TECF: Accomplishment of Content Framework Tactic to Temporal Theme Condensation

A.V.Seetha lakshmi
Asst. Prof, Department of IT
G.T.N Arts College, Dindigul, India

Dr.S.P.Victor
Head, Department of Computer Science
St.Xavier’s College (Autonomous),Palayamkottai, India

Abstract: A topic is defined as a developing event which contains many related events and activities. The convenient storage of documents in the internet leads to the difficult in exploration since there are sequence of documents published by different authors for the particular keyword. Due to the phenomenal growth of a large number of documents in the internet it is difficult for the users to read the entire contents and conclude what exactly present in the document. The situation is even worse if the event is time related. In this study we define a task which summarizes the core content in the chronological order and this makes to readers to understand the content easily. The proposed model, called TECF(Theme Encapsulation and Content Framework) derives themes from the documents on the topic with the help of Eigen vectors of the temporal block association matrix, from this the summaries of the document are extracted. Finally the extracted summaries are associated to find the temporal closeness with the help of evolution graph. In the proposed techniques the core content related to the topic is covered in the coherent and consistent manner when compared with the existing separation techniques like human based summaries.

Keywords: Data mining, Text mining, Encapsulation, Temporal, Eigen vector

I. INTRODUCTION:

The remarkable growth of number of the documents posted on the internet provides an abundant source of information. The current technologies provide efficient search requests to satisfy the keyword search request, readers still have difficulties to obtain the needed knowledge from overwhelming number of documents. This situation is even more confusing if the event is related to the time related incident. Defense Advanced Research Project Agency (DARPA) initiated Topic Detection and Tracking. The project defines the topic as the “semantic event or activity along with all directly related events and activities” The goal of our project is to detect all topics and track related documents from several document streams automatically such as online news feeds. We use topic anatomy to summarize the essential document in the chronological order.

Topic anatomy is a prominent text mining research pattern that involves three major Tasks: theme generation, event segmentation and summarization, evolution graphconstruction. The first task involves in identifying themes of the topic from all the related documents[1]. When reading the overall document one theme may reflect higher importance than the other . The defining event must be unique. Second task involves event segmentation and summarization process that extracts For instance Google News service employs TOPIC DETECTION AND TRACKING arrange documents related to the news topics from some online news website. While our system TECF(Theme encapsulation and content framework) detect the core content of the document in the effective manner.

II. RELATED WORK:

Text segmentation:

The main aim of text segmentation is to divide the input text into non overlapping segments in that each segment is related to the topic. Depends on the input text, the segmentation is classified into two types, they are story boundary detection and document sub topic identification. The input given to story boundary detection is usually a text stream. Eg: there are no distinct boundaries between the documents from online newswires. Generally, cue phrases can be used to identify the boundaries between the documents. In document subtopic identification, the input is a single document and identify the blocks in the document that are relevant to the certain subtopic. For example, search engine can retrieve the documents and return the most relevant blocks segmented from the searched results to the users. The cue phrase approach is not convenient for document subtopic identification because the subtopics in a document are similar, hence the noticeable cue phrases about the subtopic boundaries are virtually not existing. Then decomposing the document into a set of consecutive sentences and the word usage in every block to be analyzed to find the subtopic boundaries. One major problem with this approach is that the information in the block is not sufficient to determine the blocks interrelationships.

Brants et al and Choi et al [2] applied the latent semantics concepts to enrich the information in a consecutive set of sentences. Their method uses a training data to construct a domain-dependent. Blei and Moreno [3] utilized Hidden Markov models are used to detect the subtopics of document are modeled as states in an HMM, and every document is
treated as a series of blocks, which is used to calculate the best state transition. When two successive states in the best state transition sequence are different, then the boundary will occur in the documents.

Ji and Zha[4] proposed a domain-independent segmentation method that models the block of the document is treated as a square matrix and consider the matrix as a grayscale image. Then some image processing method is applied to sharpen the boundaries in the image. Finally, the significant and diagonal segments are selected as a block of the document.

**Text summarization:**

Generic text summarization automatically creates one or more documents that capture the list of documents. As a document’s content may consist of many themes, generic summarization methods are used to extend the summary diversity to provide wider coverage of the content of the documents[5]. In generic text summarization, which composes the summaries by extracting the informative sentences from the actual documents. Extraction-based text summarization methods can be classified as either supervised or unsupervised. In Supervised methods are used to summarize the document by labeling the sentences of the document as informative or non-informative. Shen et al.[6] proposed a supervised summarization method that uses conditional random fields (CRF) to train a classification model, which calculates the informative sentences. Top-ranked sentences are selected as a summary. With suitable training corpora, supervised summarization methods performs well as unsupervised summarization methods. However, domain-dependency is a drawback of supervised summarization methods. The trained summarization model is specific to a certain document domain. Deploying a supervised summarization method in a new domain involves explaining another manual training corpus, but it is a time-consuming task. In general, the number of studies show that interagreement between explainers is low, which affects the quality of the training corpus and the acquired summarization model. We recently proposed the summarization method are unsupervised. Next, we consider the method and discuss the limitations of applying them to the topic anatomy task. Gong and Liu applied Singular Value Decomposition (SVD) to a document term-sentence association matrix can be used to perform extraction-based generic summarization. This method uses the decomposed singular vectors for the themes of the document and composes diverse summaries by selecting the informative sentences from important themes. Nomoto and Matsumoto[7] proposed the X means algorithm, which is used to find the sentences that contain more useful information from the clusters. Allan et al. temporal summarization method[8] processes topic sentences in a chronological order. This method weights the informativeness of the sentences depends on the sentences usefulness and novelty. A sentence is useful for readers to comprehend a topic if its content is similar to the main themes of the topic. To avoid the extraction of redundant summary sentences, then the sentence should also be different to all previously extracted sentences. Nowadays, graph-based summarization methods are used to modeled the relationships between the sentence and terms in the document. This model considers a sentence informative if it connects with many informative terms and reinforcement procedure updates the informative scores of the terms and sentences. Finally, the summaries are composed by selecting the informative sentences.

Erkan and Radev[9] represent the set of documents as a graph in that sentence are represented as nodes and edges connect the content similarity between the sentence, a sentence is more informative, if it connects with many sentences, hence by extension, the connected sentences are also informative. From the informative scores of the sentences, then the informative term can be taken as the summary. Topic summarization differs from existing text summarization because of its temporal properties. The topic summaries should describe the storylines of the topics.

**Topic Evolution Mining**

Kleinberg[10] developed a technique that constructs the hierarchical tree from a series of documents. The technique uses an HMM-based, the transition diagram to model the status of the topics and splits atomic into diverse themes, modeled as tree branches. If the topic contains bursty information, Nallapati et al.[11] formalized the problem of topic evolution mining as a text clustering task in which the identified clusters i.e., the events of a topic are connected chronologically to form the evolution graph of the topic. Yang and Shi[12] focused on the temporal properties of a topic and showed that evolution graphs can be obtained by using the temporal information about topics. Feng and Allan[13] proposed an incident threading method that is similar to the proposed method. The method first identifies incidents from news documents, then identifies the semantic dependencies between the incidents. Swan and Allan[14] proposed a timeline system to detect the topic’s importance by graphically at the specific time.

**III. IMPLEMENTATION**

**Theme model**

In this section, we describes our model and the method used to implement the system.

A Theme can be considered as a real world incident that has one or more sentences, which are related to a particular incident. In the entire theme, one sentence may be more important than the rest of sentences and it may be repeated in many sets of documents. We specify an event as an important theme enhancement that carries out for a sequence of time period. By the nature, all the sentences are merged to form the corpus of the theme. Even though the events of the subject are chronologically disjoint, they are assumed to be language-dependent in order to enhance the sentence. The proposed model finds the inter-relationship between the theme and the event to make the theme’s progression graph.

The major tasks involves in our process are crawling and extraction, Matrix algorithm, Event segmentation and summarization and final one is Story-boundary detection.

1) Crawling and extraction

The crawling is the basic operation performed in every web search Engines. Crawling process takes place with the help of crawler. The crawler is the program, that is already developed in the search engine, which isolates the citation.
from the world wide web (WWW) with respect to the particular theme. From which we get the backlink. Normally, every backlink has the set of documents. Such documents may contain set of sentences, which helps to develop the storyline of the theme/topic. This storylines helps to understand the topic by making the user to comprehend the topic but it involves more time consumption. Hence, in our system, we reduce this complexity by introducing extraction technique.

The pictorial representation of the crawler is as follows,

```
+-----------------+     +-----------------+
| INTERNET        | --> | CRAWLER         |
|                 |     | CRAWLER         |
|                 |     | REPOSITORY      |
```

In the early days, single crawler can be used. But now, it is explored to parallel and multiple crawler to enhance the optimization of the search engine. Hence, the user can benefit with the increased speed in displaying the output to the query. The main function of the crawler is to download the citation related to the theme or query[15].

The processing step involved in crawler are as follows,

i) These seedlists which are maintained by the crawler should be cleared at first to make sure that the old data will not be allowed to the user.

ii) Every webhosts has an IPAddress, which is determined in this step.

iii) With respect to the user’s search, the crawler can download the corresponding document.

iv) The backlinks present in the document are extracted by the crawler.

v) The user can perform their own operations on the downloaded document.

vi) While processing the document, the user may switch on to other citation and if the citation is new to the user, then such citation is added to the seedlists.

vii) The process is repeating from (i) for every query requested by the user.

**PSEUDOCODE FOR CRAWLING**

1) Initialize objects SERV, DTT to access TECF service and datatable.
2) if (offline.checked == true)
3) offline text are extracted
4) else
5) online text are extracted
6) Initialize start, end to retrieve topic using indexing
7) define bind function
8) add topic, url, link to the datatable
9) Finally, results are stored in ‘result’ variable

**IV. DESCRIPTION**

1. Initialize the objects to access the database, secondly, if the user selects online mode then the online text are accessed and if the user selects the offline mode, then the offline texts are accessed for further processing.
2. In the next step, using the indexing technique the topic, and its url are retrieved.
3. Add the topic, url and the link which has been downloaded from the website, to the database.
4. And using the bind function, bind the data for further processing.

**Extraction**

In the Existing technique such as TDT, after performing the crawling operations, they directly apply some of the summarization technique to produce the summary of the original document which sometimes did not satisfies the internet user since it can produce summary upto two lines and probably this two lines will be from the first two lines or last two lines of the original document.

Some of the summarization techniques are as follows,

i) **forward method:**

   It can produce the summary by extracting the first paragraph of the Original document and display the first two lines of the paragraph.

ii) **backward method:**

   It can produce the summary by extracting the last paragraph of the Original document and display the last two lines of the paragraph.

iii) **k-means method:**

   The documents are formed into clusters and it produce the summary by randomly pick up the cluster. It sometimes may not produce the exact summary[7].

iv) **frequent-content word:**
It can produce the summary by extracting the frequently arriving words in the paragraph[16]. Later, a new technique called SVD (single value decomposition)[5] is used which enhances the summarization results. Our technique resembles SVD but we adopt soundix technique for the entries of vector.

We adopt two types of extraction called BLOCK EXTRACTION and THEME EXTRACTION.

**BLOCK EXTRACTION**
In this, the blocks, that is, paragraphs from the Original document are extracted using Indexing Technique. To ensure the chronological order, we are extracting the blocks from the Original document.

In Indexing, we assign index value to each tag used in source code. With that index value, we can identify the paragraph tag (i.e.) <p> and with this, the paragraph are extracted.

Extracting blocks from inbound links in the websites. In this block extraction extracting blocks and image for the topic. We obtain n number of blocks from the web sites.

**PSEUODOCODE FOR EXTRACTION BLOCK EXTRACTION:**
1) Create object for the arraylist.
2) Initialize the linkcount.
3) for(int i=0;i<linkcount;i++)
4) Add topic, url, link, .img path to the datatable.
5) Initialize the count split variable.
6) Obtain blocks for every block.

**DESCRIPTION:**
1) Create objects to access the arraylist and add data to the arraylist
2) Initialize the maximum of link from the website
3) Initialize the looping constraint with the variable i
4) Add the url, topic, link to the datatable with respect to the linkcount
5) Initialize the count split to split up the blocks from the document which is downloaded from the website
6) Blocks are extracted from the document

**THEME EXTRACTION**
In this, the Themes, that is, sentences from the identified block are extracted using the Cue Phrase Identification Approach. We are extracting the themes to reduce the complexity involves in Matrix Calculation.

After extracting the blocks, Events are extracted from the blocks. Set of events are extracted for every blocks.

The cue phrase identification extract the sentences by identifying the full stop at the end of each sentences.

**PSEUDO CODE FOR THEME EXTRACTION:**
1) Create object for the arraylist
2) Initialize the linkcount
3) for(int i=0;i<linkcount;i++)
4) Add block id, topic, url, link, .img path to the datatable
5) Initialize the count split variable
6) Obtain themes for every block

**DESCRIPTION:**
1) Create objects to access the arraylist and add data to the arraylist
2) Initialize the maximum of link from the website
3) Initialize the looping constraint with the variable i
4) Add the block, url, topic, link to the datatable with respect to the linkcount
5) Initialize the count split to split up the themes from the block of the topic which is downloaded from the website
6) Themes are extracted from the document

**V. MATRIX CALCULATION**
To make the summarization results better than other existing technique, we use matrix calculation[17]. We perform matrix for every themes extracted from above methods. This Matrix should be Orthogonal and nxnSymmetricMatrix. We calculate the Eigen vectors for every extracted themes. By this, we can get the effective event for the particular topic. To perform the Matrix calculation, we use two technique called StopwordRemoval[18] and Soundix[18]. The Stopword Removal helps to remove the stopwords such as nouns, verbs etc which we are frequently using in the topic or in the theme. It also helps to reduce the Matrix size, so that we can also save database space. The Soundix helps to assign integer value to each and every content word which we are giving as input to the Matrix. Let us consider, the topic as one matrix and the theme as another matrix. such that,

\[
A = \begin{bmatrix}
    t_1 & t_2 \\
    t_5 & t_6
\end{bmatrix}
\]

\[
B = \begin{bmatrix}
    T_1 \\
    T_2 \\
    T_5
\end{bmatrix}
\]

\[
AB = A \times B
\]

A -> TOPIC, B -> THEMES, t1…tn = Topic content word, T1…Tn = Theme content word
For this, Matrix multiplication is performed. From that, we obtain $\lambda \rightarrow$ EIGEN VALUES

From this, eigenvectors[17] are calculated.

**Algorithm:**

**INPUT:** Extracted Themes from Original document

**OUTPUT:** Eigen vector for each and every Themes

i) Remove the Stopwords of the Themes which we are extracted from the Original document.

ii) Form the Contentword for every Topic and Theme.

iii) We obtain ‘n’ number of Themes for a Topic.

iv) Using Soundix technique, we assign integer value for every contentword.

v) Form the ‘A’ Matrix which has Topic contentword value.

vi) Form the ‘B’ Matrix which has Theme contentword value.

vii) Multiply first row of ‘A’ matrix with first column of ‘B’ matrix. Such that,

$$ [a_{11} \ a_{12} \ a_{13}] \times \begin{bmatrix} b_{11} \\ b_{21} \\ b_{31} \end{bmatrix} $$

viii) Repeat Step[vii] until $a_{m \times b_{n}}$

ix) Obtain the Eigen vector for every themes.

**PSEUDOCODE FOR MATRIX CALCULATION**

1) serv.readthemes()
2) Initialize eigenvector
3) for (int i = 0; i < ds.Tables[0].Rows.Count; i++)
4) calc(dtb.ROW[i].theme, dtb.ROW[i].topic)
5) bind() values
6) page index changing()

**DESCRIPTION**

1) Using the readtheme function, the themes which are extracted from the block of topic are read and saved in temp files
2) Initialize the eigenvector variable
3) Using the for loop, the loop constraint is fixed up to the maximum count of rows which are already bind in datatable
4) Calculate the eigen vector using matrix calculation, in which themes and topic are taken into account and perform matrix multiplication
5) Using the bind function, bind the values in the datatable
6) Using page index changing function, more pages are binded for the obtained result.

**Fig1:** Event segmentation and summarization

A theme $v_{i}$ in $V_{L}$ is the normalized eigenvector of dimension n, where the (i,j) entry $v_{i,j}$ indicates the correlation between the block $I_i$ and the theme $J_j$. As a topic blocks are indexed chronologically, a sequence of entries in $v_{i}$ with high values can be considered as noteworthy event embedded in the theme and valleys (i.e., a sequence of small values) in $v_{i}$ may be event boundaries. However, the entries in eigen vectors are invertible. Moreover, Klienberg[20] and Nicholas and Dahlberg[21] proved that both the positive and negative entries of eigenvector may contain meaningful semantics for describing certain concepts in the document; and the amplitude of an entry determines the degree of its correlation to the concept. The task of our event segmentation and speech endpoint detection are similar in that they both try to identify
important segments in sequential data. For example for the given speech utterance in Figure 2, the speech endpoint detection task involves distinguishing the significant segment $S_2$ from the insignificant silent segment ($S_1$ and $S_3$) mixed with background noise. Here, $S_2$ points with large positive and negative amplitudes therefore, we adopt Rabiner and Sambur’s R-S Endpoint Detection Algorithm[22] for Event segmentation. To segment Events, the R-S Algorithm examines the amplitude variation of an Eigen vector to find the Endpoints that the partition the theme into a set of significant events.

In the R&S Algorithm, every block in an Eigen vector has an Energy value. To calculate the Energy, we adopt the Squaresum scheme, which has proved effective in detecting endpoints in noisy speech environments[23] and is defined as follows

$$\text{eng}(i,j) = \frac{1}{H} \sum_{h=0}^{H/2} |v_{i+h,j}|^2.$$  

Here eng(i,j) is the energy of the block I in a theme j, and H specifies the length of the sliding window that is used to smooth and aggregate the energy of a block.

Figure shows the eigenvector of a theme and its energy contour. A peak in the energy contour indicates that the corresponding sequence of blocks in a significant development of the theme; and so it is identified as an event. To segment event from energy contours, we define a segmentation threshold $th_{seg}$ as 0.1; then, we scan the energy contours linearly to find consecutive blocks whose energy values are above threshold. To reduce the number of false alarms during event segmentation and refine result we employ two frequently used heuristics: 1) we merge close events 2) we prune small events[23]. For each event the block with the largest amplitude is selected as the event summary.

A unique feature of summarization approach is the introduction of the event segmentation process to extract the semantic construct “event” before summarization. Our method further describes the development of themes via summarized events to better comprehend a topics story lines.

Figure 4: The eigenvector of of a theme and its energy contour.

**PSEUDOCODE FOR EVENT SEGMENTATION**

1) serv.readmatrix()
2) serv.segmentdel()
3) SqlCommand("truncate table segmentation", con)
4) res = serv.segment()
DESCRIPTION

1) Initially read the matrix content using readmatrix() function
2) Segment del which is the function used to delete the table content
3) Using sql command, delete the content
4) The variable ‘res’ is used to store the result of segment() function
5) Available topics are added to the variable ‘topiclst’
6) Available blocks are added to the variable ‘blocklst’
7) Using the looping constraint, enter into function
8) Using the looping constraint, select every extracted block, theme, topic, etc.
9) Read everything and add to the segmentation

STORY BOUNDARY DETECTION:

Story boundary detection (or story segmentation) is used to identify where one story ends and another story begins in a stream of text. It serves as a necessary precursor to various tasks, such as topic detection and tracking, information extraction, indexing, retrieval and summarization, etc. A typical broadcast news retrieval system is able to locate the particular Positions in a repository that match the user’s query, but lack the ability of determining where the user-interested stories begin and end[19]. The data corpus consists of a collection of information, then using the classification techniques to create the boundary between two different news documents. The input from the event segmentation is detected core parts. From that, we have to identify the endpoints between the documents.

The story boundary detection is used to create the boundary between the two different news documents. First step in this process is that identify the endpoints in the summarized documents. Second step is create the boundary; this step is useful for display the news documents efficiently and helps the user to know the different news.

PSEUDO CODE FOR STORY BOUNDARY DETECTION:

1) serv.readsegment()
2) serv.summarydel()
3) SqlCommand("truncate table summarization", con)
4) res = serv.summarize()
5) ArrayList topiclst = new ArrayList()
6) SqlCommand("select distinct(topic) from segmentation", con)
7) while (drr.Read())
8) topiclst.Add(drr[0].ToString())
9) for (int i = 0; i < topiclst.Count; i++)
10) if (theme != "" && summary != "")
11) SqlCommand("insert into summarization")
6) select the topic from the segmentation using SQL command
7) using while loop, read the selected table in database
8) add the obtained topic to the arraylist
9) using the looping constraint, select every extracted block, theme, topic, urletc
10) if the themes matches with the summary, then further step is initialized
11) add the themes into summarization

VI. CONCLUSION:

Publishing many news documents related to same topic are posted by different authors and their opinions during the topic life span. The summarization method are used to help the user to obtain the news from different documents. In this paper, we have presented a theme encapsulation and content framework (TECF), which extracts the themes, events and connects the associated events to form evolution graph. From this, we obtain a core parts of the documents. TECF can produce highly representative summaries that composed by experts.

REFERENCES:
[1] Chien Chin Chen and Meng Chang Chen, “TSCAN: A Content Anatomy Approach to Temporal Topic Summarization,” IEEE Transactions on Knowledge and Data Engineering Volume: 24 Issue: 1


