stories grow further apart over time.

Like the TimeMines system, the new event detection system described here is completed automated. Given a corpus, it provides a list of 'new events' in the form of news stories that first describe an occurrence. New event detection differs from emerging trend detection because it is not interested in the gradual change of a topic over time. Instead it is focused on the sudden appearance of an unforeseen event.

Visualization

The new event detection system is based on Lighthouse [Leuski and Allan], an interactive information retrieval system, which provides a ranked list of search results together with 2 and 3 dimensional visualizations of inter-document similarities. After events are extracted, a visual timeline is constructed to show how those events occur in time or how they relate to one another.

Evaluation

Since this is a detection task, miss (false negative) and false alarm (false positive) rates were examined more thoroughly than the more conventional precision and recall [Allan, Papka, et al.]. Arriving at meaningful thresholds for these rates was difficult, and as a complement, Detection Error Tradeoff (DET) curves [Martin] were studied. DET curves highlight how miss and false alarm rates vary with respect to each other (each is plotted on an axis in a plane). A perfect system with zero misses and false alarms would be positioned at the origin, thus, DET curves "closer" to the origin are generally better. "Close" was defined as the Euclidean distance from the DET curve to the origin [Allan, Papka, et al.].

Eleven passes were taken through the corpus, removing zero to ten leading stories about an event. Using nearly all (400) single-word attributes in the

queries resulted in averages of 46% for the miss rate, 1.46% for the false alarm rate, 54% for recall, and 45% for precision.

2.5 ThemeRiver™

Similar to TimeMines, ThemeRiver™ [Havre] summaries the main topics in a corpus and presents a summary of the importance of each topic via a robust user interface. The topical changes over time are shown as a river of information. The 'river' is made up of multiple streams. Each stream represents a topic and each topic represented by a color and maintains its place in the river, relative to other topics. See figure 6 for an example.

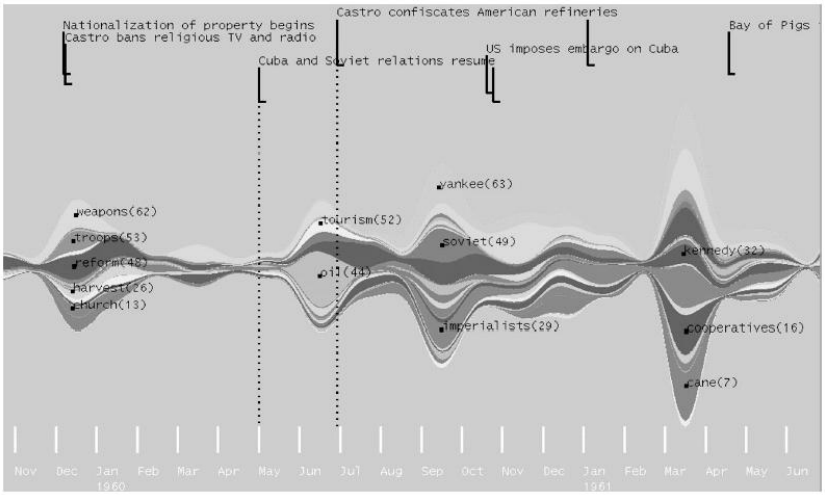

Figure 6: ThemeRiver™ sample output [taken from Havre]

The river metaphor allows the user to track the importance of a topic over time (represented on the horizontal access). The data represented in figure 6 is from Fidel Castro's speeches. You can see that Castro frequently mentioned oil just before American oil refineries were confiscated in 1960 (shown as the second vertical line from the left in figure 6). Oil is the large bubble immediately preceding this dotted line in the middle of the river. At no other time did Castro dwell on that topic in the 18-month period represented by this corpus.

Such patterns in the data may confirm or refute the user's knowledge of hypotheses about the collection. Like TOA and TimeMines, ThemeRiver™ does not presume to indicate which topics are emergent. The visualization is

intended to provide the user with information about the corpus. ThemeRiver presents a topic- or feature-centered view of the data. This topic-centered view is a distinguishing characteristic of ETD. Related areas in information retrieval, such as text filtering and text categorization, usually are document-centered.

Input Data and Attributes

The corpus in the example presented consisted of speeches, interviews, articles, and other text about Fidel Castro over a 40-year period. ThemeRiver automatically generates a list of possible topics, called "theme words", of which a subset is manually chosen for the attributes (the example narrowed the list to 64). Counts of how many documents contained a particular theme word for each time interval provide the input for the method. An alternate count, using the number of occurrences of the theme word for each time interval was suggested but not implemented. [Havre, Hetzler, Whitney, et al.]

The automatic method for generating the initial list of theme words was not specified, nor was the procedure for deciding which or how many of the theme words should be included in the subset as an actual attribute. It does appear that counts are computed after these attributes are chosen, effectively making this attribute selection a manual process (i.e., not automatic based strictly on the counts, see Table 11).

Attribute	**Detail**	**Generation**
Unigram	A single word	Manual
Frequency	Number of documents in which unigram occurs, per time interval	Automatic
Date	Given by month	Automatic

Table 11: ThemeRiver Attributes

Learning Algorithms

The ThemeRiver™ application does not use a learning algorithm per se. Like TOA, it provides a view of the data that an experienced domain expert can use to confirm or refute a hypothesis about the data. ThemeRiver™ begins by binning time-tagged data into time intervals. A set of terms, or themes, that represent the data is chosen and the 'river' is developed based on the strength of each theme in the collection. The themes are chosen by automatically developing a list of words that are present in the data and then manually selecting a subset that represent various topics. The number of documents containing the word determines the strength of each theme in each time interval. Other methods of developing the themes and strengths are possible. The visual component of ThemeRiver™ is the most important aspect of this work, particularly as it applies to trend detection.

Visualization

The ThemeRiver™ system uses the river metaphor to show the flow of data over time (Figure 6). While the "river" flows horizontally, each vertical section of the river corresponds to an ordered time slice and a colored current within the "river" identifies each topic or theme. The width of the "river" changes with the emergence or disappearance of topic thereby making the system effective in cases where there is no major variation in topic. Thus the width of the river represents the collective strength of the selected theme.

The curves in figure 6 show how interpolation is done to obtain a river metaphor. The idea is to produce a smooth curve with positive stream width for better visual tracking of stream across time. Even though this technique aids human pattern recognition, a histogram may be more accurate. The algorithm interpolates between points to generate the smooth curves (continuity in the flow of the river).

Figure 7: Color family representation for ThemeRiver, taken from [Havre, Hetzler, et.al.]

ThemeRiver™ makes judicious use of color, leveraging human perception and cognitive abilities. Themes are sorted into related groups, represented by a color family. This allows viewing of a large number of themes that can easily be separated due to color variation. For example in figure 7, "germany", "unification", "gdr" and "kohl" are represented by different shades of green and hence can easily be identified as belonging to same group.

Evaluation

Evaluation, or usability in such visual applications, was conducted with two users [Havre, Hetzler, Whitney, et al.]. After given some background information about the data, the users were asked about specifics that fell under the following five general questions:

- Did the users understand the visualization?
- Could they determine differences in theme discussion?
- Did the visualization prompt new observations about the data?
- Did the users interpret the visualization in any unexpected ways?
- How did the interpretation of the visualization differ from that of a histogram?

Observation, verbal protocol, and a questionnaire were used to gather feedback. This evaluation method is formalized well, but it lacks significance due to the small sample.

2.6 PatentMiner

The PatentMiner system was developed to discover trends in patent data using a dynamically generated SQL query based upon selection criteria given by the user [Lent]. The system is connected to an IBM DB2 database containing all granted US patents [USPTO]. There are two major components to the system, phrase identification using sequential pattern mining [Agrawal & Srikant, 1995; Srikant & Agrawal, 1996] and trend detection using shape queries [2].

The phrases that matched a increasing usage query on US patents [USPTO] in the category "Induced Nuclear Reactions: Processes, Systems and Elements" are shown in figure 8.

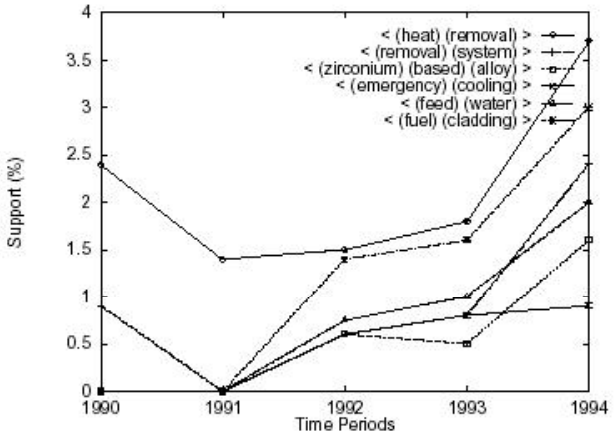

Figure 8: PatentMiner Sample Output

Input Data and Attributes

An IBM DB2 database containing all US Patents [USPTO] served as the basis for the corpus. Several procedures prepare the data for attribute extraction. Stopwords are removed. Long integer transaction ID's are assigned to the remaining words, indicating position in the document and occurrences of sentence, paragraph, and section boundaries. After a subset of patents is specified by category and date range, the Generalized Sequential Patterns (GSP) algorithm [Srikant] selects user-defined attributes, called "phrases". A phrase can be any sequence of words, with a minimum and maximum "gap" between any of the words. Gaps can be described in terms of words, sentences, paragraphs, or sections. For example, if the minimum sentence gap is one for the phrase "emerging trends", than "emerging" and "trends" must occur in separate sentences. Or if the maximum paragraph gap is one, than "emerging" and "trends" must occur in the same paragraph. A transaction time window indicates the number of words to group together before counting gap length. Only phrases above a minimum "support" – the percentage of all phrases that contain the user-defined phrase – are considered. A shape definition language (SDL) [Agrawal, Psaila, et al.] specifies which types of trends (e.g., upwards, spiked, etc.) are displayed. [Lent] Table 12 summarizes the attributes.

The number of phrases selected can be substantial, given their very open-ended nature. Two pruning methods are discussed in [Lent]. A non-maximal phrase of a maximal phrase may be ignored if the support of the two phrases is

similar. Or, a syntactic sub-phrase (general, high-level) might be preferred over a larger phrase (specific, low-level) initially, after which specific low-levels could be easier to pinpoint.

Attribute	Detail	Generation
n-gram	Search phrase, e.g., emerging trends	Manual
Size	Minimum gap, with distinct gaps for words, sentences, paragraphs, and sections.	Manual
Size	Maximum gap, with distinct gaps for words, sentences, paragraphs, and sections.	Manual
Size	Transaction time window, groups words in a phrase before determining gaps	Manual
Ratio	Support, found number of search phrases divided by total number of phrases	Manual
Date	Given by available granularities	Manual
Shape	Graphical trend appearance over time, e.g., spiked or downwards	Manual

Table 12: PatentMiner Attributes

Learning Algorithms

Most of the systems presented in this survey use traditional IR techniques to extract features from the text corpus that serves as input; the PatentMiner system takes a different approach. PatentMiner adapts a sequential pattern matching technique that is frequently used in data mining systems. This technique treats each word in the corpus as a transaction. The pattern matching system looks for frequently occurring patterns of words. The words may be adjacent, or separated by a variable number of other words (up to some maximum that is set by the user). This technique allows the system to identify frequently co-occurring terms and treat them as a single topic. [Lent] refers to the resulting collection of words as a phrase.

As with TimeMines, documents in the input data set are binned into various collections based on their date information. The above technique is used to extract phrases from each bin and the frequency of occurrence of each phrase in all bins is calculated. A shape query is used to determine which phrases to extract, based on the user's inquiry.

The shape query processing is another learning tool borrowed from data mining [2]. In the PatentMiner system, the phrase frequency counts represent a data store that can be mined using the shape query tool. The shape query has the ability to match upward and downward slopes based on the frequency counts. For example, a rapidly emerging phrase may have be frequency occurring for two concurrent time slices, then level off, before continuing on an upward trend. The shape query allows the user to graphically define various shapes for trend

detection (or other applications) and will retrieve the phrases with frequency count distributions that match the graph.

Like ThemeRiver™, TimeMines and others, the PatentMiner system presents a list of phrases to the user. The domain expert must then identify the true trends.

Visualization

The system is interactive; a histogram is displayed showing the occurrences of patents by year based on the user's selection criteria. The user has the ability to later on focus on a specific time period and to select various shape queries to explore the trends.

Evaluation

Like TOA, the PatentMiner system lacks an evaluation component. While it automatically generates and displays potential trends [Lent], no claim is made on the validity of those trends. The visualization is intuitive, but no user study on its effectiveness is reported. Furthermore, no metrics are presented that verify that the data conveyed reflects reality.

2.7 HDDI™

Our research has led to the development of the Hierarchical Distributed Dynamic Indexing (HDDI™) system [Pottenger, et al. – 2001, Bader, et al. – 2001, Bouskila and Pottenger - 2000]. The HDDI™ system provides core text processing including information/feature extraction, text mining/machine learning algorithms and support for evaluation for many applications, including ETD.

[Pottenger and Yang – 2001] describes an algorithm for detecting emerging trends in text collections based on semantically determined clusters of terms. The HDDI™ system is used to extract linguistic features from a repository of textual data and to generate clusters based on the semantic similarity of these features. The algorithm takes a snapshot of the statistical state of a collection at multiple points in time. The rate of change in the size of the clusters and in the frequency and association of features is used as input to a neural network that classifies topics as emerging or non-emerging.

In follow-on experiments we were able to duplicate the precision achieved by the neural network with the C4.5 decision tree learning algorithm [Zhou]. The run time performance was significantly better with the decision tree approach. These experiments show that it is possible to detect emerging concepts in an on-line environment.

We modeled the complex non-linear classification process using neural networks. The datasets, which included three years of abstracts related to processor and pipeline patent applications, were separated by year and a set of

concepts and clusters was developed for each year. In order to develop a training set, 14530 concepts were extracted and manually labeled. The system was, for example, able to correctly identify "Low power CMOS with DRAM" as an emerging trend in this dataset.

Like most other algorithms that we have reviewed, these algorithms rely on the domain expert for the final determination; thus the goal of the system is to screen out non-emerging topics whenever possible. Unlike the first story detection algorithms, our research focuses on integrative or non-disruptive emergence of topics, as opposed to the appearance of completely new topics.

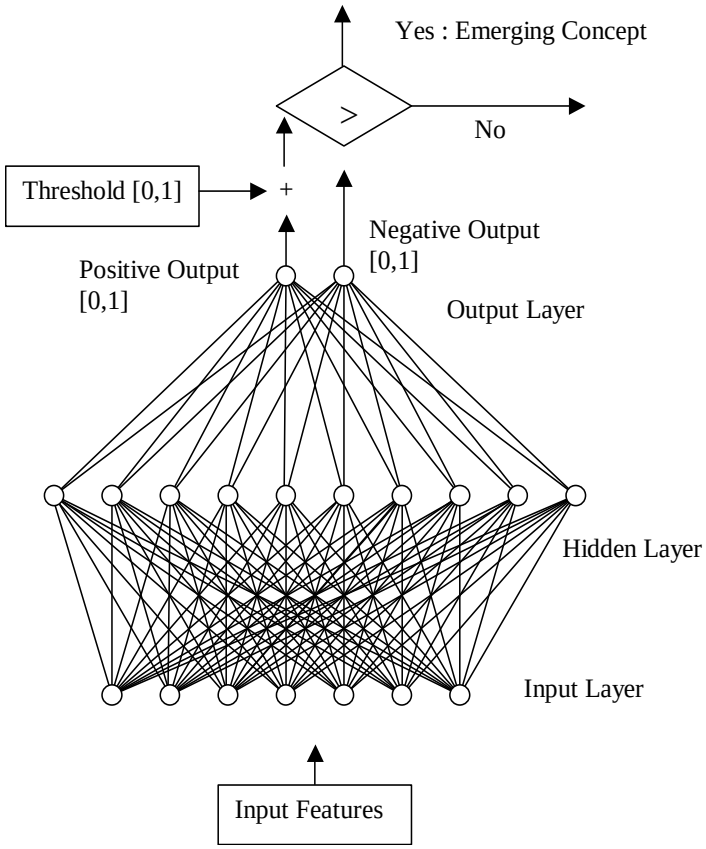

Figure 9: Using a neural net to detect emerging trends

Input Data and Attributes

Four databases were used to formulate a corpus: the US patent database [USPTO], the IBM patent database [Delphion], the INSPEC®[Inspec] database, and the COMPENDEX® [Compendex] database. Initial attribute selection

(Table 10) requires parsing and tagging before extraction. The parser retains only relevant sections of the original documents. The tagger attaches a part-of-speech label to each word using lexical and contextual rules. A finite-state machine extracts complex noun phrases ("concepts") according to the regular expression

C?(G|P|J)*N+(I*D?C?(G|P|J)*N+)*

where C is a cardinal number, G is verb (gerund or present participle), P is a verb (past participle), J is an adjective, N is a noun, I is a preposition, D is a determiner, ? indicates zero or one occurrence, | indicates union, * indicates zero or more occurrences, and + indicates one or more occurrence [Bader]. Counts of each concept and counts of co-occurrence of concept pairs are recorded [Pottenger, Yang].

A similarity between concept pairs is calculated based on co-occurrence information. The concepts are then grouped into regions of semantic locality [Chen, H.], [Pottenger]. The mean and standard deviation of the similarity, along with a parameter α of the number of standard deviations, determine the threshold τ used in the first step of the sLoc algorithm [Bouskila and Pottenger, 2000] for finding such regions; cluster size is used in the last step (both are pruning mechanisms). As τ increases, the number of arcs decreases, resulting in smaller but more focused semantic regions. Too large a τ could produce no regions, while too small a τ could produce too many regions. Thus, statistically finding the optimum values for τ (and the related α) is worthwhile, and work continues in this area. Empirical research supports an optimum value of α = 1.65. [Yang] The identification of clusters is an unsupervised learning step, which produces values used as attributes used in the supervised learning process discussed below.

An emerging concept satisfies two principles: it should grow semantically richer over time (i.e., occur with more concepts in its region), and it should occur more often as more items reference it [Pottenger and Yang]. Using a cluster-based rather than an item-based approach, an artificial neural network model takes seven inputs (and one tuning threshold parameter) to classify a concept as emerging or not [Pottenger and Yang]. The seven inputs are described in Table 14.

Attribute	**Detail**	**Generation**
Regular expression	A concept (see text for definition), e.g., emerging hot topic detection	Automatic
Frequency	Number of times each concept occurs over all documents	Automatic
Frequency	Number of co-occurrence of concept pairs over all	Automatic

	documents	
Similarity	Arc weight between concepts	Automatic
Mean	Average arc weight	Automatic
Standard Deviation	Arc weight standard deviation	Automatic

Table 13: HDDI™ Attributes for Regions of Semantic Locality Identification

Attribute	**Detail**	**Generation**
Frequency	Number of times concept occurs in trial year	Automatic
Frequency	Number of times concept occurs in the year before trial year	Automatic
Frequency	Number of times concept occurs in the year 2 years before trial year	Automatic
Frequency	Total number of times concept occurs in all years before trial year	Automatic
Count	Number of concepts in cluster containing the concept in trial year	Automatic
Count	Number of concepts in cluster containing the concept in the year before trial year	Automatic
Count	Number of words in the concept with length at least four	Automatic

Table 14: HDDI™ Attributes for Emerging Trend Detection

Learning Algorithms

As mentioned above, our fundamental premise is that computer algorithms can detect emerging trends by tracing changes in concept frequency and association over time. Our approach involves separating the data into time-determined bins (like PatentMiner and TimeMines) and taking a snapshot of the semantic relationships between terms. Two particular features were important in our model. Similar to other algorithms, the frequency of occurrence of a term should increase if the term is related to an emerging trend. Also, the term should co-occur with an increasing number of other terms if it is an emerging trend. To our knowledge, only our system has exploited term co-occurrence for automatic ETD.

The first learning model we employed is a feed-forward, back-progation artificial neural network (figure 9). We used a standard 3-layer network (one

input layer, one hidden layer, one output layer). The number of hidden neurons was varied to optimize our results.

The attributes were extracted as described in the previous section and used as input to both the neural network model [Pottenger and Yang], and to various other data mining algorithms such as c4.5 decision trees, support vector machines, etc. [Zhou]. In all cases, we determined that the algorithms could be trained to detect emerging trends. As with other systems, precision was fairly low (although much better than the baseline) and final determination as to whether or not a term displayed by the system represents an emerging trend must be left to a domain expert.

Visualization

Visualization is ongoing for trend detection within the HDDI™ system.

Evaluation

Both concept extraction and trend detection evaluations were performed. For concept extraction [Bader], mean precision (number of auto-generated correct concepts / total number of auto-generated concepts) and mean recall (number of auto-generated correct concepts / total number of human-generated concepts) were calculated for three collections. The Grainger DLI database [UIUC], the US Patent Office [USPTO] and IBM patent databases [Delphion], and Boeing Airline [need ref?] safety documents had precision ranges of [91.0, 96.4], [95.2, 99.2], and [91.0, 96.4] respectively, and recall ranges of [77.4, 91.3], [75.6, 90.6], and [61.9, 74.2] respectively, both with 95% confidence.

Automatic trend detection was measured by precision, recall, and F_β [Pottenger and Yang]. An average precision of 0.317 constituted a 4.62 factor of improvement over baseline precision; recall averaged 0.359. Either metric could be improved by altering the neural network threshold parameter. Since good recall was the primary focus, F_β, a weighted average of precision and recall with parameter β, was also examined. β is the precision weight and

$$F_\beta = \frac{(1+\beta^2)\cdot precision \cdot recall}{\beta^2 \cdot precision + recall}.$$

2.8 Other Related Work

Feldman and Dagan propose a technique for development of a hierarchical data structure from text databases in [Feldman]. This data structure then facilitates the study of concept distributions in text. The authors propose comparing the concept distributions from adjacent time periods. This approach to trend analysis seems promising; however, we were not able to obtain a more detailed description of the approach, or the experimental results, so we are unable to present a more comprehensive summary. Feldman has also been active in the

development of commercial products for emerging trend detection [ClearForest].

We have focused on research efforts that are based upon the extraction of trend information from various text databases; however, several research groups are using a different approach. [Chen and Carr] [Popescul. etal.] and [Leydesdorff] present algorithms that use citation information for trend analysis and tracking.

Several systems focus more on the visualization of textual data and can be adapted to trend detection at the discretion of the user. One such system, Envision [Nowell], allows users to explore trends in digital library metadata (including publication dates) graphically to identify emerging concepts. It is basically a multimedia digital library of computer science literature, with full-text searching and full-content retrieval capabilities. The system employs the use colors and shapes to convey important characteristics of documents, for example, the interface uses color to show the degree of relevance.

Plaisant, etal. describe a visual environment called LifeLines for reviewing personal medical histories in [Plaisant]. The visualization environment described in their work exploits the timeline concept to present a summary view of patient data.

The EAnalyst [Lavrenko] analyzes two types of data, textual and numerical, both with timestamps. The system predicts trends in numerical data based on the content of textual data preceding the trend. For example, the system predicts the trend in stock prices based on articles published before the trend occurs.

3. COMMERCIAL SOFTWARE AVAILABILITY

Commercial software products are available to aid a company interested in ETD. Some companies specialize in providing content [LexisNexis] [Moreover] [Northern Light] [IDC] [GartnerG2] [Factiva], some provide general purpose information retrieval capabilities such as text extraction [Applied Semantics] [Captiva], document management [HyBrix] [Banter], search and retrieval [Banter] [HyBrix] [LexisNexis] [TextAnalyst], categorization [Interwoven] [Invention Machine] [Semio] [Ser Solutions] [LexiQuest] [Stratify] [Verity], and clustering [Autonomy] [ClearForest] [Invention Machine] [TextAnalyst] [Thought Share]. While all of these products can be used to facilitate an ETD effort, only a few companies that we found have capabilities specifically geared toward trend analysis and detection. These products are discussed in this section.

3.1 Autonomy

Autonomy takes a corpus and produces a set of information clusters from within the data. These clusters can be used to identify new topics in the data by taking the cluster map from a previous time period and comparing it to the current one. Autonomy includes a cluster mapping tool called Spectrograph, which displays

a view of the clusters over time. This tool is designed to show trends in clusters, including the appearance/disappearance of topics, the spread and the cluster size. Applied to documents, the tool shows how the content stream changes over time, allowing the user to track trends and identify emerging trends.

Like the research tools described in the previous section, the Spectrograph is designed to provide the user with a view into the data. The domain expect can then use the data to form more general conclusions.

3.2 LexiQuest Mine

LexiQuest products use advanced natural language processing technology to access, manage and retrieve textual information. LexiQuest Mine is a text mining tool designed to help the user obtain new insights by identifying key concepts and the relationships between them. It employs a combination of dictionary-based linquistic analysis and statistical proximity matching to identify key concepts as well as the degree of relationship between concepts.

Concept identification is achieved through the use of unabridged dictionaries and thesauri. A large medical database, MeSH, may also be used. These contain terms that may consist of multiple words. Term occurrences and term co-occurrences are counted either by paragraph or document and are used to build relational concept maps. No machine learning algorithms are employed.

These relationships are displayed in a graphical map that displays the cumulative occurrence of concepts, and can be utilized for trend observation. Further analysis can be done by importing LexiMine data into the related Clementine [Clementine] tool. Ultimately the validity of the trend is left to the human domain expert and is not quantified in any way.

Peugeot is using LexiQuest Mine to monitor the activities of their competitors via web data.

3.3 ClearForest

ClearForest provides a platform and products to extract relevant information from large amounts of text and present summary information to the user. Two particular products: ClearResearch and ClearSight are particularly helpful for ETD applications. ClearResearch is designed to present a single-screen view of complex inter-relationships, enabling users to view news and research content in context. ClearSight provides simple, graphic visualizations of relationships between companies, people and events in the business world. It also provides real-time updates of new product launches, management changes, emerging technologies, etc. in any specified context. Users can also drill down further into each topic to view more information or read related articles.

4. CONCLUSIONS AND FUTURE WORK

We have described several semi-automatic and fully-automatic ETD systems, providing detailed information related to linguistic and statistical features, learning algorithms, training and test set generation, visualization and evaluation. This review of the literature indicates that much progress has been toward automating the process of detecting emerging trends, but there is room for improvement. All of the projects we found rely on a human domain expert to separate the emerging trends from noise in the system. Research projects that focus on creating effective processes to detect emerging trends, developing effective visualizations, and applying various learning algorithms to assist with ETD can and should continue. We are in the process of building an IT Infrastructure that includes algorithms for formal evaluation of ETD systems (www.xxx.com).

Furthermore, we discovered that few projects have used formal evaluation methodologies to determine the effectiveness of the systems being created. Development and use of effective metrics for evaluation of ETD systems is critical. The results published to date simply do not allow us to compare systems to one another. Wider use of the TDT [TDT] data sets is encouraged. Usability studies should be conducted for visualization systems. Additional data sets geared specifically toward trend detection should be developed. Toward this end, we have developed a back end system to the CIMEL tutorial [Blank]. This system will gather together the data generated by the students who use the tutorial. Eventually we will be able to automatically build training sets from this data.

We also note that projects tend to focus either on applying machine learning techniques to trend detection, or on the use of visualization techniques. Both techniques, when used alone, have proved inadequate thus far. Techniques that blend the use of visualization with machines learning may hold more promise. We are expanding our HDDI™ system to include a visualization component for trend detection. Early prototypes seem intuitively promising, but, as noted above, usability studies must be conducted to prove the effectiveness of our approach.

5. INDUSTRIAL COUNTERPOINT: IS ETD USEFUL? – Dan Phelps

Background: The Information Mining Group at Eastman Kodak has been following developments in the text mining field since 1998. Initially, our interest was in using text mining tools to help us do a better job of understanding the content of patents as part of our patent intelligence work. More recently, we have expanded our interest to include mining the science and technology literature to get an earlier indication of new developments that our R&D

management should consider in doing their tactical and strategic planning. We have had practical experience identifying suitable data sources, working with both custom and commercial tools, and presenting information in a form that our clients find useful. This background gives me a good perspective for commenting on the potential usefulness of Emerging Trend Detection tools and some of the challenges that will come up in actually trying to use them in the corporate environment.

The objective of Emerging Trend Detection is to provide an alert that new developments are happening in a specific area of interest in an automated way. It is assumed that a detected trend is an indication that some event has occurred. The person using the ETD software will have to look at the data to determine the underlying development. Whether the development is important or not is a judgment call that depends on the situation and the particular information needs of the person evaluating the data.

The need to become aware of new developments in science, technology, or business is critical to decision makers at all levels of a corporation. These people need to make better data-driven decisions as part of their daily work. They need data that is complete and available in a timely manner. Traditionally, people have learned about most new developments by reading various types of text documents or getting the information from someone else who has read the documents. As the pace of new developments accelerates and the number of documents increases exponentially, it will no longer be possible for an individual to keep up with what is happening using manual processes. There is a clear need for new tools and methodologies to bring some level of automation to these processes. ETD tools have the potential to play an important role in identifying new developments for corporate decision makers. These tools should help make it possible to look through more data sources for new developments and do it in less time than relying on current manual methods.

To better understand what capabilities an ETD tool must have to be useful, one has to look at who will be using the tool. There are three broad classes of potential users in a corporation. The first is the analyst or information professional who works to fulfill the information needs of others. The second is the individual contributor looking for information relevant to his own project. The third is a manager who needs to make strategic and/or tactical decisions.

The analyst works with information as the main component of his job. This person works on projects specified by clients. The output of a given project will be a report delivered to the client for use in the decision making process. The analyst is trained in information retrieval techniques, text mining techniques, etc. and is familiar with the various information sources needed to complete a given project. He or she has to be able to communicate the results of the work in a form the client can easily use and understand. Taking time to learn new tools and methodologies is an expected part of the job.

An ETD tool targeted for use by analysts can be quite complex because the analyst will be given sufficient training to become proficient in its use. Since the analyst would be using the tool for multiple projects, she will learn the

capabilities and limitations of the tool and only apply it where it is appropriate. One would expect a sophisticated user interface that would allow the analyst to access the relevant data sources, process the underlying text, and display the results in a meaningful way using computer graphics visualization techniques. The visualization scheme used must draw the analyst's attention to the trend and then allow the analyst to drill down in the data to find out what development(s) lead to the trend. The determination of whether or not a detected trend is important is complicated by the fact that the analyst does not always know what her client would judge to be important. Interaction between the analyst and the client is critical to make sure the needs of the client are met. This is typically an iterative process as the analyst learns more about what information the client really needs and the client finds out what information is actually available. Once the analysis is done, the ETD tool should have the ability to export information to facilitate the report generation process.

The scientist, engineer, or business person wanting to use an ETD tool to obtain project specific information needs a tool that is easy to learn and very intuitive use. Connecting to the appropriate data sources and the processing of the data inside the ETD tool must be transparent to the user. This person will typically have limited training in the tool and will use it only occasionally. He will not have the time to learn all the nuances of using the software. The information that comes out automatically will be what he will use. A simple graphical user interface with easily interpreted graphical visualizations is required. This person has the advantage that he is doing the work for himself so he can make the determination whether a newly detected trend is actually important or not.

An ETD tool meant to be used by management personnel must automatically be connected to the appropriate data sources, have an intuitive user interface, be very easy to learn, and provide output that is consistent with a format the manager is comfortable with. Extremely sophisticated visualizations that are difficult to interpret and require high levels of interaction will not be useful in this environment. In the current corporate culture, managers will not have the time to take any but the most cursory training in new tools. This means they will probably not be able to operate the ETD tool effectively to do the complete analysis themselves. They are typically much more comfortable having an analyst assemble the information and provide a executive summary. The summarized information and the underlying data could be presented to them using a browser format that would allow them to look at the high level results and then drill down into the detail when they find something they are particularly interested in.

No matter how capable an ETD tool becomes, the quality of the output will depend on the quality of the input data. Since ETD tools are supposed to identify new developments, the data processed through the tool must be current. There are several news type services (Dow Jones Interactive, Reuters, etc.) that supply all types of news on a continuous basis. Currently there are not equivalent services for science and technology types of information. One has to

search a variety of sources to find the required information. The sampling frequency used to extract the data from the news feed needs to reflect the rapidity in which things change in the area. Business and financial events happen much more frequently than changes in technology. One might set up the ETD tool to collect data each day and then process it looking for new trends. Since developments in science and technology happen at a slower pace, it might be appropriate to work in blocks of one week or one month. Processing information in one year blocks may be ok for a retrospective look at what has happened, but is not acceptable for most decision making situations.

ETD systems will have to be personalized to meet the needs of specific clients. The ETD algorithms will eventually become quite good at determining that something has happened. However, whether or not the development is significant depends on the needs of the person looking at the data. Broadly speaking, there are two types of client profile information that must be obtained. The first is the definition of the area of interest and the second is the definition of what characterizes an important event in the area. Traditional alerting systems handled the first problem by setting up a user profile containing a search query that was run on a periodic basis against specified data sources. Typically, the query was built on a trial and error basis by the information professional and the client. This was an iterative process. Some of the newer knowledge management systems use document training sets to build the "query". Which approach will work best with a given ETD tool remains to be seen. Either process can take a significant amount of time for the client who wants the information. There is also the problem that the client's areas of interest will expand and change over time. Each time this happens, a new profile will have to be generated.

The problem of determining what is a significant event for a given user is handled in interactive systems by letting the decision maker work directly with the tool. If the decision maker is unwilling to work directly with the tool, then an analyst will have to interview the decision maker and get basic guidelines to work with. He can then do the analysis and give the decision maker a list of potentially significant events from which to choose from. It would be best if these suggested developments where presented in a browser format that would allow the decision maker to drill down into the detail underlying any development he found to be of interest.

It is too early to tell what the cost of an ETD software system will be. If it turns out to be in the same class as the knowledge management and text database mining software of today, it will cost ten's to hundred's of dollars. Assuming it will be a software product, it probably will carry the same sort of structure as the high end software packages available today. The vendors charge an initial purchase cost and then require an annual maintenance fee to provide technical support and updates of the software. Sometimes it is possible to buy the software by the seat, but often the vendors push to sell a corporate license. If only a few people will be using the software, then purchasing seats makes sense. If the software is actually going to be used across the enterprise, then a corporate

license is probably the better choice. Another cost that is often overlooked is the impact on the corporate IT infrastructure. Even when the application is run on existing in-house servers, there is usually the need to have a system administrator and possibly a database administrator available to keep the application up and running.

Sometimes it is useful to develop a perfect world scenario to better understand what needs to be done in an evolving application area. Table 15 shows the characteristics of what I think would be the perfect world information system would be for a busy executive who is involved in decision making processes. My assumptions are that ETD tools would be part of this system and that system I specify would also meet the needs of an analyst or individual contributor.

• Raw data processed into useful information • Sufficient information presented to meet current need • No irrelevant information prevented • All information available immediately when needed • Information prioritized for the current need • Information presented in a format that is intuitively easy to understand • Information can be viewed at different levels of detail • Information can be viewed from multiple perspectives

Table 15: Perfect World Executive Information System

The perfect world scenario can be summarized to say that the decision maker wants to see the unique information he should see, not see any irrelevant or redundant information, and have the information in a form that is easy to understand and interpret. ETD tools need to fit in to this scenario if they are really going to get used at high levels of a corporation.

There continues to be good progress made in knowledge management and text mining tools. Since any ETD system will make use of these types of tools, I think there is a good possibility the practical ETD systems will eventually become available for fixed information needs. Building a system that will keep up with a given decision maker's changing information needs will be difficult, unless a good way is found to automatically translate the subject areas of interest and the important developments criteria from the words of the user to the ETD system. There will always be a challenge to make sure that data sources available are adequate to support the needs of the decision maker.

In this section I have reviewed some to the practical aspects of working with an ETD tool in a corporate environment. The real test for an ETD system is whether or not it provides useful information about new developments to the decision maker in a timely manner. The current systems do not seem to provide

the required level of performance to be used extensively in the corporate environment. There is hope that new generations of ETD tools will be useful to corporate decision makers.

6. ACKNOWLEDGMENTS

7. REFERENCES

1. R. Agrawal, R. Srikant: "Mining Sequential Patterns", *Proc. of the Int'l Conference on Data Engineering (ICDE)*, Taipei, Taiwan, March 1995.
2. R. Agrawal, G. Psaila, E. L. Wimmers, M. Zait: "Querying Shapes of Histories", *Proc. of the 21st Int'l Conference on Very Large Databases*, Zurich, Switzerland, September 1995.
3. J. Allan, L. Ballesteros, J. Callan, W. Croft, and Z. Lu. Recent experiments with inquery. In *The Fourth Text Retrieval Conference (TREC-4)*, pages 49–63, 1995.
4. J. Allan, R. Papka, V. Lavrenko, "On-line New Event Detection and Tracking," Proceedings of ACM SIGIR, pp.37-45, 1998.
5. Applied Semantics: http://www.appliedsemantics.com/ June, 2002.
6. Autonomy: http://www.autonomy.com/ June, 2002.
7. Bader, R., M., Callahan, D. Grim, J. Krause, N. Miller and W. M. Pottenger. The Role of the HDDI™ Collection Builder in Hierarchical Distributed Dynamic Indexing. *Proceedings of the Textmine '01 Workshop, First SIAM International Conference on Data Mining*. April 2001.
8. Banter: http://www.banter.com/ June, 2002.
9. D. Bikel, S. Miller, R. Schwartz, and R. Weischedel. Nymble: a high-performance learning name-finder. In *Fifth Conference on Applied Natural Language Processing*, pages 194–201. ACL, 1997.
10. G. D. Blank, William M. Pottenger, G. D. Kessler, M. Herr, H. Jaffe, S. Roy, D. Gevry, Q. Wang. CIMEL: Constructive, collaborative Inquiry-based Multimedia E-Learning. *The 6th Annual Conference on Innovation and Technology in Computer Science Education* (ITiCSE), June 2001.
11. Fabien Bouskila, William M. Pottenger. The Role of Semantic Locality in Hierarchical Distributed Dynamic Indexing. *In Proceedings of the 2000 International Conference on Artificial Intelligence (IC-AI 2000)*, Las Vegas, Nevada, June, 2000.
12. Captiva: http://www.captivacorp.com/ June, 2002.
13. Chaomei Chen, Les Carr: A Semantic-Centric Approach to Information Visualization. IV 1999: 18-23
14. Chaomei Chen, Ray J. Paul: Visualizing a Knowledge Domain's Intellectual Structure. IEEE Computer 34(2): 65-71 (2001)
15. H. Chen and K.J. Lynch, *Automatic Construction of Networks of Concepts Characterizing Document Databases*, IEEE Transaction on Systems, Man and Cybernetics, vol. 22, pp. 885-902, 1992.
16. ClearForest: http://www.clearforest.com/ June, 2002.

17. Clementine: http://www.spss.com/spssbi/clementine/ June, 2002.
18. compendex: http://edina.ac.uk/compendex/ June, 2002.
19. Davidson, G. S., Hendrickson, B., Johnson, D. K., Meyers, C. E., & Wylie, B. N. Knowledge mining with VxInsight™: Discovery through interaction. *Journal of Intelligent Information Systems*, 11(3), 259-285.
20. Delphion: http://www.delphion.com/
21. E. Edgington. *Randomization Tests*. Marcel Dekker, New York, NY, 1995.
22. Factiva: http://www.factiva.com/ June, 2002.
23. *Facts on File, 1996*, Facts on File, New York, 1997.
24. Ronen Feldman and Ido Dagan. *Knowledge discovery in textual databases* (kdt). In Proceedings of the First International Conference on Knowledge Discovery (KDD-95). ACM, August 1995.
25. D. Fisher, S. Soderland, J. McCarthy, F. Feng, and W. Lehnert. Description of the umass systems as used for muc-6. In *Proceedings of the 6th Message Understanding Conference, November, 1995*, pages 127–140, 1996.
26. Gartnerg2: http://www.gartnerg2.com/site/default.asp June, 2002.
27. D. Gevry, *Detection of Emerging Trends: Automation of Domain Expert Practices.* M.S. Thesis, Department of Computer Science and Engineering at Lehigh University, 2002.
28. Havre, Susan, Elizabeth G. Hetzler, Lucy T. Nowell: ThemeRiver: Visualizing Theme Changes over Time. INFOVIS 2000: 115-124
29. S. Havre, E. Hetzler, P. Whitney, and L. Nowell. *ThemeRiver: Visualizing Thematic Changes in Large Document Collections*, IEEE Transactions on Visualization and Computer Graphics, vol. 8, no. 1, Jan-Mar, 2002.
30. HyBrix : http://www.siemens.com/index.jsp June, 2002.
31. IDC: http://www.idc.com June, 2002.
32. Inspec: http://www.iee.org.uk/Publish/INSPEC/ June, 2002.
33. Interwoven: http://www.interwoven.com/products/ June, 2002.
34. Invention Machine: http://www.invention-machine.com/index.htm June, 2002.
35. Yumi Jin's thesis
36. V. Lavrenko, M. Schmill, D. Lawrie, P. Ogilvie, D. Jensen, J. Allan, "Mining of Concurrent Text and Time-Series," KDD-http://citeseer.nj.nec.com/378125.html
37. LDC: http://www.ldc.upenn.edu/ June, 2002.
38. Lent, B.; Agrawal, R.; and Srikant, R. 1997. Discovering Trends in Text Databases, Proc. 3 rd Int Conf. On Knowledge Discovery and Data Mining, California.
39. Anton Leuski, James Allan: Strategy-based interactive cluster visualization for information retrieval. Int. J. on Digital Libraries 3(2): 170-184 (2000)
40. A. Leuski and J. Allan. Lighthouse: Showing the way to relevant information. In *Proceedings of the IEEE Symposium on Information Visualization (InfoVis*), pages 125–130, 2000.
41. lexiquest: http://www.spss.com/spssbi/lexiquest/ June, 2002.
42. LexisNexis: http://www.lexisnexis.com/ June, 2002.
43. Loet Leydesdorff Indicators of Structural Change in the Dynamics of Science: Entropy Statistics of the *SCI Journal Citation Reports*, *Scientometrics* 53(1), 131-159

44. A. Martin, T. K. G. Doddington, M. Ordowski, and M. Przybocki. The DET curve in assessment of detection task performance. *In Proceedings of EuroSpeech'97*, volume 4, pages 1895–1898, 1997.
45. Moreover: http://www.moreover.com/ June, 2002.
46. Northern Light: http://www.northernlight.com/ June, 2002.
47. L. T Nowell, R. K France, D. Hix, L. S Heath and E. A Fox. Visualizing Search Results: Some Alternatives to Query-Document Similarity. In *Proceedings of SIGIR'96*, Zurich, 1996
48. C. Plaisant, R. Mushlin, A. Snyder, J. Li, D. Heller, and B. Shneiderman. Lifelines: Using visualization to enhance navigation and analysis of patient records. In Proceedings of the 1998 American Medical Informatic Association Annual Fall Symposium, pages 76-80, 1998.
49. A. Popescul, G. W. Flake, S. Lawrence, L. Ungar, and C. L. Giles. Clustering and identifying temporal trends in document databases. In Proc. IEEE Advances in Digital Libraries 2000, 2000.
50. Porter, A.L., and Detampel, M.J., "Technology Opportunities Analysis," Technological Forecasting and Social Change, Vol. 49, p. 237-255, 1995.
51. Allan L. Porter and Donghua Jhu. Technological Mapping for management of technology. Presented at International Symposium on Technology 2001. www.tpac.gatech.edu/~donghua/nanotechnology/isf2001_dhz_alp_06_14.ppt
52. W. M. Pottenger. *Theory, Techniques, and Experiments in Solving Recurrences in Computer Programs*, Ph.D. thesis, University of Illinois at Urbana-Champaign, Center for Supercomputing Res. & Dev, 1997.
53. Pottenger, William M. and Ting-hao Yang. Detecting Emerging Concepts in Textual Data Mining. *In Computational Information Retrieval, Michael Berry, Ed., SIAM*, Philadelphia, PA, August 2001.
54. Pottenger, William M., Yong-Bin Kim and Daryl D. Meling. HDDI™: Hierarchical Distributed Dynamic Indexing. *In Data Mining for Scientific and Engineering Applications*, Robert Grossman, Chandrika Kamath, Vipin Kumar and Raju Namburu, Eds., Kluwer Academic Publishers, July 2001.
55. Quiver: http://www.quiver.com/ June, 2002.
56. S. Roy. *A Multimedia Interface for Emerging Trend Detection in Inquiry-based Learning*. M.S. Thesis, Department of Computer Science and Engineering at Lehigh University, 2002.
57. Soma Roy, David Gevry, William Pottenger. Methodologies for Trend Detection in Textual Data Mining. *Proceedings of the Textmine '02 Workshop, Second SIAM International Conference on Data Mining*. April, 2002.
58. Semio: http://www.semio.com/ June, 2002.
59. Ser Solutions: http://www.sersolutions.com/ June, 2002.
60. SPSS: http://www.spss.com/PDFs/LQMBRO-0302.pdf June, 2002.
61. Srikant, R., & Agrawal, R. (1996). Mining sequential patterns: Generalizations and performance improvements, . Proc. of the Fifth Int'l Conference on Extending Database Technology (EDBT). Avignon, France.
62. Stratify: http://www.stratify.com/ June, 2002.
63. R. Swan and J. Allan. Automatic generation of overview timelines. *In Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (these proceedings)*, Athens, Greece, 2000. Association for Computing Machinery.

64. Russel Swan and David Jensen. TimeMines: Constructing Timelines with Statistical Models of Word Usage. In *the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2000.
65. Tacit: http://www.tacit.com/ June, 2002.
66. R. E. Tarjan, Depth first search and linear graph algorithms, *SIAM Journal of Computing*, 1:146–160, 1972.
67. TDT: http://www.nist.gov/speech/tests/tdt/index.htm. June, 2002.
68. textAnalyst: http://www.megaputer.com/products/ta/index.php3
69. ThoughtShare: http://www.thoughtshare.com/. June, 2002.
70. UIUC: "UIUC Digital Library Initiative", http://dli.grainger.uiuc.edu/.
71. USPTO: http://www.uspto.gov/main/patents.htm. June, 2002.
72. Verity: http://www.verity.com/. June, 2002
73. Pak Chung Wong, Wendy Cowley, Harlan Foote, Elizabeth Jurus, Jim Thomas. Visualizing Sequential Patterns for Text Mining. Pacific Northwest National Laboratory. *In Proceedings of IEEE Information Visualization* 2000, October 2000.
74. Jinxi Xu, J. Broglio, and W. B. Croft. The design and implementation of a part of speech tagger for English. Technical Report IR-52, Center for Intelligent Information Retrieval, University of Massachusetts, Amherst, 1994.
75. *T. Yang. Detecting Emerging Conceptual Contexts in Textual Collections.* M.S. Thesis, Department of Computer Science at the University of Illinois at Urbana-Champaign, 2000.
76. Yiming Yang, Tom Pierce and Jaime Carbonell. A study on Retrospective and On-Line Event Detection. Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval.