Latent Semantic Analysis: An Approach to Understand Semantic of Text

Ms. Pooja Kherwa[1], Dr.Poonam Bansal[2]
Maharaja Surajmal Institute of Technology
Affiliated to GGSIPU, New Delhi

[1]***Abstract*— Latent semantic analysis (LSA) is a method for analyzing a piece of text with certain mathematical computation and analyzing relationship between terms in the documents, between the documents in the corpus..Various application of intelligent information retrieval, search engines, internet news sites requires an accurate method of accessing document similarity in order to carry out classification, clustering, summarizing or search tasks.**

So in this paper we are studying latent semantic analysis based on single value decomposition. The aim of Latent semantic analysis is to exploit the global structure of documents. The emphasis of latent semantic analysis is to find hidden relationship in document for better understanding the relationship between terms and document in dataset. In this paper, we have conducting a study using Latent semantic analysis (LSA) to find correlation of terms in a dataset consisting of research papers of various natural language processing applications.LSA shows that single value decomposition collapse multiple terms with same semantic and can identify terms with multiple meaning and represent documents in lower dimensional conceptual space.

Keywords- Latent Semantic Analysis; Singular Value Decomposition; Conceptual space; Semantics;

I INTRODUCTION

For intelligent information retrieval, recommendation, similarity, visualization, summarization and for many other NLP applications where extracting relevant information is very important, the underlying approach of search should be based on the actual semantic content of the documents instead of just keyword matching. The conceptual search is the need of time.. In this scenario, LSA models called as bag-of-words (not concerned with word order) are adequate for many applications [2,4].

LSA is a statistical technique for analyzing a piece of text for extracting the concept of document by aggregating the different word usage in document collection. In LSA a truncated vector space model is made that represent terms and documents in a particular high dimensional vector space. The Vector space model is made from single value decomposition (SVD) that decomposes a term by document matrix into a product of three simpler matrices [7].

So, latent semantic analysis is complete mathematical/statistical technique for retrieving and revealing hidden relations of contextual usage of words in document .It is not a traditional natural language processing or artificial program, it uses no human made dictionary, ontology, grammars, syntactic parser, or morphologies etc.

In this paper, LSA is used for find association of terms in semantic spaces, with user queries using some natural language processing research papers where LSA is used as a major technique for developing applications. The goal of LSA is extracting semantic relationship between terms of documents, analyzing results of correlation with different values of truncated semantic space. These semantic relation will be helpful for the researcher in finding appropriate techniques to make hybrid models for intelligent information retrieval, topic modeling and merging of similar taxonomies etc. It will again help researchers in finding new research direction in the area of Natural Language Processing. The paper is organized as follows. In section 2 methodology of latent semantic analysis is explained, in section 3 our experimental dataset and approximation of matrix with K singular values are done. In the section 4, results are explained and papers is concluded in section 5.

II METHODOLOGY OF LATENT SEMANTIC ANALYSIS

A. Single Value Decomposition

The Single value Decomposition (SVD) [5] decomposes matrix 'A' into orthogonal factors that represents both types and documents. Vector representation for both types and documents are achieved simultaneously.2.,The SVD sufficiently captures the underlying semantic structure of a collection .3. It allows for adjusting the representation of types and documents in the vector space with optimal no. of reduced dimension. And finally the computation of the SVD is manageable for large datasets,

The matrix we are interested is term by document matrix(TDM) C is M*N .where M #N, furthermore ,C is very

978-1-5386-3243-7/17/$31.00 ©2017 IEEE

unlikely to be symmetric ,because no. of terms will always will be much more than the no. of documents in the dataset.

Given C, Let U be the M*M matrix whose column are the orthogonal eigenvectors of CC^T, and V be the N*N matrix whose column are the eigenvectors of C^TC .
Where C^T the transpose of a matrix C.

SVD Theorem: Let r be the rank of the M*N matrix C. Then the singular value decomposition (SVD) of C of the form

$$C = U\sum V^{T,}$$

Where

1. The Eigen values $\lambda_1..\lambda_r$ of CC^T are the same as the Eigen values of C^TC;[3].
2. For $1 < i < r$, let $\sigma i = \sqrt{\lambda i}$ with $\lambda i > \lambda i+1$;Then the M*N matrix $\sum$ is composed by setting $\sum Ii = \sigma i$ for $1<i<r$,and zero otherwise[3].

B. Local and Global weighting:

LSA applies both a local and global weighting function to each nonzero element a_{ij} in term document matrix ,in order to increase or decrease the importance of terms within documents(known as local)and in entire document collection (known as global).The local and global weighting function for each element ,a_{ij} are usually indicates how frequently a term occurs within a document and inversely related to how frequently a term occurs in documents across collection[6].

So, a_{ij}= local(i,j)*global(i),where local(i,j) is the local weighting for term i in document j ,and global(i) is the terms global weighting.[9].local weighting functions include binary frequency(1 if term is present in the document and 0 if the term is not in the document)and log of term frequency plus 1.Global weighting function include normal, gfidf, idf, and entropy[3], all of which basically assign a low weight to term occurring often or in many documents.

C. Algorithm for LSA:

1. Get a collection of text files related to a specific domain or multiple domains.
2. Merge all files into a single corpus or directory.
3. Create a term document matrix, the cell A_{ij} in this matrix represent no. of occurrence of term j in document i.
4. To reduce the effect of most commonly used words in the corpus. A common weighting method is" log entropy", is applied to each cell in the term document matrix.
5. The output of SVD is reduced to factor k which represents the desired number of dimensions. SVD decomposes the original matrix into three matrices, which when multiplies together, produced the original term document matrix. But for space optimization, it is required to compute only the most significant k dimension.

III EXPERMENTAL SETUP WITH CORPUS

D. Creating a Corpus of document

In this experiment articles from different areas of natural language processing are chosen for dataset. All the articles are using LSA as a technique for implementation. We choose article including different area of NLP with different topic, So we can detect the semantic relation between them. These semantic relation will be helpful for the researcher in finding appropriate techniques to make hybrid models for intelligent information retrieval , topic modeling and merging of similar taxonomies etc. It will again help researchers in finding new research direction in the area of NLP.

In the dataset we have collected ten articles including different application areas of NLP. In the dataset only title and abstract of documents are including keep the experiment as precise as possible.

D1: Discovering biomedical semantic relations in PubMed queries for information retrieval and database curation.
D2: A Latent Semantic Indexing-based approach to multilingual document clustering.
D3: Detecting Cyberbullying using Latent Semantic Indexing
D4: Classification of Genomic Sequences by Latent Semantic Analysis
D5: Latent Semantic Analysis
D6: Latent Semantic Indexing for Patent Documents
D7: A Research Taxonomy for Latent Semantic Analysis-Based Educational Applications.
D8: Generic Text Summarization Using Relevance Measure and Latent Semantic Analysis.
D9: TopicView: Understanding Document Relationships Using Latent Dirichlet Allocation Models.
D10: TopicView: Visual Analysis of Topic Models and Their Impact on Document Clustering.

The collection of documents or corpus was converted into a term-document matrix (TDM) using R. Initially, the corpus was a 263*10 matrix. In order to reveal the semantic relationship in the corpus, the corpus has great amount of noise in it. The noise should be removed and after that Single value Decomposition of the term-document matrix is done. In this study latent semantic analysis (LSA) discovers the relationship of these documents and relationship between the documents.

E. Pre-processing The Corpus

In text mining application pre-processing is the most desired as well as crucial step, in this step searching for noise in data is important. Theses should be removed.
Pre-processing methods are defined as:

1. Stemming: This process removes suffixes from words to make it simple and to get the common origin- like class is retained from classification and classify.
2. Stop words are any common words like “as”, ”the”, ”which”, ”and”, ”at” etc. These words and other words like most commonly used in each documents are removed from the data.
3. All the numbers and punctuation marks are also deleted.

A new term document frequency matrix A was created using R’s tm package. Pre-processing of dataset is very important to deduce the semantic similarity efficiently, because the low frequency of these words creates a sparse matrix.

Table 1. A Sample of Term by document frequency matrix (62*10)

Terms.	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
Biolog	1	0	0	2	0	0	0	0	0	0
Cyberbulli	0	0	2	0	0	0	0	0	0	0
Dictionary	0	0	2	0	0	0	0	0	0	0
Document	0	0	0	2	0	5		2	2	8
Identify	2	0	0	3	0	0	1	1	0	0
Latent	2	0	1	2	1	3	1	1	1	2
Lda	0	0	0	0	0	0	0	0	4	5
Lsa	1	0	0	2	5	0	3	0	0	5
Patent	0	0	0	0	0	5	0	0	0	0
Semant	7	0	1	3	1	3	1	2	0	0
Taxonomi	0	0	0	0	0	0	2	0	0	0
Topicview	0	0	0	0	0	0	0	0	1	3
Topic	1	0	0	0	0	0	0	0	2	4
vsm	0	0	0	0	0	2	0	1	0	0
weight	0	0	0	0	0	0	0	1	0	2
system	1	0	3	0	0	1	0	0	0	0

C. Extracting semantics from documents:Between the documents and between the terms.

In R, we find the associate() function of the LSA package[1], more appropriate and this function arrange the similarity values in descending order. In this experiment to show the power of latent semantic analysis, we have compared the results with simple vector space of term &documents.

Table 2. Correlation analysis of Sample Terms in Semantic space with different values of K.

Term	Correlation Threshold	Correlation Measure	Semantic Space	Term Returned with similarity score
Biolog	0.7	cosine	K=10	identifi (0.92) dna(0.89) evolutionarily(0.89) structur (0.89) techniqu (0.80)
Biolog	0.7	cosine	K=9	identifi (0.92) dna(0.89) evolutionarily(0.89) structur (0.89) techniqu (0.80)
Biolog	0.7	cosine	K=8	identifi (0.92) dna(0.89) evolutionarily(0.89) structur (0.89)techniqu (0.83)
Biolog	0.7	cosine	K=7	identifi (0.92) dna(0.89) evolutionarily(0.89) structur (0.89)techniqu (0.83)
Biolog	0.7	cosine	K=6	identifi (0.96) semant (0.92) dna(0.83) evolutionarili (0.83) structur(0.83) technique(0.72) chemical’(0.72) biocur(0.72) biomed (0.72) gene (0.72) literature(0.72) pubm (0.72)
cyberbulli	0.7	cosine	K=10	Dictionary(1) emoticon(1) preprocess(1) system(0.90) depend (0.70)
cyberbulli	0.7	cosine	K=9	Dictionary(1) emoticon(1) preprocess(1) system(0.90) depend (0.70)
cyberbulli	0.7	cosine	K=8	Dictionary(1) emoticon(1) preprocess(1) system(0.90) depend (0.70)
cyberbulli	0.7	cosine	K=7	Dictionary(1) emoticon(1) preprocess(1) system(0.91) depend (0.70)
cyberbulli	0.7	cosine	K=6	Dictionary(1) emoticon(1) preprocess(1) system(0.90)
semant	0.7	cosine	K=10	chemical’(0.81) biocur(0.81) gene (0.81)literature(0.81) pubm(0.81)biomed(0.81)

				user(0.80) identifi (0.78) latent(0.77
semant	0.7	cosine	K=9	chemical'(0.81) biocur(0.81) gene (0.81)literature(0.81) pubm(0.81) biomed(0.81) user(0.80) identifi (0.78) latent(0.77)
semant	0.7	cosine	K=8	chemical'(0.82) biocur(0.81) gene (0.81)literature(0.81) pubm(0.81) biomed(0.81) user(0.80) identifi (0.80) latent(0.77)
semant	0.7	cosine	K=7	chemical'(0.81) biocur(0.82) gene (0.82)literature(0.82) pubm(0.82) biomed(0.82) user(0.82) identifi (0.82) latent(0.77)
semant	0.7	cosine	K=6	Biolog(0.90) identify(0.88) user (0.82)chemical'(0.82) biomed(0.82) gene(0.82) literature(0.82) pubm (0.82) biocur(0.82) latent(0.78) system (0.71)
taxonomi	0.7	cosine	K=10	Comprehens(1) framework(1) publish(1) research(0.96)
taxonomi	0.7	cosine	K=9	Comprehens(1) framework(1) publish(1) research(0.96)
taxonomi	0.7	cosine	K=8	Comprehens(1.0) framework(1.0) publish(1.0) research(0.97) textprocess(0.73)
taxonomi	0.7	cosine	K=7	framework (1) publish(1) comprehens (1) research(0.97) textprocess(0.85)
taxonomi	0.7	cosine	K=6	Comprehens(1) framework(1) publish (1) research(0.97) textprocess(0.95)
topicview	0.7	cosine	K=10	relationship (0.98)topic(0.96)cluster (0.94) corpora (0.94)contribut(0.94)factor (0.94) multipl(0.94)new(0.94) model(0.94) lda(0.93) dirichlet (0.89)weight (0.84)
topicview	0.7	cosine	K=9	relationship (0.98)topic(0.96)cluster (0.94)corpora (0.94)contribut(0.94)factor (0.94)multipl(0.94)new(0.94)model(0.94) lda(0.93)dirichlet (0.89)weight (0.84)document(0.81) term (0.73)
topicview	0.7	cosine	K=8	relationship (0.98)topic(0.96) cluster (0.94) corpora (0.94)contribut(0.94) factor (0.94) multipl(0.94)new(0.94)model(0.94) lda(0.93) dirichlet (0.89)weight (0.84) document(0.81) term (0.73)
topicview	0.7	cosine	K=7	Relationship(0.99)topic(0.96) cluster(0.94) corpora(0.94) contribut(0.94)factor (0.94) multipl(0.94) new (0.94)model (0.94)lda(0.93) dirichlet(089) weight(0.85) term(0.83)document(0.81)
topicview	0.7	cosine	K=6	relationship (0.99)topic(0.96)cluster(0.95)contribut (0.95)corpora(0.95)factor(0.95)multipl(0.95 new(0.95) model (0.95)lda (0.94)dirichlet(0.90) weight(0.89)term(0.84)document (0.81)

Table 3. Correlation analysis of Sample Terms in Simple Document Term Matrix with term frequency

Term	Correlation Method	Correlation Threshold	Term Returned with similarity score
Biolog	Cosine	0.7	identifi (0.92) dna(0.88) evolutionarily(0.88) structur(0.88)technique(0.76)
Cyberbulli	Cosine	0.7	Dictionary(1.0) emoticon(1.0) preprocess(1.0) system(0.9)
Semant	Cosine	0.7	retriev (0.74)
Taxonomi	Cosine	0.7	Comprehens(1.0) framework(1.0) publish(1.0) research(0.96)
Topicview	Cosine	0.7	Relationship(0.99)topic(0.96) cluster(0.95) contribut(0.95)corpora(0.95)factor(0.95)multipl (0.95) new(0.95) lda(0.93)model (0.93)dirichlet(0.87) weight(0.82) document (0.79)

IV Results and Analysis

In this experiment Latent Semantic Analysis is applied on term document matrix M ,with different values of k :1≤k≤10.In this study we found that huge no. of sparse entry in the semantic space as well as matrix constructed from it. So as shown in Table 2, we take a optimal value of k =6 having very limited sparse entries in semantic space. For finding semantic relation in the terms, we used "associate" function of lsa package in R and find very optimal and efficient results. For e.g. "biology" is a term in dataset in the first document ,and after preprocessing we get "biolog" as in term document matrix. We find correlated term in semantic spaces for terms like biolog, cyberbulli, semant, taxonomi, topicview etc with different values of k in semantic space. as shown in table 2,3

We have taken k=6,7,8,9,10 and find correlated terms with with a threshold value of correlation 0.7.It is surprising that we get the same result with reduced dimension with k=10,9,8.and with k=7,and k=6,.Hence truncated SVD provide a latent semantic space that is a closer approximation of actual term document matrix.

Again in Table 3,we find the correlated terms in simple term document matrix with term frequency weighting, and as clearly interpreted from the result that the result of truncated space in table 2 are more closer to each other. In table 3 only those terms are extracted that are present within the corpus ,but in table 2,where we have semantic space, correlated terms are retrieved although it is not in the concerned docs(no keyword matching) but semantically related to query term in semantic space. So latent semantic algorithm with optimal no. of dimension reduction using singular value decomposition provides us query results that are semantically correlated in a corpus.

V Conclusion

It is shown that even though, the latent semantic analysis dimension reduction establishes correlation between terms, the

latent semantic analysis is causing a de-gradation in the correlation of a term to itself. Latent semantic analysis provides a foundation for computational theory of meaning. At a minimum, the work presented here demonstrates that it is reasonable to preserve in latent semantic analysis research.

REFERENCES

[1] Fridolin Wild, "LSA Package Latent semantic analysis"-2015

[2] McNamara DS," Computatioinal methods to extract meaning from text and advance theories of human cognition. Topics Cognitive" Sci 3:3–17(2011)

[3] Christopher D. Manning,Prabhakar Raghavan,Hinrich Schütze, "An Introduction to Information Retrieval" Cambridge University Press Cambridge, England (2009)

[4] Haley DT, Thomas P, Roeck AD, Petre M.,"Seeing the whole picture: evaluating automated assessment systems". ITALICS 6:1473–1507(2007)

[5] Martin, D.I., & Berry, M.W, "Mathematical foundations behind Latent Semantic Analysi". In T.K. Landauer, D.S. McNamara, S. Dennis, & W. Kintsch (Eds.), Handbook of Latent Semantic Analysis (pp. 35–56). Mahwah, NJ: Erlbaum. . (2007)

[6] Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R, "Indexing by Latent Semantic Analysis" Journal of the American Society for Information Science, 41:391–407 1990.

[7] Letsche, T., & Berry, M.W.,"Large-scale information retrieval with latent semantic indexing". Information Sciences, 100, 105–137. 1997.