TEXT MINING USING KEYPHRASE EXTRACTION

Shobha S. Raskar
Bharati Vidyapeeth University, College of Engineering BVUCOE ,Dhankawadi,Pune

D. M. Thakore
Bharati Vidyapeeth University, College of Engineering BVUCOE, Dhankawadi ,Pune

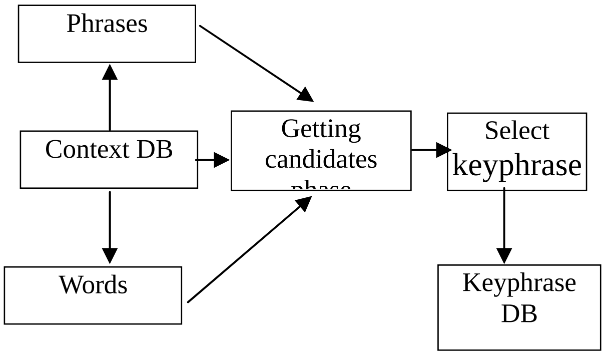
Abstract :

Text mining is powerful tool to find useful and needed information from huge data set. For context based text mining, keyphrases are used. Keyphrases provide brief summary about the contents of documents. In document clustering, number of total cluster is not known in advance. In K-means, if prespecified number of clusters modified, the precision of each result is also modified. Therefore Kea ,is algorithm for automatically extracting keyphrases from text is used. In this kea algorithm, number of clusters is automatically determined by using extracted keyphrases. Kea-means clustering algorithm provide easy and efficient way to extract test document from large quantity of resources. Keyphrase play important role in text indexing, summarization and categorization. Keyphrases are selected manually. Assigning keyphrases manually is tedious process that requires knowledge of subject. Therefore automatic extraction techniques are most useful.

Keywords: Clustering, Keyphrases, kea algorithm

1. INTRODUCTION

Kea algorithm used for automatically extracting keyphrases from text. Keyphrase provide a brief summary of document contents. Keywords and keyphrases are particularly useful. They can be used in information retrieval system as description of the document returned by a query. Keyphrases help the users to get idea about the content of document. The word *keyphrase* implies two features: *phraseness* and *informativeness*. Phraseness is a somewhat abstract notion which describes the degree to which a given word sequence is considered to be a phrase. In general, phraseness is defined by the user, who has his own criteria for the target application. One user might want only noun phrases while another user might be interested only in phrases describing a certain set of products. Informativeness refers to how well a phrase captures or illustrates the key ideas in a set of documents.Because informativeness is defined with respect to background information and new knowledge., Infomativeness uses relationship between *foreground* and *background*. The target document set from which representative keyphrases are extracted is called the foreground corpus. The document set to which this target set is compared is called the background corpus. In order to get a ranked keyphrase list, both phraseness and informativeness combined into a single score. A sequence of words can be a good phrase but not an informative one, like the expression



(Diagram 1.1: Workflow of keyphrase extraction)

In this figure, phrases and words are extracted from context database. And candidate phase starts. Next step is to select keyphrase from keyphrase database.

2. CLUSTER

It comprised of number of similar objects collected or grouped together. The basic members of k- mean type algorithm family include K-Means, K-Modes and K-Prototypes. K-Means algorithm is used in value data, K-Modes algorithm is used in attribute data, and K-Prototypes algorithm is used in mixed data of value and attribute. The k-means type algorithm has such advantages as fast speed, easy realization and so on. It is suitable for those kinds of data clustering analysis as in text, picture characteristic and so on.

2.1 The K-Means Clustering:

The K-means clustering algorithm is one of the simplest clustering algorithms in which the number of clusters to be grouped is fixed by the user. The algorithm proceeds by randomly defining *k* centroids and assigning a document to the cluster that has the nearest centroid to the document. In general, obtaining the nearest centroid for a given document and re-calculating new centroids use the cosine measure or the Euclidean distance measure. In K-Means clustering algorithm values of *k is given and k-means* algorithm is implemented in 4 steps:

1. Partition objects into *k* nonempty subsets.
2. Compute seed points as the centroids of the clusters of the current partition. The centroid is the center (mean point) of the cluster.
3. Assign each object to the cluster with the nearest seed point. Go back to Step 2, stop when no more new assignment.

2.2 Advantage K-mean :

Rapidity, simplicity, high scalability, easy realization, fast speed, improves efficiency.

2.3 Disadvantage:

Numbers of cluster should known in advance. In K-means, if pre-specified number of clusters modified, the precision of each result is also modified.

3. KEA ALGORITHM

Keywords and keyphrases are widely used in large document collections. They describe the content of single documents and provide a kind of semantic metadata that is useful for a variety of purposes. The task of assigning keyphrases to a document is called *keyphrase indexing*. KEA is an algorithm for extracting keyphrases from text documents. It can be either used for free indexing with a controlled vocabulary. Kea, an algorithm for automatically extracting keyphrases from text. Kea identifies candidate keyphrases using lexical methods, calculates feature values for each candidate, and uses a machine- learning algorithm to predict which candidates are good keyphrases. In addition, keyphrases can help users get idea about the content of a collection, provide sensible entry points into it. In the specific domain of keyphrases, there are two fundamentally different approaches: *keyphrase assignment* and *keyphrase extraction*. Keyphrase assignment seeks to select the phrases from a controlled vocabulary that best describe a document. Keyphrase extraction, the approach used here, does not use a controlled vocabulary, but instead chooses keyphrases from the text itself. Kea’s extraction algorithm has two stages:

1. *Training:* create a model for identifying keyphrases, using training documents where the author’s keyphrases are known.
2. *Extraction*: choose keyphrases from a new document, using the above model.

Both stages choose a set of *candidate phrases* from their input documents, and then calculate the values of certain attributes (called features) for each candidate [3]. Training stage uses a set of training documents for which the author's keyphrases are known for each training document, candidate feature values are calculated. Discard any phrase that occurs only once in the document to reduce the size of the training set. To select keyphrases from a new document, kea determines candidate’s phrases and feature values and then applies model built during training. The model determines the overall probability that each candidate is a key-phrase and then a post-processing operation selects the best set of keyphrases.

3.1 Candidate phrases:

Kea chooses candidate phrases in three steps.
It first cleans the input text, and then identifies candidates, and finally stems and case-folds the phrases. Following are rules,

1. Candidate phrases are limited to a certain maximum length (usually three words).
2. Candidate phrases cannot be proper names (i.e. single words that only ever appear with an initial capital).
3. Candidate phrases cannot begin or end with a stopwords.

4. FEATURE CALCULATION

Two features are calculated for each candidate phrase and used in training and extraction. They are TF X IDF, measure of phrase's frequency in document. Different schemes for assigning term weight w^{ij} have been proposed.

The standard version is TF and then is refined as TF X IDF,

TF is Frequency of term t^{i} in document $d^{j.}$ IDF is Inverse document frequency of term ti in corpus .

TF X IDF can select terms which occur frequently in parts of document but appear rarely in corpus. So these terms have abilities to distribute documents into different clusters. TF X IDF , this feature compare frequency of phrases use in particular document with frequency of that phrase in general use. DF means document frequency is number of document containing the phrase in some large corpus. TF X IDF for phrase P in document D is ,

TF X IDF = $\dfrac{freq(P,D)}{size(D)} X - \log \dfrac{df(p)}{N}$, where,..................................Equation 1

1. frequency (P,D) is the number of times P occurs in the document D
2. Size (D) is the number of words in D
3. df(P) is number of document containing P in the global corpus.
4. N is size of global corpus.

The second term in the equation is the log of the probability that this phrase appears in any document of the corpus(negated because the probability is less than one), if document is not part of the global corpus, df(p) and N are both incremented by one before the term is evaluated to simulate the its appearance in the corpus.

4.1 Mathematical definition.

The Euclidean *distance between two points/objects/items defined by point X and points is Y is defined by Equation 1A below:*

Euclidean *distance* $(X,Y) = (|X_1-Y_1|^{2} + \ldots |X_{N-1}-Y_{N-1}|^{2+} \quad |X_N-Y_N|^{2})^{½}$.............................. *Equation 1A,*

Another distance measure is the Manhattan ***or*** City Block *distance measure which is defined, for the distance between two data points X and Y as Equation 2A below:*

Manhattan distance$(X,Y) =(|X_1-Y_1| + \ldots\ldots.. |X_{N-1}-Y_{N-1}|^{+} \quad |X_N-Y_N|)$.............................. *Equation 2A*

Kea algorithm uses both *Euclidean distance and cosine measure values*.

5. **OVERALL PERFOMANANCE : (Table 5.1** : Performance)

Keyphrase Extracted	**Avg matches with Author keyphrases**
5	0.93
10	1.39
15	1.68

Table 5.2 :Effect of document length

Document length	**Average # matches (5 extracted)**	**Average # matches (15 extracted)**
Full Text	0.91	1.71
Abstract	0.66	1.03

This shows number of correct keyphrases extracted using both short and full document. Kea extracts fewer keyphrases from abstracts than from full document text.

6. ANALYSIS

Performance depends on size of training document and document length. Performance is better for relatively small set of training document and kea extracts less keyphrases from small documents than from full document text. Keyphrase extraction is powerful tool for text summarization and similarity analysis. Feature values are useful for phrase's frequency in document.

7. CONCLUSION

Kea-mean clustering tries to save the main drawback of k-mean that number of total clusters is pre-specified in advance. Kea-means clustering algorithm provide efficient ways for extracting test document from large quantity of resources. As k-means algorithm uses cosine measure or Euclidean distance measure to calculate feature values and kea-mean algorithm uses these two measure simultaneously. Kea extract fewer keyphrases from short text as compare to large text. This kea- is powerful and easy way to extract keyphrase from documents. Keyphrase extraction is powerful tool for text summarization and similarity analysis. Keas performance is good for application such as browsing searching and clustering. K-mean algorithm is not powerful algorithm as compare to kea-mean algorithm.

REFERENCES:

[1] Jeffrey L. Solka,Naval Surface Warfare Center Dahlgren Division, Attention Jeff Solka, 18444 Frontage Rd,Suite 328 Dahlgren Virginia, 22448-5161, 'Text data mining:Theory and methods'
[2] CollabRank: Towards a Collaborative Approach to Single-Document Keyphrase Extraction' Xiaojun Wan and Jianguo Xiao Institute of Computer Science and Technology, Peking University, Beijing 100871, China
[3] Ian H. Witten, Gordon W. Paynter , Eibe frank, carl Gutwin, university of waikato, Newzealand,'KEA: practical Automatic keyphrase extraction'
[4] Simona Balbi, Emilio Dimeglio, Dip. Di mothe statistica university, federico, ildi Napol, 'Text mining strategy, based on local context of words.
[5] N. Kanya, S. Geetha, 'Information extraction Text mining Approach'