The SITAC Approach for Time-Aware Query Translation in Text Archives

Abstract

With an exponential growth in archival of time-stamped documents such as newswire articles, blog posts and other web-pages, information retrieval (IR) has become a challenging task. The degree of complexity in this IR task increases when these archives cover long time-spans and the terminology in them has undergone significant changes. When users pose queries pertaining to historical information over such document collections, the queries need to be translated, incorporating temporal changes, to provide accurate responses. For example, a query on Sri Lanka should automatically retrieve documents with its former name Ceylon. We call such concepts SITACs i.e., Semantically Identical Temporally Altering Concepts. To discover SITACs from a given corpus, we propose a methodology which integrates natural language processing, association rule mining, and contextual similarity. By using the SITACs discovered, historical queries over text corpora can be addressed effectively. Proposed methodology was experimented with Gutenberg corpus which contains speeches of American presidents since first speech of Mr. George Washington in 1795 to speech of Mr. George W. Bush in 2006. Search engines and IR systems can be benefited by the techniques we provide in this research.

MONTCLAIR STATE UNIVERSITY

THE SITAC Approach for Time-Aware Query Translation in Text Archives

by
Amal Kaluarachchi
A Master's Thesis Submitted to the Faculty of
Montclair State University
In Partial Fulfillment of the Requirements
For the Degree of
Master of Science
May, 2010

School College of Science and Mathematics
Department Computer Science

Thesis Committee:

_____
Dr. Aparna Varde
Thesis Sponsor

Certified by:

_____
Dr. Robert Prezant
(Dean)

_____
Date

_____
Dr Anna Feldman
Committee Member

_____
Dr. Jing Peng
Committee Member

_____
Dr. Dorothy Deremer
Department Chair

THE SITAC APPROACH FOR TIME-AWARE QUERY TRANSLATION IN TEXT ARCHIVES

A THESIS

Submitted in partial fulfillment of the requirements for the degree of Master of Science in the Department of Computer Science in the Graduate Program of

by

AMAL KALUARACHCHI

Montclair State University

May 2010

Acknowledgements

I very respectfully acknowledge my advisor Dr. Aparna Varde for all her guidance and advice at every step of the way, for patiently listening to me for uncountable hours, for advocating me how to perform a research, for imparting so much valuable knowledge and for all her encouragement and words of kindness. Any noteworthy aspects of this thesis are entirely because of her.

I am also very grateful to Dr. Jing Peng and Dr. Anna Feldman who not only agreed to serve on my masters' thesis committee but also spent a lot of their time to read and comment on the thesis in excruciating detail. Thank you for your very valuable advice and for the interest you have shown.

I would like to acknowledge the help of the Department of Computer Science at University of Montclair; all the faculty professors taught me in various subjects which helped me in this thesis. Specifically I would like to thank Dr Dorothy Deremer and Dr. James Benham for their help with administration work related to this thesis.

Finally I would like to thank my wife Loyala and my son Timindu for their support.

This research started when my advisor Dr. Aparna Varde was a Visiting Senior Researcher at the Max Planck Institute for Informatics, Saarbrucken, Germany in the summer of 2008. She worked in the Databases and Information Systems Group headed by Dr. Gerhard Weikum with whose guidance this project was initiated. We gratefully acknowledge their inputs in this work.

Table of Contents

List of Figures

1. Introduction

1.1 Background and Motivation

Time-stamped documents such as newswire articles, blog posts and other web-pages are often archived online. Large Internet Archives (e.g.: archive.org) have billions of articles. London Times archive contains documents since 18th century. As a consequence, when these archives cover long spans of time, the terminology within them could undergo significant changes. Hence, when users pose queries pertaining to historical information, over such documents, the queries need to be translated, taking into account these temporal changes, to provide accurate responses to users. We present a few motivating examples of such queries.

1. How many states were in the USA at the time of independence?

2. When was the constitution of the USA established?

3. What has been the USA policy on the UK over 200 years?

These queries would be entered on a search engine using appropriate keywords or sentences. In response to these queries, multiple text documents need to be referenced. An example of a document with answers to queries 1 and 2 is Lincoln's presidential address [2]. However, he refers to the USA as the Union. Query 3 can be answered from speeches of many presidents [2] who use terms such as British Isles and Great Britain in referring to the UK. For example, if this query is executed on a search engine such as Google, the results contain documents containing the terms USA, UK or 200 years. Unless there is a document worded similarly to USA foreign policy we do not get the

exact answer even though that information is available in some articles. The problem addressed in this thesis, therefore goes beyond the semantic aspects of queries. Semantic aspects of user queries are addressed in WordNet [19]. We add one more piece, i.e., the temporal factor along with semantics. It is also to be noted that this is not just an issue of synonymy, e.g., the terms USA and Union would not be detected in the literature as obvious synonyms. However, from a study of history, it is known that when the former presidents referred to the Union specifically during Civil Wars, they meant the United States of America or the USA, as it is called today.

Moreover, users tend to formulate queries with current terminologies. Reviewing Google Trends TM we can see the name "*Sri Lanka*" has been mainly used when searching documents related to "*Sri Lanka*". But there are many relevant documents available with its former name "*Ceylon*". Any article related to Sri Lanka, generated before 1948 had no information about name "*Sri Lanka*". Though query expansion and refinement techniques are currently being employed in information retrieval systems, it does not inject semantic concepts to user queries. For an example, query "*President of Canada*" automatically translated to "*Prime Minister of Canada*". These techniques primarily speak to the correlations between words. Temporal changes of concepts we describe here are yet to be explored.

Some previous research can be found in this context. Some work is done using frequency based methods while other work uses graph/network methods. In our research, the focus is on finding a method to discover SITACs from a given text corpus using linguistic

properties of concepts (i.e. verb, noun, subject, object, etc). Concepts involved in this context will have two categories; 1) concepts in anticipated queries (supervised approach) 2) concepts of any generic query (unsupervised approach), datasets will be pre-processed to handle them.

1.2 Problem Definition

This research addresses two problems in IR:

1. Finding concepts which are changed over time (SITACs);

2. Incorporate SITAC knowledge in responding to user queries.

The definition of the term SITAC is given below.

"*The term SITAC is an acronym for a Semantically Identical Temporally Altering Concept. It refers to a concept whose names change over time, although they in principle refer to the same entity. SITACs could be under different categories such as persons names, places, organizations, item names and more."*

Examples of SITACs include:

- *Person:*
    - *Hillary Clinton; Hillary Rodham@1974; First Lady@1993-2001*
    - *Mohandas Karamchand Gandhi; Mahathma Gandhi*
- *Place:*
    - *Sri Lanka; Ceylon @ 1972*
    - *United States; Union during civil war*
    - *Zimbabwe; Rhodesia*

- *St Petersburg; Leningrad*

- *Organization:*
  - *Virginia Agricultural and Mechanical College and Polytechnic Institute; Virginia Polytechnic Institute; Virginia Polytechnic Institute and State University (Virginia Tech)*
  - *AT&T; Lucent Technologies; Alcatel-Lucent (the parent organizations of Bell Labs)*

The goal of this work is that queries involving SITACs should be translated and answered appropriately, e.g., "When did Sri Lanka get its independence?" (Sri Lanka received independence in 1948 when it was called Ceylon. In providing the answer, system must have this knowledge.) It is also required to develop a method to obtain such SITACs from large time-spanned corpora.

This problem can be encountered with information retrieval systems and web search engines. The real dilemma for most of the Web search engines is that users request for accurate search results, but search engines only provide results based on the term frequencies of the words in input query. By entering queries "When did *Sri Lanka* get its independence?", "When did *Ceylon* gets its independence" and "When did *Sri Lanka Ceylon* gets independence" to verity of search engines, we can see how varied results will be. Obviously, users can enter first and second queries, but third query is different. By experimenting such kind of queries (3^{rd} query), we found that it produced better results than first and second.

1.3 Layout and Organization

The rest of the thesis is organized as follows. Section 2 explains our proposed approach to discover the SITACs and incorporate them in responding to user queries. Section 3 describes the implementation and experimental evaluation of our approach. Section 4 presents an overview of related work in the area. Section 5 gives the conclusions.

2. Proposed Approach: SITAC

2.1 General Description

In order to discover SITACs in text archives for time aware query translation, we propose an approach by the same name: SITAC, which is an integrated framework of natural language processing, association rule mining, and contextual similarity. We present a general description of the SITAC approach followed by a discussion of its steps.

In order to explain the SITAC approach, we consider following two example sentences.

1. Sri Lanka hangs like a jewel off India's tip, surrounded by the Indian Ocean.
2. Ceylon is called the pearl of the Indian Ocean by explorers who came to Asia in 15th and 16th centuries. (Source: Internet)

Upon reading these two sentences, the human brain can identify that Sri Lanka and Ceylon have something in common. That judgment is made by understanding common words associated with Sri Lanka and Ceylon. In this research, we simulated that human judgment of identifying such concepts by using computational methods. First we came up with a heuristic which is "*if one or more concepts (nouns) are referred by similar events (verbs) those concepts are semantically related".* To improve that logic, we also used objects, adjectives, bigrams and omplimentizers (comp) along with verbs.

In our research, the focus on finding a method to discover SITACs from a given text corpus using linguistic properties of concepts (i.e. verb, noun, subject, object, etc.),

putting them in data sets with different matrices followed by extensive data mining tasks. This general problem has two categories:

1) Concepts in anticipated queries (supervised approach). Anticipated queries are some common queries that a user will execute on a given corpus. Those queries are used to guide all solution approaches, thus making it supervised. One drawback of the supervised method is that it can easily miss important SITACs if they don't appear in anticipated queries.

2) Concepts in any generic query (unsupervised approach). In this situation, no preconceived queries are in mind, thus making it unsupervised. The related concepts are derived directly from raw data which is the text corpus. Thus, there is no risk of missing important SITACs.

Association rule mining was the primary focus of the proposed solution because we try to simulate the manner in which humans mentally associate concepts that evolve over time. Other machine learning techniques such as clustering and classification were also experimented; however due to the nature of the knowledge we were trying to discover, it was found that association rule produces the best results. One general problem in mining association rules is the selection of interesting association rules within the overall and possibly huge set of extracted rules. It is also required to set the minimum thresholds because of the frequencies of such associations are lower in text corpora. We overcome this by introducing a ranking mechanism.

Once the rules are identified, similarity and correlation measurements such as Jaccard's coefficient are used to rank the rules. The proposed solution in this research thus includes the following challenges:

1. Discovering knowledge in the form of SITACs in text archives with the following subtasks:

    a. Preparing datasets from text archives, a challenging task involving linguistics, especially the appropriate harnessing of natural language processing techniques;

    b. Defining adequate transactions to capture the essence of the problem and deriving association rules on the generated data sets, which is also a non-trivial task.

2. Answering user queries in an intelligent manner using temporal knowledge of concepts obtained on discovering SITACs, thereby developing a query translation engine.

3. Ranking the responses to user queries appropriately incorporating factors that human users would consider.

Prior work in this area has been done by our colleagues [4,6] assuming anticipated queries, found suitable at the initial stage of the research. Hence we do not go into description of the solution using anticipated queries.

Our further research covers the anticipated queries as well as general queries. We also work on implementation of the SITAC approach to develop a query translation engine for resolving time-aware queries. Our aim is to discover rules of the type (C1, T1) => (C2, T2) where C1 and C2 are concepts and T1 and T2 are corresponding time stamps for C1 and C2. The basic logic proposed in this work is explained in the following subsections.

2.2 Concept Extraction

In this step, text archives are processed to extract information as concepts in documents over time, related by events. So, we have:

- *Document:* Text source *D* with time-stamp *T*
- *Concept:* Individual term *C* (word or phrase)
- *Event :* The event *E* relating concepts
- *Other words in the neighborhood :* The other words linguistically connected to the concepts (as object, adjective, omplimentizer, bigrams and so on)

The concepts are primarily nouns and noun phrases referring to entities such as persons and places. Events correspond to verbs referred by concepts. This involves natural language processing exploiting semantic features from a linguistic perspective.

2.3 Rule Derivation

This step discovers rules of the type (C1,T1)=>(C2,T2) from the corresponding time-stamped documents. We use the classical Apriori Algorithm [1] to mine the association

rules, for which we define transactions with respect to the text archives. Our transaction set is built based on a logic using linguistic properties such as subject, object, noun and verb. If an event is referred to by two distinct nouns, and such events occurring multiple times, then we consider that those two nouns (concepts) are related. We extend that logic to other words in the neighborhood of concepts. This can be further explained using set theories.

Consider the sets Events *{E1,E2,E3..,En}* and Concepts with Time stamps *{(C1,T1), (C2,T2)…,(Cp,Tq).* If *(Ci,Tj)* and *(Cx,Ty)* are referred by *Er,* a distance value *r* is assigned to that relationship. Initially every pair of *C,T* has *r* = infinity (very high). Each appearance of *(Ci,Tj)* and *(Cx,Ty)* together in a relationship decrements the value of *r.* The pairs *(Cx,Cy)* with smallest *r* values are considered as SITACs. We thus obtain rules of the (Cx,Tx) => (Cy,Ty)

A transaction defined for the purpose of association rule mining in this problem consists of two or more concepts (as identified by nouns) {C1, C2 … Cn} that are referred to by any common event E occurring (as identified by verbs). Based on this linguistic relationship of concepts that we propose in this research, the following data sets will be generated from the text archive.

a. {EVENT, TIME1,TIME2….,,TIMEn}

TIME1.. TIMEn will have concepts that appeared in the text archive associated with the events listed under the EVENT attribute

b. {CONCEPT, TIME1,TIME2….. TIMEn}

Table1 shows the transaction set in tabular form. Cx = Concepts

Cx := {C1.. Cp}

Event	T1	T2	T3	T4	..	Tn
Verb	Cx	Cx	Cx	Cx	Cx	Cx
Adjective and other Relations	Cx	Cx	Cx	Cx	Cx	Cx

Transaction set

2.4 Ranking of Rules

Once the SITACs are discovered, this part of the system finds how strongly they are related. Thus, it serves to give an order-based measurement to the temporal relationships captured by SITACs. Among many similarity methods used in IR, (i.e. Dice Coefficient, Weighted Sum, Cosine value, Euclid distance, etc), Jaccard's coefficient is used because it is found to be useful in capturing contextual similarity as per the study of the literature e.g., [12]. In our problem, we deploy this as follows. For two relationships, *R1{(Cx,Ts), (Cy,Tt) }, R2{(Cx,Tt) , (Cz,Tu) }*, we count other words (nouns, verbs, adjectives, etc) were used with *Cx,Cy* and *Cz.* As per Jaccard's coefficient, we calculate score for similarity J as:

$J(Cx,Cy) = (Cx \cap Cy) / (Cx \cup Cy)$

$J(Cx,Cz) = (Cx \cap Cz) / (Cx \cup Cz)$ **……** and so forth.

Now, we argue that $JS(Cx,Cz) > JS(Cx,Cy)$ means *Cx* is more related to *Cz* than *Cy* based on adapting the definition of Jaccards' coefficient in this context. This logic is used to rank the SITACs.

2.5 Query Translation

The SITACs discovered are then used to perform query translation as follows. SITACs have been filtered by ranking and are stored in database along with some additional linguistic knowledge incorporating all parts of the speech with their time-stamps acquired during text parsing. The following piece of SQL code shows the example of the storage.

Tbl_word : {year, word}

Tbl_SITAC : {word, year, SITAC}

When a user enters a query, it goes through a parser and stores all words in an array after eliminating stop words (the, an) and common words (I, We). The user query (W1, W2,W3.., Wn) is then translated to a nested SQL query as follows:

```
/* Find SITAC * /
SELECT
     *  FROM Tbl_SITAC
     WHERE  WORD  = <W1>  AND SELECT
/*Find Documents */
SELECT
    * FROM Tbl_word
    where word =<W1> AND
        year in (SELECT year from Tbl_word where word=<W2> AND
```

year in (SELECT year from Tbl_word where word=<W3> AND

year in (SELECT year from Tbl_word where word=<Wn>)))

Combining results of two queries, documents can be appropriately retrieved from a given corpus.

2.6 Algorithms in the SITAC Approach

Based on our explanation given so far, the two algorithms we propose below summarize all the steps involved in finding SITACs and utilizing them effectively in responding to user queries.

Algorithm1: Discovering and Rankling SITACs from Text Corpora

Input: Text Corpus with time-stamped documents

D[i]= {D1,D2,..Dn} // Documents separated based on the year
For *i = 1* to *n* {
  Run parser on *S[i] -> F[i]* // Text files after parsing }
For *i = 1* to *n* { // Generate each instance from parsed files
  E=" " ; C=" " // *E*=Event *C*=Concept
  While (not End_of_File *F[i])* {
    If (*F[i]*.Readline() contains token *"V:"* { //*V*=Verb, *N*=Noun
      E=word before *"V:"* token, *C*=word next to *"N:"* token
      P=P+ {i,E,C} // *P*=Processed dataset: rows, columns }}}
Transpose *P* with instance *i* as columns and *E C* as rows
Run Apriori association rule mining algorithm
For-each transaction ( )
  Use relationship distance *r* to get time-stamped SITAC pairs}
For-each SITAC pair { // *J* = Jaccard's coefficient based score
  J(Cx,Cy) = (Cx ∩ Cy) / (Cx U Cy)
  J(Cx,Cz) = (Cx ∩ Cz) / (Cx U Cz)
  J(Cx,Cz) > J(Cx,Cy) means *Cx* is more related to *Cz* than *Cy }*
Rank SITACs using *JS* values

Output: Ranked SITACs stored in SITAC Database

Algorithm2: Using SITACs for Translating and Answering User Queries

Input: user query

```
Q={W1,W2,W3,W4…Wn}  //Q is the user query
Parse Q and remove stop words
For each Q{Wn}
{
  Search word = w1 in SITAC database
  Q = Q + SITAC.SITAC     // include SITAC in the user query
   year = SITAC.year        //  Store year of SITAC
}

For each Q{Wn}
{
        SELECT year form Tbl_word where exists word ….. // SELECT statement to find
words
}
```

Output : The list of documents that contain words from user query

3. Implementation of the SITAC System

Given the SITAC approach we have proposed, we have implemented a system also by the same name SITAC that uses this approach for time-aware query translation in text archives. We describe the system architecture and development in the next subsections.

3.1 System Architecture

As shown in Figure [1] there are 4 core components in SITAC system. Each component produces an output and passes it to the next level.

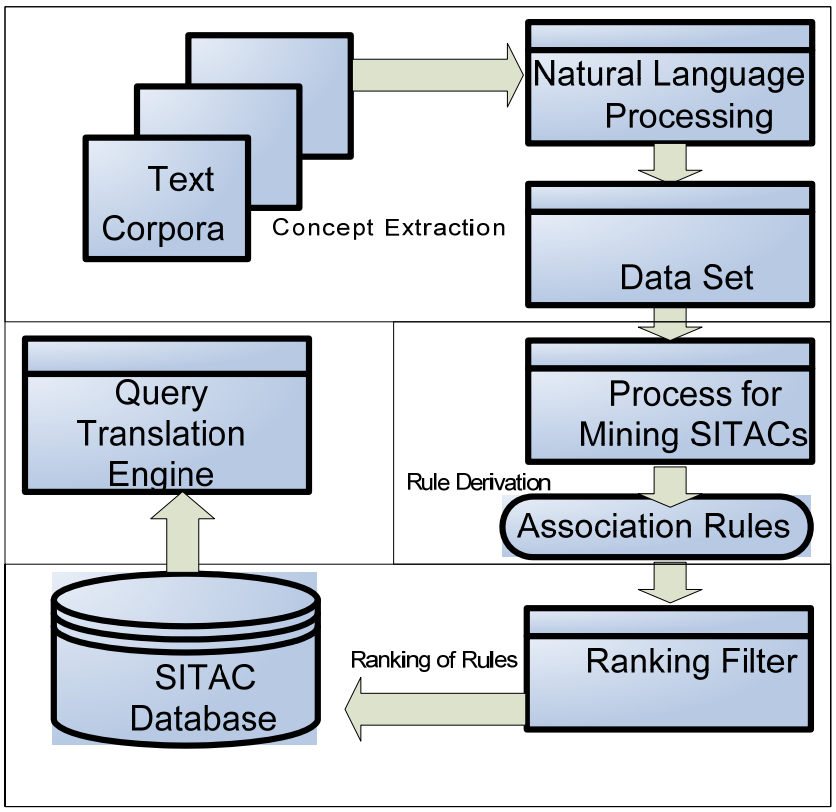

Figure 1 : SITAC System Architecture

Concept extraction process takes text corpora as an input and produces the dataset for association rule mining. Some NLP tasks are involved in this process. We have developed the functionality for automating this process.

Rule derivation is performed with the dataset created from the concept extraction step. This process is implemented using the Apriori algorithm with code reuse from the WEKA data mining tool, appropriately adapted in this context.

Rules such as (Cx,Tx) => (Cy,Ty) derived through the data mining tool are fed to Ranking filter to evaluate the relevancy between concepts Cx and Cy. The ranked rules are stored in SITAC database.

The SITAC database is used by Query translation engine when answering to user queries.

3.2 System Development

Our SITAC system has been developed on a Windows XP system using Java for the implementation of the SITAC approach with MySQL for database tasks, using text corpora spanning large periods of time, we first separated them into individual documents based on time-stamps. Then using natural language processing, we parsed the documents, storing all nouns, verbs, types of noun (subject or object) , adjectives, compliments of words in a database. We used Minipar for parsing, developing a Java program to store unformatted large text corpora in a processed database with attributes: year, verb, noun, subject, object, adjective and comp word. We then created a dataset of transactions and sent it to the WEKA data mining tool to mine SITACs using the Apriori algorithm. This was followed by ranking.

The user interface for discovering SITAC is shown in the figure below:

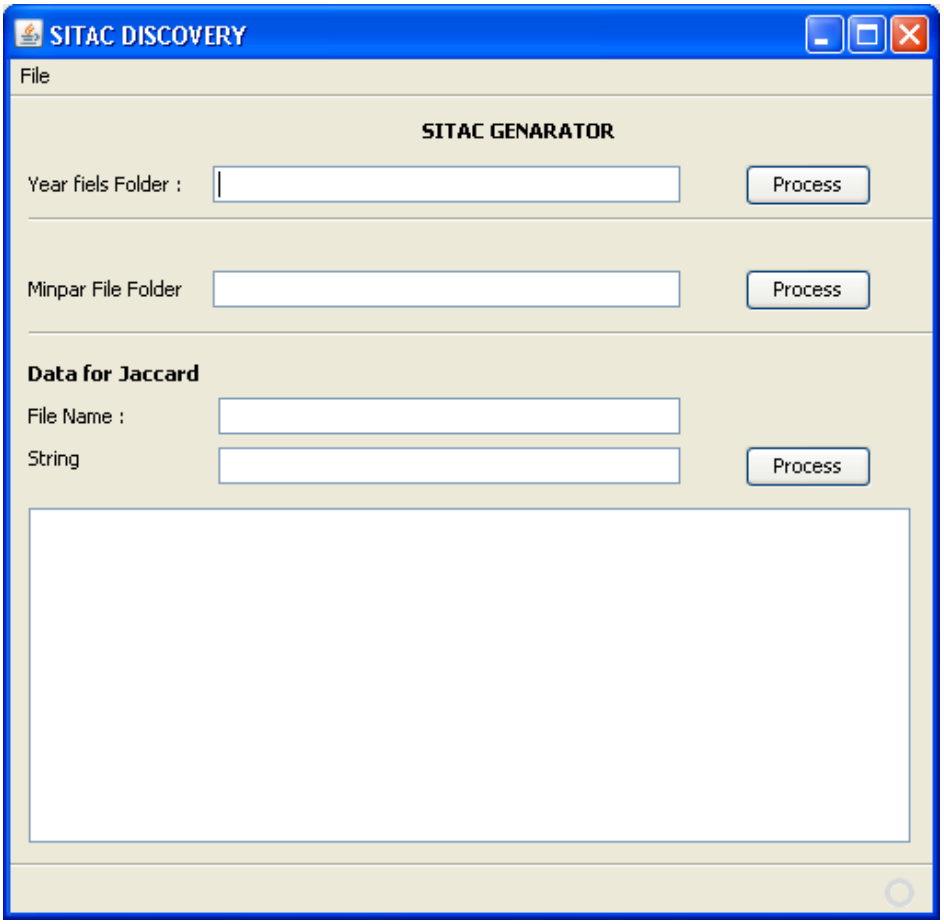

Figure 2: The SITAC User Interface

There are three sections in this interface.

1. Year Files Folder : By giving folder name which contains year files, the system feeds each file in the folder to Minipar application. Year files are sub files separated from the main corpus. This process creates a shell script which can be used with Minipar parser.

   A sample of the script is given below.

   #!/bin/sh

   print Minipar Parser Execution

pdemo -l 1790.txt

pdemo -l 1791.txt

pdemo -l 1792.txt

pdemo -l 1793.txt

pdemo -l 1794.txt

pdemo -l 1795.txt

Creating year files is a combination of auto and manual process. No common methodology is experimented in this regard because of the presentation of the corpus can be varied from one to another.

2. Minipar File Folder : This process is used to create the data set as described in section 2.2 rule derivation. System reads Minipar parsed files and based on their linguistic tokens, it creates a dataset figure [4].

3. Processing data for Jaccard's : By giving the corpus name and string to search, the system scans the entire corpus and produces the list of words in the neighborhood of a given string, along with their frequencies.

4. Experimental Evaluation

The SITAC system has been subject to experimental evaluation using real text archives in order to assess its effectiveness. We present the details of our experimentation here.

4.1 Experiments and Observations

The text source for experiments shown here is the Gutenberg corpus [2] of the USA Presidents' speeches from 1790 to 2006. This single large corpus consists of 209 speeches required to be separated in order to feed in to next step of parsing. It was done by the year files option using the Java program we developed. We present the stepwise execution of the experiments along with the following observations.

Those separated documents were fed to Minipar and obtained parsed documents as figure [3].

```
> fin   C:i:V   Trust
Trust   V:s:N   I
Trust   V:subj:N        I
Trust   V:fc:C  fin
fin     C:i:V   deceive
deceive V:s:N   I
deceive V:aux:Aux       do
do      Aux:neg:A       not
deceive V:subj:N        I
deceive V:obj:N myself
deceive V:mod:C fin
fin     C:wha:A when
fin     C:i:V   indulge
indulge V:s:N   I
indulge V:subj:N        I
indulge V:obj:N persuasion
persuasion      N:det:Det       the
fin     C:c:COMP        that
fin     C:i:V   meet
meet    V:s:N   I
meet    V:have:have     have
meet    V:amod:A        never
meet    V:subj:N        I
meet    V:obj:N you
meet    V:mod:Prep      at
at      Prep:pcomp-n:N  period
period  N:det:Det       any
meet    V:mod:C fin
fin     C:wha:A when
fin     C:mod:A more than
more than       A:lex-mod:U     more
more than       A:mod:Prep      at
at      Prep:pcomp-n:N  present
present N:det:Det       the
```

Figure 3: Partial dump of Minipar output

Parsed data is primarily stored in a Table (Figure [4]) and later it is transformed to the table used for WEKA AR Mining.

year	verb	noun	obj	adj	adv	conj
1977	work	ability	genius	better	together	striving
1961	pay	ability	effort	important	first	development
1977	contribute to	ability	peace	positive	only	stability
1976	influence	ability	stand	effective	short	rivalry
1970	become	ability	program	American	more	program
1963	adjust	ability	our	changing	worse	challenge
1963	adjust	ability	we	changing	more	challenge
1980	collect	ability	intelligence	rapid	rapidly	accountability
1977	work	ability	genius	remarkable	together	prosperity
1938	pay	ability	tax	graduated	especially	tax
1946	achieve	ability	objective	business	directly	authority
1968	achieve	ability	America	better	not	:
1976	improve	ability	life	governmental	finally	spending
1951	put	ability	force	fighting	part	ours
1995	answer	ability	question	private	not	head

Figure 4: Initial dataset produced from parsed file

Figure [5] is a partial snapshot of the output of the Concept Extraction step showing the processed dataset. We converted this into transactions in ARFF (Attribute Relation File Format) needed by WEKA [20], with our Java program. We then use that as the input to derive association rules. We got several rules of which we list an arbitrary sample in Figure [6].

verb	1790	1791	1792	1793	1794	1795	1796	1797	1798	1799	1800
best	endeavor										
bless	fruit										temple
call for	session		occasion				revision				
call out	president										
carry	others	expedition					treaty				law
add	sanction		information			sanction					
except						part					
form						judgment	security	city	city		
import						Algiers					
indulge						persuasion		hope	hope		expectation
invite						situation	United States				
lament						property					
menace						our					
admit	foreigner										
adopt	measure	measure		rule			In				
afford	retreat					situation	present				pleasure
allow	work										
answer	end		hope								
appear	it		contentment								
ascertain	right			fact			satisfaction	state	state	expedient	
assure	us		Congress							neither	
attend	it						none				
authorize	protection		proportion								
avoid	licentiousnes						war				
bear	testimony										name
become	member	it			people		truth	subject	subject	attempt	

Figure 5: Partial Snapshot of Concept Extraction

---

1. 1795=Union ==> 1958=United States
2. 1872= Union ==> 1995=United States
3. 1958= Nation ==> 1999= United States
4. 1995=work ==> 1999=teacher
5. 1895=Administration 1999=teacher ==> 1958=work
6. 1952=war ==> 1999=terrorist
7. 1952=war 1952= weapon ==> 1999=terrorist

Figure 6: Arbitrary Sample of Rule Derivation

SITACs discovered through association rule mining are then processed through Jaccard Coefficient. Rules which have higher Jaccard Score are stored in the following table. (Figure [7]) The neighboring words for each word participate in rules are derived from the Gutenberg corpus [2] using the Java program explained in SITAC user interface section.

Word	Used with W1	Used with W2	Used with W3	Used with W4	Used with W5	Used with W6	Used with W7	Used with W8
Union	2	1	1		1		3	
USA	1	1		2		1	1	
EU			1		1			1

We ranked these rules as illustrated in the following example. Consider any 2 rules: *Union => USA* and *Union => EU*. We tabulate words as shown in Figure 6 such that *W1….Wn* are words used with a given word such as *Union*. Here, *W1=accession, W2=abandon, W3=Train W6=government W7=blessed/*

In simple terms Jaccard's coefficient compares two sets and the similarity of those two sets is measured as ratio of intersection and union. Thus, the union becomes "Total number of distinct words used" with given words and intersection becomes "number of distinct words used in between two given words"

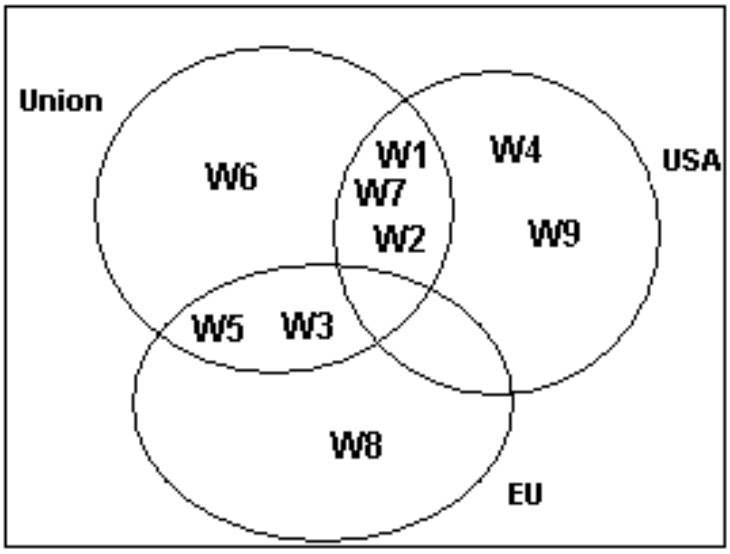

Figure 7: Example of Ranking of Rules

Hence, Jaccard's Coefficient based similarity score *J* is: (as per figure [7])

J (Union, USA) = 3/6 = 0.5, J (Union, EU) = 2/6 = 0.33

Thus, *J (Union, USA) > J (Union, EU)* and therefore, *(Union, USA)* has a higher rank than *(Union, EU)*.

Below is the partial result for word "union"

Neighboring Word	Frequency
Great	13
Trust	1
Aggravate	1
Immovable	1
Endeavor	2
Demonstrated	1
Blessed	1

Partial result for word "United States"

Neighboring Words for USA	Frequency
Great	8
abandonment	4
Trust	1
abating	1

Aberdeen	1
abet	1
abettor	1
abeyance	1
Abhorrence	1
Abide	4

Partial result for word “European Union”

Neighboring Words for EU	Frequency
formation	4
among	1
assistance	2
Atlantic	1
Baghdad	1
collective	1
cooperation	1
countries	1
Defense	1
developing	1

Above frequency values are used in Jaccard’s Similarity calculation.

Likewise, after obtaining the SITACs and their ranking, we stored the results in a SITAC database to serve as the basis for answering queries on the text archives. Thus, a query on the *USA* would use the SITAC *Union* and be answered more accurately,

SITAC are stored in a database to answer to user queries. When a concept is entered in a query it automatically includes its SITACs if there are any to the query and then feed it to the search engine. The search engine for the system presents a sample output as follows.

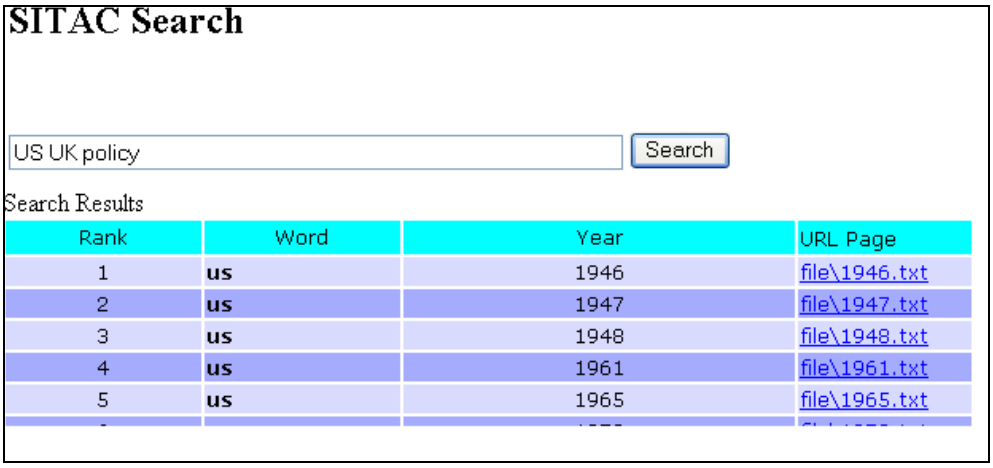

Figure 8 : Partial screen dump of SITAC search engine

4.2 Discussion on Experimentation

It is found that we obtain almost 100% precision and recall with our SITAC system. Precision and recall are widely used benchmarks in evaluating accuracy in IR systems. We use them in our research in order to assess the accuracy of our approach. In IR precision speaks to the exactness of the document retrieval process while recall speaks to the completeness of the document retrieval process. Getting higher number for both will increase the accuracy of the IR process.

The Corpus used in this experiment contains 209 documents which are from Speeches of American Presidents. Reviewing all documents, it was found that only 203 documents have word “United States”. Thus;

$$\text{Precision} = \frac{\text{Number of Relevant Documents Retrieved}}{\text{Total Number of Items Retrieved}} = \frac{203}{203} = 100\%$$

$$\text{Recall} = \frac{\text{Number of relevant Items retrieved}}{\text{Number of relevant Items in the collection}} = \frac{203}{209} < 100\%$$

But by adding the word "*Union*" to the query, we can retrieve all 209 documents since all documents in this corpus have some kind of relationship to "*United States"*. Similarly, we can also prove of gaining higher recall (almost 100%) using concepts UK and British Isle which were used interchangeably in different speeches.

For comparison, when we execute the same query on any general search engine such as Google or Yahoo, we do not essentially get documents where the former term Union was used. Entering "*State of Union policy on UK*" in a search engine did result an article from a university in the UK which has no relation to USA policy on UK. "*United States policy on UK*" gives some related information but not complete.

Search String	Google Result
State of Union policy on UK	1. The Welfare State This page introduces comparative social policy. It discusses the welfare state in Britain, France, Sweden, Germany, the United States, the European Union ... www2.rgu.ac.uk/publicpolicy/introduction/wstate.htm 2. Common Agricultural Policy - Wikipedia, the free encyclopedia. The Common Agricultural Policy (CAP) is a system of European Union agricultural ... 3.7 UK rebate and the CAP; 3.8 CAP as a form of State intervention ... en.wikipedia.org/wiki/Common_Agricultural_Policy 3. Member State of the European Union - Wikipedia, the free encyclopedia. Before being allowed to join the European Union, a state must fulfill the economic .... the first countries changed their policy and attempted to join the Union, ... The UK & Switzerland. http://www.britishembassy.gov.uk/servlet/Front? ... en.wikipedia.org/.../Member_State_of_the_European_Union – 4. STATEWATCH - monitoring the state and civil liberties in Europe EU: Council of the European Union: 8th Annual report for 2009 on the Regulation ... UK: Home Office: Police powers and procedures 2008/9 report (pdf). ... It shows how European governments and EU policy-makers are pursuing unfettered ...

	www.statewatch.org/
United State policy on UK	1. BBC News - History lessons for US policy on Israel? Mar 20, 2010 ... With US-Israeli ties at a low ebb over plans to build new homes for settlers in disputed East Jerusalem, the BBC's Paul Adams wonders if a ... news.bbc.co.uk/2/hi/programmes/from_our_own.../8577691.stm 2. US government rescinds 'leave internet alone' policy • The RegisterFeb 27, 2010 ... The US government's policy of leaving the Internet alone is over, according to Obama's top official at the Department of Commerce. ... www.theregister.co.uk/2010/02/27/internet_3_dot_0_policy/ 3. Position statement in support of open access publishing | Welcome ...Gibbs Building, 215 Euston Road, London NW1 2BE, UK T:+44 (0)20 7611 8888. Home > About us > Policy > Policy and position statements > Position statement in ... www.wellcome.ac.uk › ... › Policy › Policy and position statements 4. LRB · John Mearsheimer and Stephen Walt · The Israel Lobby. For the past several decades, and especially since the Six-Day War in 1967, the centerpiece of US Middle Eastern policy has been its relationship with ...
SITAC Search Result On either "United State policy on UK" or "State of Union policy on UK" produces the similar result which are	Resulted 10 speeches which are absolutely related "United state or State of Union policy on UK Year 1799 speech Year 1823 speech Year 1882 speech Year 1941 speech Year 1942 speech Year 1945 speech Year 1952 speech Year 1960 speech Year 2001 speech Year 2003 speech

Figure 9: Google Search results on "United State Policy on UK"

WordNet [19] has closely related words called synset where each synset refers to a unique concept. It may give some information on correlated concepts but does not

incorporate any temporal aspects. Moreover WordNet relations are unidirectional (i.e. C1=>C2). Word "*United States"* does not show "*Union*" as a related concept but word "*Union*" appears as a related concept of "*United States*". SITAC bi-directional relationship serves much better in this situation.

Another significant advantage of our approach is that since the SITACs are pre-computed and stored in a database, they can directly be used to answer user queries, which is efficient with respect to time because it only incurs a one-time cost of computation as opposed to a recurrent cost of performing certain operations to find relevant terms in real time whenever querying is performed. However, this does pose a trade-off in terms of space because it requires additional storage of the SITAC database, so it should be used when using such storage space is feasible.

We have presented a summary of the working of our system through these demonstration snapshots. The detailed interaction between the user and the SITAC system, considering various example queries along with SITAC system responses and comparative samples will be presented in the live demonstration.

4.3 Performance Improvements

Proposed association rule mining for discovering SITAC would consume too space and would be too corpus-specific to implement on a large scale. It will not be feasible to implement real time discovery of SITAC when responding to user queries. Prior to processing user queries, it is required to pre-process text sources, perform AR mining,

and store discovered SITAC's through AR mining in a Database. Moreover, it would need huge amounts of data and metadata from text corpora in order to satisfy the minimum support and confidence thresholds. Another critique of this method is that since it takes into account only frequent concepts in the rule mining process, it may not discover rare but interesting associations between the concepts. Therefore, an important issue is to figure out whether we can modify this rule mining approach to overcome these drawbacks. Some potential suggestions include not considering all pairs of possible concepts while deriving associations but only a few top-k combinations; and using the association rule mining approach in conjunction with another methods such as the random walk approach and / or a graph model in order to enhance performance.

5. Related Work

5.1 Text Mining

There has been much works done in the area of text mining. Most works in this area address the word sense disambiguation (WSD) and document classification. Our approach has some similarities to Michael Lesk algorithm[7] for word senses disambiguation. Michael Lesk's algorithm has two parts; 1. When two words are used in close proximity in a sentence, they must be talking of a related topic 2. If one sense each of the two words can be used to talk of the same topic, then their dictionary definitions must use some common words. We exploited Lesk's algorithm by studying the neighborhood of concepts, applying linguistic properties to each word and finding similarities of their usage in various contexts.

Most word disambiguation approaches have used clustering methods. However Lin (1997) introduces an unsupervised method which is able to tag test data with appropriate senses from machine readable dictionaries (MRD). Similar to our approach he uses syntactic dependency and semantic similarity as disambiguation information. The basic principle of this method is the observation that two occurrences of the same word have identical meaning if their local context is the same. Lin's further studies propose an algorithm based on the intuition that two different words are likely to have similar meanings if they occur in identical local context. We also proved that if two words of the same occurrence have identical meaning if they participate in multiple different occurrences. We also consider the time when they appear.

Many other researches are done in finding word sense disambiguation (WSD) which helps with improving performance and accuracy of an IR system. As per Lu and Keefer (1994) users frequently use short queries in Internet searches, which make it difficult to retrieve relevant documents. Those user queries can be expanded by adding semantically related words to the same. Mihalcea(1999). Our approach enhanced user queries by adding the SITACs which contain sense and its time factor, giving more meaningfulness to the query.

Named entity recognition is addressed in works such as [3]. Research has been conducted on sequence classification and mining as in. [8,11]. Similarity measures over text and web documents have been studied in the literature, e.g., [5,14]. Xu et al. [20] address a somewhat related problem of local context analysis but without taking into account any temporal factors.

5.2 Sequence Mining

The mining of sequential patterns has been studied in the literature. Agrawal et al. in [1] they propose two algorithms, AprioriSome and AprioriAll for mining a large database of customer transactions to discover association rules over sequences. They find the maximal sequences among all sequences that have a certain user-specified minimum support. In [18], the authors enhance their earlier work presenting an algorithm called GSP for discovering generalized sequential patterns. It is faster than the AprioriAll algorithm. It also considers time constraints specifying a minimum and maximum time period among adjacent items in a pattern.

Zaki et al. in [21] perform the mining of subsequences that are frequent using minimum support levels and extend this paradigm to sorting. Their goal is to reduce input-output and computation needs in handling incremental updates to the data, while mining, since data sources could undergo changes. Their techniques; “mine sequences” taking into account of user interaction and database updates. In [11], they also deal with sequential data. They develop sequence mining methods for selecting features to serve as an input for classification with algorithms such as Naive Bayes. The data in this work consists of examples, each represented as sequences of events, each event having a set of predicates. Thus, their goals are quite different from ours. In all these sequence mining approaches, we can draw some analogy with our work. However, none of these consider terminology evolution where terms change over time.

5.3 Temporal Information Retrieval

Among many classification and mining researches, Klaus Berberich et al. [6] have addressed Temporal Terminology using Hidden Markov Model (HMM). They consider frequency of co-occurrence terms between concepts. For example, IPod and Walkman are mostly used with words “*portable*”, “*music*” and “*earphones*”. This word overlapping is used to determine their semantic similarity.

In addition, some prior work in this area has been done by our team [13]. One of the approaches assumes anticipated user queries as additional inputs, which have been found

suitable at the initial stage of this research. Our approach in this paper does not require any such anticipated queries and is thus more generic.

Other related work includes inter-transaction associations on temporal document collections as in [12]. We can draw an analogy here, though we make an additional contribution in terms of intelligent query processing by the discovery of semantically identical temporally altering concepts for historical time-aware query translation.

While the work in this thesis is orthogonal to some of the existing literature, our focus is on both the contextual as well as the temporal aspects in discovering concepts for answering queries incorporating terminology evolution in text archives. Moreover once the SITACs are discovered, they can be directly used to answer user queries with minimal processing time. Thus the approach is efficient with respect to time.

6. Conclusions

6.1 Summary

In this thesis, we have addressed the problem of terminology evolution in text archives spanning long time periods. This motives the need for time-aware query translation in order to provide accurate responses to user queries over such text sources. We have proposed a solution that involves discovering SITACs in text archives, i.e., Semantically Altering Temporally Identical Concepts, with the goal of performing the required time-aware translation.

Our proposed solution approach by the name SITAC constitutes an integration of natural language processing, association rule mining and contextual similarity, for translating and answering user queries appropriately. Accordingly, we have implemented the SITAC system in order to provide a good understanding of the approach and have conducted evaluation with this using real online text archives.

The focus of our experiments is the Gutenberg [2] corpus of the USA Presidential State of the Union addresses and our experimentation has revealed that we achieve good information retrieval with our approach. This is indicated by the precision and recall figures and also by comparison with existing search engines.

We claim that our approach would be useful in web-based information retrieval in text archives that contain historical data and require intelligent time-aware query translation.

6.2 Technical Contributions

This work involves the following technical contributions and has been accepted for publication in the AAAI 2010 conference.

1. Identifying the problem of terminology evolution in text archives that motivates the need for time-aware query translation and presenting the challenges in problem definition.
2. Introducing the terminology of SITACs to address the given problem and propose a solution accordingly.
3. Simulating human thinking by developing a methodology to discover the SITACs using the manner in which humans associate concepts that co-occur frequently, thus making a contribution to artificial intelligence.
4. Proposing a collaborative framework of natural language processing, association rules and contextual similarity as a novel data mining approach to solve the problem of time-aware query translation.
5. Solving various non-trivial subtasks such as:
    - Preparing datasets from text archives, a challenge involving linguistics, especially in harnessing the semantic properties; Defining adequate transactions to capture the essence of the problem
    - Selecting the appropriate contextual similarity measures and adapting them with respect to the problem Implementing the SITAC system as a useful software tool in Java, including a query translation engine with user interaction

6. Developing the SITAC software tool available for use in information retrieval systems and suitable for demonstration in appropriate venues.
7. Evaluating this SITAC system with real data and depicting its effectiveness incorporating a fair comparative assessment with pros and cons.

6.3 Future Work

Given that we have presented a reasonable solution to the problem of time-aware query translation with our SITAC approach, we outline further issues here. Future work in this area mainly includes performance improvements with respect to space and time in the proposed solution. Some of this involves exhaustive comparative studies with the state-of-the-art in terms of several parameters, and the potential use of the SITAC approach in conjunction with other approaches such as random walks to solve the given problem. This could also involve working in collaboration with other researchers who have addressed this problem and seek to enhance their solutions. Other future work entails the development of large scale systems analogous to the SITAC search engine developed here in order to be used in information retrieval and suitable related applications.

This work encourages further research in intelligent systems for query processing and information retrieval that incorporate human thinking and address various issues such as temporal and semantic factors from a user perspective. As is widely acknowledged in this field, there is a growing need for systems of this nature to make computational systems more intelligent. The future looks promising and presents great opportunities for more accomplishments.

7. References

[1] Agrawal, R., and Srikant, R.: ”Mining Sequential Patterns”. In proceedings of ICDE (March 1995), Taipei, Taiwan, pp. 3–14.

[2] Gutenberg Corpus: ”U.S. Presidential Inaugural Addresses”. In The Project Gutenberg EBook of U.S. Presidential Inaugural Addresses, www.gutenberg.net (Jan 2004), EBook Number 4938, Edition 11.

[3] Hasegawa, T., Sekine S.,Grishman R., “Discovering Relations among Named Entities from Large Corpora”,ACL (Aug 2004), pp. 415-422.

[4] Jeh., G., Widom., J.: ”SimRank: A Measure of Structural-Context Similarity”. KDD (Jul 2002), pp. 538–543.

[5] Janetzko D, Cherfi H, Kennke R, Napoli A, and Toussaint Y. Knowledge-based selection of association rules for text mining. In *Proceedings of ECAI’2004*, 2004.

[6] Klaus Berberich Srikanta Bedathur Mauro Sozio Gerhard Weikum

“Bridging the Terminology Gap in Web Archive Search!”,Max-Planck Institute for Informatics Saarbr¨ucken, Germany, Twelfth InternationalWorkshop on theWeb and Databases (WebDB 2009), June 28, 2009, Providence, Rhode Island, USA.

[7] Lesk, M. E. (1986). Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In Proceedings of the Fifth International Conference on Systems Documentation, pages 24{26, Toronto, Canada. ACM.

[8] Mei Q., Zhai C. “Discovering Evolutionary Theme Patterns from Text An Exploration of Temporal Text Mining”, KDD’05, August 21–24, 2005, Chicago, Illinois, USA. KDD’05, August 21–24, 2005, Chicago, Illinois, USA. .

[9] Mihalcea, R. (1999). Word sense disambiguation and its application to the Master's thesis, Southern Methodist University.

[10] Mihalcea, R., Tarau, P., and Figa, E. (2004). Pagerank on semantic networks, application to word sense disambiguation. In Proceedings of The 20st International on Computational Linguistics (COLING 2004), Switzerland, Geneva.

[11] Miller G.A and Charles W.G. Contextual correlates of semantic similarity. Language and Cognitive Processes, 6(1), 1991.

[12] Norvag, K., Eriksen, T.O. , Skogstad, K.I : ”Mining Association Rules in Temporal Document Collections”, Dept. of Computer and Information, Systems (2006), NTNU, Norway.

[13] Roychoudhury D.., Varde A., “Terminology Evolution in Web and Text Mining Using Association Rules”, Dept. of Computer Science (May 2009), Montclair State University, NJ.

[14] Strehl A. Ghosh, J. and Mooney R.: “Impact of Similarity Measures on Web-page Clustering”, AAAI, (Jul 2000), pp. 58-64.

[15] Srikant, R. and Agrawal, R.: ”Mining Sequential Patterns: Generalizations and Performance Improvements”. EDBT (Mar 1996), pp. 3–17.

[16] Strehl A. Ghosh, J. and Mooney R.: “Impact of Similarity Measures on Web-page Clustering”, AAAI, (Jul 2000), pp. 58-64.

[17] Tan P.N., Kumar V., and Srivastava J. Selecting the right interestingness measure for association patterns. In *Proceedings of KDD’2002*, 2002.

[18] Varde A., Bedathur S., Berberich K, Weikum G., “Time Aware Query Translation over Text Archives”, Max Planck Institute for Informatics (Jul 2008), Saarbrucken, Germany.

[19] WordNet http://wordnetweb.princeton.edu/perl/webwn

[20] Xu, X. and Croft. B.: “Cluster-based Language Models for Distributed Retrieval”, SIGIR (Aug 1999), pp. 254-261.

[21] Zaki, M.J., Parthasarthy, S., Ogihara, M., Dwarkadas, S.: ”Incremental and Interactive Sequence Mining”. *CIKM* (Nov 1999), pp. 251–258.