Bangladesh J. Sci. Res. 24(2): 117-126, 2011 (December)

INFORMATION RETRIEVAL WITH TEXT MINING FOR DICISION SUPPORT SYSTEM

Mahmuda Rahman*

Institute of Information Technology, University of Dhaka

Dhaka-1000, Bangladesh

Abstract

Getting the proper contextual clues from a body of text provided to any automated system is a difficult but open challenge. It is vital for the knowledge base on which large decision support systems reside. The level of accuracy positively affects the critical factors of such a module. This paper discusses the methodology adopted for the novel area of Text Mining to retrieve information from a critical system and the result has been considered to measure to what extent it can be fit into the large scale Decision Support System (DSS).

Key words: Natural language processing, C4.5 classification, DSS, machine learning, KNN clustering, SVM

Introduction

By definition text mining means method of term extraction and text categorization from a document. Electronic publication facilitates a huge amount of information readily available to the readers on-line. Searching for specific values over a large amount of search document must depend on good design of query processing. As the journal papers and scientific magazines also use natural language to describe the content, to explore the information out of it automatically, people depend a lot on reasoning. Logical inference and deductions are intuitive methods that human use to grab the "materials" out of it. As for an automated system, to make it do that requires a high degree of Human Computer Interaction (HCI). Otherwise it seems quite easy to get overwhelmed with too many results. Information Retrieval (IR) from documents is supported by the query generated from the Boolean combination of key words. To get the relativeness of the document according to the interest of the user, clustering is often used with the distance function to group the documents and it eventually boils down to the mining of text to be a pattern recognition problem.

Data preprocessing

For the sake of simplicity, if the following example is considered it might give a clear picture about mining the text:

"Agatha Christie was famous for her murder mysteries..."

*E-mail: mahmuda@univdhaka.edu

The query request for <*author*> and <*genre*> should populate the user knowledge base (can be a table in a Relational Database with the appropriate value extracted from this line of text (Weiss *et al.* 2005) For this, extensive preprocessing is required and in terms of text mining from natural language processing (NLP) this is popularly called "Corpus Filter". This is occasionally done by the conventional steps.

Stop Word Removal

Despite of their high frequency of occurrence in any given document, some common words are of no significance in case of generating search terms, such words like a, an, the, on, at, in of preposition and articles can be removed from the body of the text.

Stemming/Suffix removal

Stemming is done for identifying the root or stem of the word to cluster them accordingly. For example 'walk' can be taken as a stem for the word 'walking', both search meta and the target text needs to be stemmed to get a match.

Parts of Speech Tagging

Sometimes it is really important to tag the text according to the parts of speech each word belongs to. This type of 'binning' helps to figure out the context easily forms the annotated text. Stanford Parser is one of the most popular one in this regard.

Lemmatisation

Grouping of words with respect to similarity in meaning and attitude or tone makes a difference in text mining as it can sense out the rest of the message that is being conveyed throughout the documents. For example 'Better' has "Good" as Lemma for lemmatisation.

Duplication removal with Jaccard Index

Duplication of data is a potential noise to the data set we hold, it also creates unexpected delay in text processing. For the similarity measurement Jaccard Index is used with scaling function F_n for 2 documents D_1 and D_2:

$$J\,(A, B) = \frac{A\ .\ B}{A\ U\ B}$$

$A \Rightarrow Fn\ (D_1)$

$B \Rightarrow Fn\ (D_2)$

Jaccard Distance for mismatch:

$1 - J\,(A, B)$

Method of Classification

For some problem the ultimate target is to define a knowledge base for the automated system so that it can decide considering the given textual information. For this, text format is somewhat not appropriate to process data to conclude with a specific result. For this reason, data needs to be formatted into a table after initial preprocessing form where we can employ C4.5 and come up with a Decision Tree (DT) by induction.

Table 1 is showing the placement of words W_i's as feature attributes for DT induction from documents D_i's. But the main challenge in feature selection, where the document content is vast this matrix eventually becomes a sparse one. This has to be normalized numeric values mapped through a predefined codebook so that Jaccard Index can be measured for these entries.

Table 1. Document by word (term) matrix format.

	W_1	W_2	W_3	Result
D_1				
D_2				
D_3				
D_4				

Role of Clustering

Clustering is used to help in selection of feature to avoid the problem of huge search space, it somewhat narrows down the domain for it is equivalent to assigning labels for text categorization. To predict where to place a new document, as discussed before, Extended Jaccard Index (Kloesgen ansd Zytkow 2002) is used with the cosine function to get the cosine similarity measure. It has some good application like placing a new document in the filter of an appropriate directory or forwarding a mail to a marked folder.

Measurement Criteria

Classical Data Mining supports the following measurement techniques which can also be deployed on text mining if we view it as a pattern classification problem:

For Pattern Classification:

T_p => true +ve
F_p => false +ve
T_n => true -ve
F_n => false -ve

Measurement Matrics:
Precision = T_p / (T_p + F_p)
Recall = T_p / (T_p + F_n)
F-measure = 2 X (precision.recall)/ (precicion + recall)

Uniqueness of Text Mining in a DSS

There are some unique quality of Text Mining which makes it a little different than the conventional data mining, they are:

1. Sparse feature space
2. Association rule generation might not be not sufficient
3. Each document is a vector of words
4. Need automated construction of Text Classifiers
5. We can alternatively use normalized terms (lemmatized) words

Term Extraction for DSS

Each feature is called a term and term extraction from a given document is a critical decision for designing the automated system which can process them. The following procedure can be adopted for extracting useful candidate terms form a collection of documents. Figure 1 reflects this.

1. Reading from the text
2. Linguistic Preprocessing with NLP techniques as described before
3. Generate Candidate terms with standard pattern mining and association rules
   3.1. Candidate terms are chosen based on their relevant morph syntactic pattern
   3.2. They are then combined according to the association coefficients
   3.3. Co-occurrence frequency, phi square, association ratio, log-likelihood are used as common techniques
4. Filter out the insignificant terms with the help of metrics decided for IR
   4.1. It runs into the risk of over generation of terms
   4.2. It can reduce the candidate and take top M ones
   4.3. It considers deviation Based Approach (remove uninteresting words)
   4.5. Sometimes takes into account the statistical Significance Approach (relative frequency with chi square)
   4.6. Some useful methodologies are Info Retrieval Approach (tf-idf: term frequency-inverse doc frequency score for maximal gain)

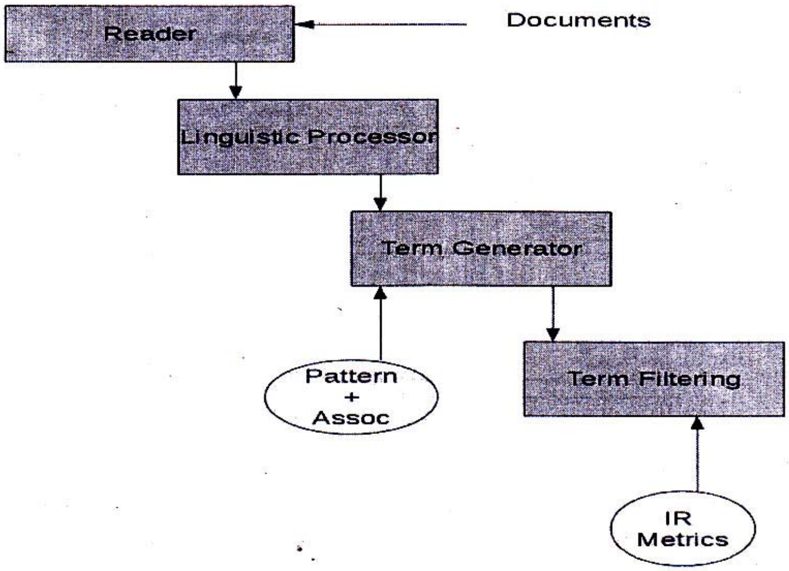

Figure 1. Architectural Module for Term extra

Usage of Text Mining in a DSS

Text mining is a very handy tool for the following areas in a DSS:

1. For Natural Language Processing
2. Finding trend and relations between documents and terms
3. Classify documents by category
4. Retrieve appropriate document by search
5. Cluster documents according to their content

Figure 2 shows these components.

Machine Learning Approach

By definition, ML approach to Text mining has the following rules:

Where, $D = \{d_1, d_2, ... d_n\}$ is a set of training doc collection and $C = \{c_1, c_2, ... c_n\}$ is a set of possible category and $T = \{t_1, t_2, ... t_n\}$ is a set of terms appeared if $CSV_{(i)}$ denotes the certainty that cat c_i is assigned to $d_j[0,1]$ then $Dis(d_i,d_j)$ = distance between docs d_i and d_j according to Bayes theorem: $P(c_i|d_j) = [P(c_i).P(d_j|c_i)]/P(d_j)$ gives us $P\ (d_j|c_i)$ = product$[P(w_{kj}|c_i)]$ when w_{kj} is the word belonging to kth row and jth column of Table 1.

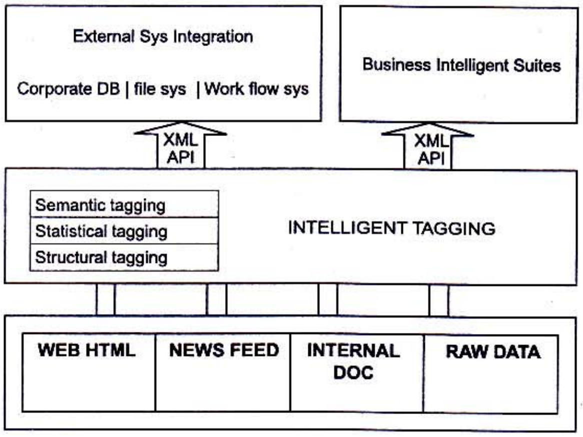

Figure 2. Modules of Text Mining System.

KNN Computing

The adopted ML approach gives way to KNN clustering providing that $CSV_i(d_j)$ = $SUM[Dis(d_j,d_z).C_i(d_j)]$ where D_z subset $Trk(d_j)$ an Support Vector Machine (SVM) can be used meaning that a hyperplane separating maximum margin distances between +ve and -ve sets of data feed to the system.

Other Alternatives

Some other alternative approaches are:

Hidden Markov Model

The model is based on state transition and symbot emission

Stochastic Context free Grammar

Defined as, G = (T, N, S, R, P) where T = alphabet and N= Non terminals gets S= start on R= Rules set where P= Probability, by this procedure, we adopt similar steps as we do for a finite state machine for verification of a programming language.

Maximal Entropy Modeling

By this model we use statistics of a random process generating y given x, the conditional probability help to infer posterior probability occurrence of a term given the prior probability of occurrence of some other terms.

Text Mining Applied

Text mining has been successfully applied to medical journal sites to process the abstracts of papers on similar biological problems. Poon (Poon and Domingos 2008, 2009) recision and recall than the other methods adopted earlier. Silvui Cuecerzan (Jain *et al.* 2008) compared context of Wiki with content of the document stored there with the help of category tags and declared a good result. Where he took for Id to be the Mentioned Entity (Surface form) and their label (needs co-reference resolution) using RegExps for: Enamex, Times and Numex extensively for entity, temporal and numeric terms and got a mentionable achivement in Structural Disambiguation such as to understand "Alliance of Democracy in Mali"is one entity instead of 3 or more. He did the Entity Labeling where an ambiguity like "Washington" – is a person or place could get resolved. For this he used 3 clues:

1. Entity: Surface form (Concept)
2. Redirection (Category Tags)
3. Context (reference to and from Wikipedia page)

And did the document analysis assuming that it T_0, T_1 and T_2 are terms then if recursively T_0 -> T1 & T2 where '&' stands for appropriate conjunction, we can refine the search space looking up for T0 as well as "T1& T2" to measure the relative implication on the search result to choose which one is better.

So he searched for the left hand side and then combination of right and measures the match "Bush", "W. Bush", "George Bush" for co reference resolution deducing they both are the same entity. The statistical base behind this formulation depends mainly on e (entity), d (document), C (category) and T (terms) where maximum value has been taken from the Figure 3.

$$\underset{\substack{(e_1,\text{----},e_n)\in \\ \in(s_1)\text{x.x}\in(s_n)}}{\arg\max} \sum_{i=1}^{n} < \delta_{e_i}|_C, d > + \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j\neq i}}^{n} < \delta_{e_i}|_T, \delta_{e_j}|_T >$$

Figure 3. Formula for maximal match.

The experiment could successfully tag almost all major terms according to their context and category information available in Wikipedia

Recent Work

Recently researchers are getting keen about using Latent Dirichlet Allocation (Blei *et al.* 2002) which is a Generative probabilistic model for text corpora collection using 3 level Bayes where Bayes parameter estimation is done by variation method and Expectation Maximization (EM) algorithm. Here, a word is the basic unit of discrete data, defined to be an item from a vocabulary indexed by {1, . . . ,V }. We represent words using unit-basis vectors that have a single component equal to one and all other components equal to zero. Thus, using superscripts to denote components, the vth word in the vocabulary is represented by a V -vector w such that wv = 1 and wu = 0 for u = v and a document is a sequence of N words denoted by w = (w_1 , w_2 , . . . , w_N), where wn is the nth word in the sequence whereas a corpus is a collection of M documents denoted by $D = \{w_1, w_2, . . . , w_M \}$.

The pictorial illustration of matrix and vectors are depicted by figure 4.

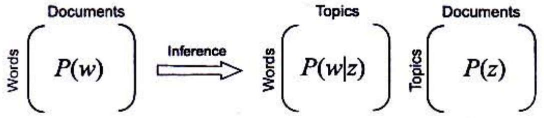

Figure 4. Relation between document, word and topic.

Thus in this model, Unsupervised Bayesian inference used to estimate parameters of LDA to:

1. Extract set of topic and
2. Estimate topic distribution
3. Topic-word distribution for each topic
4. Probability is assigned based on iterative sampling

This has been tried on a number of Electronic journal publications with respect to defined terms and performance is compared against predefined group level and individual document context. And finding shows that group level turns out to have a better result. Table 2 projects the output.

Table 2. Comparison between Document and Group level on terms detection for journals.

Identity group	Number of example	Document level accuracy (%)	Group Level accuracy (%)
ACM CSUR	210	88.91	96.91
AVI	100	74.3	98.06
CACM	100	80.11	91.77
CGIT	120	82.01	100
IEEE Comp Graphics	130	80.87	94.4
IEEE Symp on InfoVis	1520	77.49	87.34
IEEE Transaction	130	71.81	87.6
IEEE Visualization	320	80.03	90.94
LNCS	220	95.4	97.12
SIGCHI	210	87.35	90.06
UIST	170	93.68	90.73
Other	1060	75.31	80.89
Overall	4290	79.74	88.7

A popular NLP tool named LingPipe which can also be used to mine the semantic web is based on LDA.

Conclusion

This paper conveys the application of Text Mining, its architectural modules and components with the description of the mathematical terms related to its mechanisms. Then it tries to give the reader a respectable idea about its usability in system automation and information retrieval. On the way of doing that, it dedicates itself to discuss the interesting problem and of its integration to strengthen Decision Support Systems (DSS). Finally, it exhibits some results cumulated from their simulations to exhibit to what extent this mechanism can be used for large scale DSS.

References

Blei, D. M., A. Y. Ng and M. I. Jordan, 2002. Latent Dirichlet Allocation. University of Berkeley CA and Stanford University, *Journal of Machine Learning Research*, **3:** 993-1022.

Jain, A., S. Cucerzan and S. Azzam. 2008. Augmenting Wikipedia with Named Entity Tags. *The Third International Joint Conference on Natural Language Processing* (IJCNLP), Hyderabad.

Kloesgen, Willi and J. Zytkow, (eds.) 2002. Handbook of Data Mining and Knowledge Discovery, Oxford University Press, Oct 2002 pages 33-60. ISBN 0-19-5118316.

Poon, Hoifung and P. Domingos. 2008. Joint unsupervised coreference resolution with Markov logic. *In Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 649–658, Honolulu, HI. ACL.

Poon, Hoifung and P. Domingos. 2009. Unsupervised semantic parsing. *In* Proceedings of the 2009 Conference on Empirieal Methods in Natural Language Processing, pages 1–10, Singapore. ACL.

Weiss, S. M., N. Indurkhya, T. Zhang and F. Damerau, 2005. Predictive Methods for Analyzing Unstructured Information, XII, 236 page 79 illus. ISBN 978-0-387-95433-2

Manuscript received on 23 July, 2011; revised on 10 August, 2011