WORDS SEQUENCE PATTERN MINING USING PATTERN TAXONOMY MODEL

1SUBRAMANYAM 2CHRISHIKESAVA REDDY

Professor & HOD Dept of Computer Science II M.Tech (CSE)
Annamacharya Institute of Technology & Sciences
Annamacharya Institute of Technology & Sciences Rajampet, A.P, Rajampet, A.P, India
Email: hrushi.c@gmail.com

Abstract — The text data of text mining has gradually become a new follow a line of investigation. Text clustering can greatly simplify browsing large collections of documents by reorganizing them into a smaller number of patterns in text documents manageable clusters. Text clustering is mainly used for a document clustering system which clusters the set of documents based on the user typed key term. Firstly the system preprocesses the set of documents and the user given terms. We use the feature evaluation to reduce the dimensionality of high-dimensional text vector. The system then identifies the term frequency and then those frequencies are weighted by using the inverted document frequency method. Then this weight of documents is used for clustering. Feature clustering is a powerful method to reduce the dimensionality of feature vectors for text classification.

Terms: Text mining, text clustering, words mining, Reshuffling Pattern Mining.

I. INTRODUCTION

Text mining offers a solution to this problem by replacing or supplementing the human reader with automatic systems Undeterred by the text explosion. It involves analyzing a large Collection of documents to discover previously unknown Information. The information might be relationships or Patterns that are buried in the document collection and which Would otherwise be extremely difficult, if not impossible, to discover. Text mining can be used to analyze natural language documents about any subject, although much of the Interest at present is coming from the biological sciences. Originally, research in text categorization addressed the binary problem, where a document is either relevant or not. Text mining involves the application of techniques from are as such as information retrieval, natural language Processing, information extraction and data mining. Information Retrieval (IR) systems identify the documents in a collection which match a user’s query. The most well known IR systems are search engines such as Google, which identify those documents on the World Wide Web that are relevant to a set of given words. IR systems are often used in libraries, where the documents are typically not the books themselves but digital records containing information about the books. This is however changing with the advent of digital libraries, where the documents being retrieved are digital versions of books and journals. IR systems allow us to narrow down the set of documents that are relevant to a particular problem. As text mining involves applying very computationally-intensive algorithms to large document collections, IR can speed up the analysis considerably by reducing the number of documents for analysis. For example, if we are interested in mining information only about protein interactions, we might restrict our analysis to documents that contain the name of a protein or some form of the verb ‘to interact’ or one of its synonyms.

Natural Language Processing (NLP) is one of the oldest and most difficult problems in the field of artificial intelligence. It is the analysis of human language so that computers can understand natural languages as humans do. Although this Goal is still some way off, NLP can perform some types of Analysis with a high degree of success. For example:

Part-of-speech tagging classifies words into categories such as noun, verb or adjective. Word sense disambiguation identifies the meaning of a Word, given its usage, from among the multiple meanings that the word may have. Parsing performs a grammatical analysis of a sentence. Shallow parsers identify only the main grammatical elements in a sentence, such as noun phrases and verb phrases, whereas deep parsers generate a complete representation of the grammatical structure of a sent.

Data Mining (DM) is the process of identifying patterns in large sets of data. The aim is to uncover previously unknown, useful Knowledge. When used in text mining, DM is applied to the facts generated by the information extraction phase. Continuing with our protein interaction example, we may have extracted a large number of protein interactions from a document collection and stored these interactions as facts in a database. By applying DM to this database, we may be able to identify patterns in the facts. This may lead to new discoveries about the types of interactions that can or cannot occur, or the relationship between types of interactions and particular diseases and so on.
We put the results of our DM process into another database that can be queried by the end-user via a suitable graphical interface. The data generated by such queries can also be represented visually; techniques “mine” large amounts of data, looking for meaningful patterns. DM looks for patterns within structured data, that is, databases. The underlying technologies are based on statistics and artificial intelligence, littering the field with buzzwords such as classification and regression trees (CART), chi-squared automatic induction (CHAID) neural networks and genetic algorithms. Many data mining techniques are available for performing different knowledge tasks in text mining. But all the methods are adopted on term based approach. The primary aim of the text mining Discovering knowledge from large text documents without duplicating the data with semantic meaning of the data but this term based approaches ignoring the relationship between the words present in the documents. Some of the term based methods are Boolean model, vector model and probabilistic model [3].

The rest of this paper presents as follows:

2. Related work
3. Definitions
4. PTM and d-pattern mining algorithm
5. Conclusion

II. RELATED WORK

IR Systems have been an important role for the development of Web search engines, and involved a range of tasks: filtering, classification and question answers. Currently, there are some big research issues in IR and Web search [6], such as evaluation, information needs, effective ranking and relevance. Relevance is a fundamental concept of information retrieval, which is classified into topical relevance and user relevance. The former discusses document’s relevance to a given query and the latter discusses document’s relevance to a user. The popular term-based IR models include the Rocchio algorithm, Probabilistic models and Okapi BM25, and language models, including model-based methods and relevance models. In a language model, the key elements are the probabilities of word sequences which include both words and phrases (or sentences).

Text categorization is the assignment of natural language texts to one or more predefined categories based on their content which is an important component in many information organization and many tasks. Machine learning methods, including Support Vector Machines (SVMs), have tremendous potential for helping people to effectively organize the electronic

SVM is one of the main machine learning methods for text classification. It also performed better on Reuters data collections than kNN and Rocchio. The classification problems include the single labeled and multi-labeled problem. The most common solution [51] to the multiple labeled problem is to decompose it into some independence binary classifiers, where a binary one is assigned to one of two predefined categories.

In real-world situation, however, the great variety of different sources and hence categories usually poses multi-class classification problem, where a document belongs to exactly one category selected from a predefined set. The Data Mining Community have been focused to implementing the effective pattern mining for discovering patterns from large amount collecting data from Text Documents. Clustering is a widely studied data mining problem in the text domain. The Clustering problem is very useful in text domains because text documents contains the unstructured data. So the clustering’s problem of clustering finds applicability for a number of tasks, such as Document Organization, Browsing and Corpus Summarization. Clustering techniques provide a coherent summary of the collection in the form of cluster-digests [83] or word-clusters [17, 18], which can be used in order to provide summary insights into the overall content of the underlying corpus. Variants of such methods, especially sentence clustering, can also be used for document summarization.

A text document can be represented either in the form of binary data, when we use the presence or absence of a word in the document in order to create a binary vector. In such cases, it is possible to directly use a variety of categorical data clustering algorithms on the binary representation. A more enhanced representation would include refined weighting methods based on the frequencies of the individual words in the document as well as frequencies of words in an entire collection (e.g., TF-IDF weighting). Quantitative data clustering algorithms can be used in conjunction with these frequencies in order to determine the most relevant groups of objects in the data. However, such naive techniques do not typically work well for clustering text data. This is because text data has a number of unique properties which necessitate the design of specialized algorithms for the task. The dimensionality of the text representation is very large, but the underlying data is sparse. This problem is even more serious when the documents to be clustered are very short. The number of words (or non-zero entries) in the different documents may vary widely. Therefore, it is important to normalize the document representations appropriately during the clustering task.

A common representation used for text processing is the vector-space based TF-IDF representation. In the TF-IDF representation, the term frequency for each word is normalized by the inverse document frequency, or IDF. The inverse document frequency normalization reduces the weight of terms which occur more frequently in the collection. This reduces the importance of common terms in the collection.
ensuring that the matching of documents be more influenced by that of more discriminative words which have relatively low frequencies in the collection. In addition, a sub-linear transformation function is often applied to the term frequencies in order to avoid the undesirable dominating effect of any single term that might be very frequent in a document the work on document-normalization is itself a vast area of research. Pattern mining has been extensively studied in data mining communities for many years. A variety of efficient algorithms such as Apriori-like algorithms, Prefix Scan, FP-tree, SPADE, SLPMiner and GST have been proposed. These works have mainly focused on developing efficient mining algorithms for discovering patterns in databases. In the field of text mining, but interpreting useful patterns remains an open research problem.

Many effective term-based methods provided by IR to solve this Challenge [28, 24]. The term-based methods have advantages includes Efficient computational performance; as well as mature theories for Term weighting. Phrases have been provide by some IR models, as phrases are more discriminative and carry more “semantics” than finding the useful phrases only for text mining and classification is challenging because phrases are contains statical lesser properties for words and there are number of noisy phrases. so finding clustering for text documents is challenging issue. some researchers proposed closed patterns these are alternative for phrases for removing redundant patterns. There are two reasons for introducing pattern mining algorithms finding the useful information from relevant and irrelevant documents low sustain problem and the misunderstanding that means that measures are not useful for finding the support of pattern. In order to finding an effective pattern for solving problems in the text clustering. this paper presents a words sequence pattern mining for effective clustering of the text documents. It is solving the low frequency problem using d-pattern mining and it reduces the noisy pattern using Inner pattern evolution.

III. DEFINITIONS

In the proposed work we are mining words based on the features selection methods. this methods are documents are considered as input and the features of the results are based on the set of documents are collected. These can be achieves by using Tfidf method.

The basic idea is to assign weights to terms and sentences based on their frequency and some other statistical information. Terms with high weights capture the topical information.

i. Tfidf Weighting:

The tf-idf (term frequency–inverse document frequency) sis calculating terms weights in Text Mining and Information Retrieval. This is a statistical measure used to assess the importance of a word in a collection of documents or corpus. 

Tf-idf weights the frequency of a term in a document with a factor that discounts its significance when it appears in almost all documents. Therefore terms that appear too rarely or too frequently are ranked lower than terms that balance between the two boundaries and, hence, are likely to be better able to contribute to clustering results

II Closed Sequential Patterns

Given a pattern (an ordered termset) X in document d [X] is still used to denote the covering set of X

\[ \text{termset}(Y) = \{ t | \forall dp \in Y \Rightarrow t \in dp \} \]

Its absolute support is the number of occurrences of X in PS(d)

\[ \sup_a(X) = |\text{X}^n| \]

Its relative support is the fraction of the paragraphs that contain the pattern, that is,

\[ \text{Cls}(X) = \text{termset}(\text{X}^n) \]

Its relative support is the fraction of the paragraphs that contain the pattern, that is,

\[ \sup_r(X) = \frac{|X^n|}{|PS(d)|} \]

if \( \sup_a(X_1) = \sup_a(X) \)

\[ X_1 \supseteq X \]

A sequential pattern X is called frequent pattern if its relative support (or absolute support) is a minimum support. Some property of closed patterns can be used to define closed sequential patterns.

IV. PATTERN TAXONOMY MODEL

This model follows two steps in first it describes how to extract the patterns from the text documents. In second step it describes How to update the discovered patterns effectively for performing the knowledge discovery from the text documents. In PTM, here first we split the document into a paragraphs And each paragraph is to be taken as a one document. let us assume a given document is considered as d and it yields PS(d).

Here the meaning of taxonomy a tree structure form so it construct this model in the form of tree structure and it derives from a sub set of relations from a given paragraphs of the Sequential patterns or words in a given text documents.
The System perform preprocessing of text documents for the inputs are given to the PTM. The preprocessing has consists of two steps first one is Stop Word removal and Stemming process

Stop word removal:
Stop words are words which are filtered out prior to, or after, processing of natural language data. They typically comprise prepositions, articles, and so on. There is no specific list of stop words for all applications and these stop words are controlled by the human but not automated.

Stemming Process:
Stemming is the process for reducing inflected (or sometimes derived) words to their stem base or root form. It generally a written word forms. In this preprocess the text documents have to be processed using the Porter stemmer. It removes the Suffix’ of the words these words are useful in the text mining for clustering the text documents in the text mining process we collects the documents and each documents are composed into the set of terms or words. The words having stem have a same meaning in stem process the suffixes of the words, singular and plural words are considered into a one single word for meaning ful text clustering process.

After the preprocessing the Text documents we give the documents input to PTM in PTM we split the given documents into the paragraphs and each paragraph is treats as a individual document which is consists of the words(or) terms. At each document we applies the data mining methods for finding the Sequence words Patterns and it generates taxonomy model.

Table1: Taxonomy Model Paragraphs

<table>
<thead>
<tr>
<th>Paragraph</th>
<th>Terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>dp1</td>
<td>t_1, t_2</td>
</tr>
<tr>
<td>dp2</td>
<td>t_3, t_4, t_6</td>
</tr>
<tr>
<td>dp3</td>
<td>t_3, t_4, t_5, t_6</td>
</tr>
<tr>
<td>dp4</td>
<td>t_3, t_4, t_5, t_6</td>
</tr>
<tr>
<td>dp5</td>
<td>t_1, t_2, t_6, t_7</td>
</tr>
<tr>
<td>dp6</td>
<td>t_1, t_2, t_6, t_7</td>
</tr>
</tbody>
</table>

V. D-PATTERN MINING

Providing the semantic information in the pattern taxonomy to improve the performance of closed patterns in text mining, we need to interpret discovered patterns in order to accurately evaluate term weights (supports). The rational behind this motivation is that discovered patterns include more semantic meaning than the terms that are selected based on a term-based technique (e.g., tf*idf). In term-based approaches, the evaluation of term weights (supports) is based on the distribution of terms in documents. The evaluation of term weights (supports) is different to the normal term-based approaches. As suggested in [1], in deploying method, terms are weighted according to their appearances in discovered closed patterns. It simply deploys patterns through the use of a pattern composition operator.

Let DP be a set of d-patterns in Dp, and p 2 DP be a d-pattern. We call p the absolute support of term t, which is the number of patterns that contain t in the corresponding patterns taxonomies. In order to effectively deploy patterns in different taxonomies from the different positive document.

the problem of this method was the low frequency due to the fact that it is difficult to match patterns in documents especially when the length of...
the pattern is long. Therefore, a proper pattern deploying method to overcome the low frequency problem is needed.

4.1 D-Pattern Method

The pattern taxonomy model improves the semantic meaning of the discovered pattern by using the SPMining, which helps to reduce the search space. The algorithm describes the training process of finding the set of d-patterns. For every positive document, the SP Mining algorithm is first called giving rise to a set of closed sequential patterns. The main focus is the deploying process, which consists of the d-pattern discovery and word support evaluation. Here words supports are calculated based on the words normal forms for all words in the d-patterns.

For each positive document \( d \in D^+ \), a set of patterns are discovered in order to be merged into a dedicated vector

\[
\hat{d} = ((t_1,n_{t_1}), \ldots, (t_n,n_{t_n}))
\]

where \( k \) represents each term support value in order to finding the Sequence patterns in a given text documents. Here each individual pair representing the term and support value from the text documents.

- \( d1 = \{(t1,1),(t2,1),(t3,1),(t4,1)\} \)
- \( d2 = \{(t5,1),(t6,2),(t2,1)\} \)
- \( d3 = \{(t5,1),(t6,1),(t2,1)\} \)
- \( d4 = \{(t1,1),(t3,2),(t7,1)\} \)
- \( d5 = \{(t2,1),(t6,1),(t4,1)\} \)

By using the below formula we can calculate the discovered pattern \( \hat{d} \).

Let \( p1 \) and \( p2 \) are set of number of pairs of document

\[
\hat{d} \subseteq \{(t_1 \in p_1 \cap p_2, n_{t_1} \in p_1 \cap p_2)\}
\]

For Example:

\[
\{(t1,1),(t2,1),(t3,1),(t4,1)\} \cup \{(t5,1),(t6,2),(t2,1)\}
\]

Now the results of \( d1,d2,d3,d4 \) and \( d5 \) patterns results is stored into the discovered pattern \( \hat{d} \). That is

\[
d = \{(t1,12/29),(t2,6/29),(t3,6/29),(t4,4/29),(t5,7/12),(t6,4/29),(t7,7/30)\}
\]

In the above \( t6 \) and \( t5 \) are the highest supporting words in the Documents. These are the sequential patterns appears in the documents.

4.2 Finding Sequence Pattern Mining

Fig 1:D-Pattern Mining Method

<table>
<thead>
<tr>
<th>Doc</th>
<th>Pattern Taxonomy</th>
<th>Sequential Patterns</th>
</tr>
</thead>
<tbody>
<tr>
<td>d1</td>
<td>PT(1,1)</td>
<td>{(t1)}</td>
</tr>
<tr>
<td></td>
<td>PT(1,2)</td>
<td>{(t3)}</td>
</tr>
<tr>
<td>d2</td>
<td>PT(2,1)</td>
<td>{(t5)}</td>
</tr>
<tr>
<td></td>
<td>PT(2,2)</td>
<td>{(t6)}</td>
</tr>
<tr>
<td>d3</td>
<td>PT(3,1)</td>
<td>{(t5)}</td>
</tr>
<tr>
<td></td>
<td>PT(3,2)</td>
<td>{(t6)}</td>
</tr>
<tr>
<td>d4</td>
<td>PT(4,1)</td>
<td>{(t1)}</td>
</tr>
<tr>
<td></td>
<td>PT(4,2)</td>
<td>{(t3)}</td>
</tr>
<tr>
<td>d5</td>
<td>PT(5,1)</td>
<td>{(t2)}</td>
</tr>
</tbody>
</table>
Table 2 represents the each document contains terms and their support values in the document.

VI. RESHUFFLING ALGORITHM

The main theme of this algorithm is finding useful information from a set of negative documents because the negative documents may contain the some useful information. To do this algorithm describes how to re-shuffle for finding the useful terms available in the negative documents. The main process of Reshuffling is implemented by the algorithm IPEVolving: this algorithm follows two steps: first one is the inputs of this algorithm is set of the discovered patterns find by using the d-pattern method: a training set $\mathbf{D} = \mathbf{D'} \cup \mathbf{D''}$. 

Second one is it identifies the relevant and irrelevant documents based on the specified threshold value for finding the negative documents and this useful to reduce the side effects on the documents i.e. noisy patterns. in this algorithm the similarity between the documents and the relevance between the are calculated by the following formula $R(d)=d,v$

For assigning the weights to the documents the weights are calculated

$$\text{weight}(d) = \sum_{t \in T} \text{support}(t) \tau(t, d).$$

For identifying the relevant documents this algorithm works based on the offenders. it follows two types of offenders there are conflict offender and partial offender if it is conflict offenders it removes the d-patterns and partial offenders.

In this algorithm first step is calculating the threshold value for identifying the relevant and irrelevant documents if the weight of the negative document (nd) is greater than the threshold then it calculates the offenders after that it performs the reshuffling based on the offenders.

CONCLUSION

Many data mining techniques have been proposed for performing the different knowledge tasks but all the existing methods adapted based on the term support. The prime aim of the text mining is to identify the useful information without duplication from various documents with synonymous understanding. But those methods working based on the term support that methods are ignoring the relationships between the terms. In order to enable an effective clustering process, the word frequencies need to be normalized in terms of their relative frequency of presence in the document and over the entire collection. This paper presents an innovative approach for relevance feature and it performs the words normalization in the documents in order to reduce the duplication of the words in the documents and it performs pre-processing before going load the documents into the database. the pre-processing reducing the unnecessary data in the documents. It also proposes method to revise low-level discovered patterns in the documents and their categories. it also solved the misapprehension problem by using reshuffling algorithms.

REFERENCES


