The quality of LSA space directly determines the performance of LSA applications. Factors that could affect LSA space quality include the kind and size of corpus, the dimensions, and the term-weighting measures.

Fixing an optimal dimensionality to be retained in LSA is an empirical issue. Retaining larger dimensions reconstructs closer approximations to the original matrix but may span many unessential relationships. On the other hand, retaining smaller dimensions saves much of computation but with a compromise on the essential relationships. Typically, the number of dimensions retained should be large enough to capture the semantic structure in the text, and small enough to omit trivial correlations. The proper way to make such choices is an open issue in the factor analytic literature [1].

The semantic space obtained after dimensionality reduction through LSA can be used for document classification. In this context, LSA is viewed from a geometrical perspective where words and documents are considered as points in space [1]. The combination of SVD and dimensionality reduction establishes a k-dimensional orthogonal semantic space where the words and documents are distributed according to their common usage patterns. The semantic space reflects those words that have been used in the document to give information about the concepts (the axes) to which the words are closer. Essentially, LSA is a proximity model that spatially groups similar points together. As the dimensional space is reduced, related points draw closer to one another. The relative distances between these points in the reduced vector space show the semantic similarity between documents and is used as a basis for document classification. A test document (a set of words) is mapped as a pseudo-document into the semantic space by the process of "Folding-in" [3]. To fold-in an $m \times 1$ test document vector d into the LSA space of lower dimensions k, a pseudo-document representation ds based on the span of the existing term vectors (the rows of U_{mk}) is calculated as:

$$ds = d^{T} U_{mk} S^{-1} \quad (3)$$

Then the pseudo-document's closeness with all other documents is measured using any of the standard measures of similarity like Cosine measure, Euclidean distance, etc. The category of the document that is located in its nearest proximity in space is the category of the test document. One of the standard approaches for document classification like k-Nearest-Neighbor (kNN), Decision Trees, Naive Bayes, Support Vector Machines (SVM), etc. is applied for classification purposes.

In contrast to many other methods of text classification, LSA is categorizes semantically related texts as similar even when they do not share a single term. This is because in the reduced semantic space, the closeness of documents is determined by the overall patterns of term usage. So documents are classified as similar regardless of the precise terms that are used to describe them. As a result, terms that did not actually appear in a document may still end up close to it if that is consistent with the major patterns of association in the data.

3 Supplemented Latent Semantic Analysis

Being purely mathematical, LSA performs quite well even without relying on any external sources of semantics like word definitions, parts-of-speech or grammar rules, etc. However, when additional information is added into the process, LSA's capability to understand document semantics improves. Extra information is added to LSA by adding new words or documents to the initial term-by-document matrix. So extra rows or columns get added for the information that is intended to be given as supplements to the process. From the geometrical perspective, the newly added supplementary information are new points in the initial space represented by VSM. With the addition of new words, the correlations that existed between words earlier may now change with respect to these newly added words. Words that might have not had any correlation with other words may now start getting correlated with them via the newly added words. LSA's ability to capture correlations in this space improves.

There are several extensions of LSA that were empirically shown to perform better for a variety of tasks. Many of these were specifically extended for classification problems. Relevant prior work is that of Wiemer-Hastings [20] in which surface parsing was employed in LSA by replacing pronouns in the text with their antecedents. The model was evaluated as a cognitive model. Zelikovitz [21] used LSA for document classification by accommodating background knowledge for constructing the semantic space. The work reported increased accuracy rates in classification. Serafin [22] suggested that an LSA semantic space can be built from the co-occurrence of arbitrary textual features which can be used for dialogue act classification. Kanejiya [23] attempted to capture the syntactic context in a shallow manner by enhancing words in LSA with the parts-of-speech of their imme-

diately preceding words to use it an intelligent tutoring system. The results reported an increased ability to evaluate more student answers. Rishel [24] achieved a significant improvement in classification accuracy of LSA by using part-of-speech tags to augment the term-by-document matrix and then applying SVD. The results of the work showed that the addition of parts-of-speech tags decrease word ambiguities.

In the present work, extra information is supplemented to LSA in two forms – document category and domain information. The model supplemented with these two forms of supplements is referred as Supplemented Latent Semantic Analysis (SLSA) throughout this paper. The category of a document conveys some information about semantics to a human being. So including it as supplement to LSA provides some amount of benefit to the overall process. The human knowledge about the category of documents may allow LSA to develop a better semantic representation of words and documents. When using LSA for document classification, the labels of categories of the training documents which human already knows are added as supplements (rows) to the initial term-by-document matrix of LSA. For each added label (row), the cells are set to either 1 for the documents corresponding to the label or 0 for the rest. LSA may use this information to form paths of higher-order correlations between words and derive a better semantic structure.

Domain information is provided as supplements to LSA by including extra documents and in turn extra words other than the existing training set but contextually similar to the existing training set. So extra rows and columns get added to the initial term-by-document matrix. Specifically when the training set is small, the documents in it may not be sufficient to include more number of words that are important to cover the concepts within a domain. The extra documents that are added to the training documents may contain some extra words related to the concepts within the domain but never used in the training set. Such words may provide significant patterns of word combinations by forming paths of higher order correlations between words in the given domain.

4 Using Summaries in SLSA

The summary of a document gives a brief information about the document. Just by reading the summary one understands the central idea of the document. Summaries are either extractive or abstractive. Extractive summaries are generated by picking the important sentences of the text and placing them in the order in which they occur in the text. Abstractive summaries are generated by writing new sentences that capture the main concepts in the text. Most automatic text summarization systems generate extractive summaries as it is difficult to generate abstractive summaries. There exists prior work related to the use of summaries in categorization which are of interest in view of the present work. Ker [25] combined word-based frequency and position method to get categorization knowledge from only the title field for text categorization. Ko [26] considered features of important sentences for improving text categorization. Mihalcea [27] used essence of texts to improve document classification. Hulth [28] reported an improvement in text categorization when the full-text representation is combined with the automatically extracted keywords. Recently document classification was performed based on the latent topics of important sentences within documents [29]. Not much of work is done yet using summaries in LSA to obtain semantic structure of documents with better conceptual correlations.

Summaries of documents are observed to contain only those sentences that highlight the main insights in a document and thus they contain words that actually contribute towards the concepts of the document. Intuitively if summaries are used in LSA, they improve the quality of the semantic structure of documents by removing those sentences which in turn removes those words that cannot actually contribute to build meaningful correlations. The present work is to consider extractive summaries in various proportions instead of the entire full-length documents in SLSA along with two forms of supplements – document category and domain information. The resulting semantic space is assessed by using it in a document classification application. The initial term-by-document matrix for SLSA is constructed by taking the weights of words appearing only in the summaries of documents and not their entire full-lengths. This reduces the initial term-by-document matrix to contain only those important words that contribute solely to the concepts of that category. The semantic structure that is reconstructed is based upon only those word co-occurrence patterns that contribute better towards the document category. This high quality semantic space when used for document classification would increase the classification performance potentially.

For generating summaries of documents in the present work, the LSA-based extractive summary generation method adopted by Krishnamurthi [30] is used. In this method, the matrix V^T resulting after performing LSA is used to select sentences that become part of the extractive summary. The columns of matrix V^T represent the sentences and the rows represent the concepts. The most important concept in

the text is placed in the first row and the row order indicates the importance of concepts. The cells of this matrix gives information about how much a sentence contributes towards a concept. A higher cell value means the sentence contributes more to the concept. For sentence selection, the first concept is chosen and the sentence that contributes the most to this concept is selected as a part of the extractive summary. Then the second concept is chosen and in the same way the sentence with the highest contribution to this concept is selected and added to the summary. This repetition of choosing a concept and then the sentence that contributes the most to that concept is continued until a predefined number of sentences are extracted as a part of the summary.

5 Dataset

For the present work, the large amount of data available on the Internet is explored. The dataset is harvested from a Hindi language news website. Many online news providers like BBC Hindi, Dainik Bhaskar, NDTV Khabar, etc., provide Hindi news articles from a broad range of categories such as science, business, politics, sports, entertainment, education, etc. There are many advantages of choosing news articles to create an in-house Indian language dataset. Firstly, they are available in abundance and are freely accessible. Secondly, news articles are essayed by journalists with the aim of highlighting important insights of the news story. Such articles have a lot of scope to contain natural co-occurrences of words. These natural co-occurrences provide scope for modeling word correlations. Thirdly, the rich linguistic information naturally embodied in the Hindi language text allows to gather syntactic and lexical knowledge necessary for extracting words and documents that are close to the concepts grasped by humans.

The chosen dataset contains 900 news articles downloaded randomly from the "science", "sports" and "entertainment" categories of the BBC Hindi news website [31] with 300 articles in each category. Each document was associated with a category label based on the categorization of the articles on the BBC website. The documents were further validated for its category against its content by a human expert. From each category 50 documents were randomly selected to be used as supplements to provide domain information about that category. 50 articles from each category were randomly selected for performance testing of SLSA and the remaining documents of each category were used for training. Table 1 presents the statistics of the BBC Hindi news dataset.

Table 1: Statistics of the BBC Hindi news dataset

Document attributes	**Values**
Number of documents in the dataset	900
Number of categories	3
Number of documents per category	200
Number of documents in training set	600
Number of documents in test set	150
Number of documents used for providing domain information	150

The in-house dataset may be suspected to be noisy in nature. However it can be argued that this dataset subject to proper preprocessing can be used as a testbed for LSA. During preprocessing of documents in the dataset, initially the corpus was divided into individual documents. Then each document was broken down to a list of words. Then the punctuations, special characters and numbers were removed. Subsequently, the stop-words that were used across all the documents just as language constructs were removed as they cannot actually infer any meaning. This elimination was based on the stopword list provided by the University of Neuchatel [32]. After this, the duplicate occurrences from the remaining word set were removed leaving only unique words. These words were further stemmed to their root forms because it is the root words of a language that infer meaning of a document. For stemming, the work of Ramanathan [33] was used, in which suffixes are stripped off on a longest match basis. After all the preprocessing, the dataset contained only unique root words spread across multiple documents.

6 Empirical Evaluation of SLSA with Summaries

In the experiments that are carried out, plain LSA for full-length documents is the baseline of comparison. Extractive summaries of various proportions of the documents are used in both training and testing phases of SLSA. The semantic space that is derived upon dimensionality reduction is used for classification of Hindi texts. One of the kNN type classifiers i.e. 1-Nearest-Neighbor (1NN) classifier is used for its intuitiveness. This classifier assigns a point (document) in space to the class of its closest neighbor in the semantic space. For measuring closeness, Cosine similarity is used in the empirical evaluations. The accuracies of classification using plain LSA for full-length documents (baseline) and SLSA for extractive summaries of various proportions are calculated for each of the two supplements – document category and domain information. The performance of SLSA is com-

pared against the baseline under various dimensions of the semantic space.

With 600 full-length documents and 10780 words in the training set, the initial term-by-document matrix is of order 10780×600. This matrix is used by plain LSA. Summaries for SLSA are generated by retaining 20%, 30%, 40%, 50% and 60% of the full-length documents resulting in initial term-by-document matrices of order 6588×600, 7661×600, 8376×600, 8870×600 and 9366×600 respectively. Summaries of approximately equal sizes to those in the training sets are generated from 150 documents for testing and 150 documents for providing domain information. Experiments are conducted on each of these sets. Plain LSA with full-length documents is labeled as LSA and SLSA with summaries of various proportions are labeled as SLSA-20, SLSA-30, SLSA-40, SLSA-50 and SLSA-60 in the figures of the following sub-sections.

6.1 Summaries in SLSA with Document Category

The category labels of the training documents that correspond to the categorization of documents on the BBC Hindi news website namely "science", "sports" and "entertainment" are added as supplements (rows) thereby adding 3 rows to the initial matrices. For each added label (row), the cells are set to either 1 for the documents corresponding to the label or 0 for the rest. The average accuracy of classifying 150 test documents with full-length documents in plain LSA is 87.6%. The classification accuracies across various dimensions using summaries in various proportions including the document category labels are plotted against the baseline in Fig. 1 to 5. SLSA with summaries performs better than the baseline across majority of dimensions of the semantic space. Table 2 gives the average classification accuracies obtained in the experiments. It is observed that there is an overall increase in performance by 0.8% to 4.2% by using summaries in SLSA.

Table 2: Average Classification accuracies with summaries in SLSA

Model	Accuracy(%)	Improvement(%)
LSA (Baseline)	87.6	-
SLSA-20	88.4	0.8
SLSA-30	89.1	1.5
SLSA-40	91.2	3.6
SLSA-50	91.8	4.2
SLSA-60	91.6	4.0

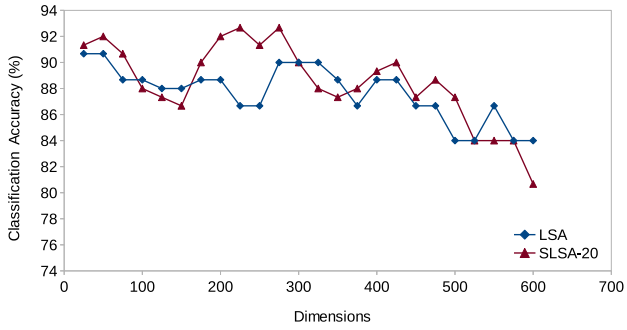

Figure 1: Classification accuracies using SLSA with 20% document summaries

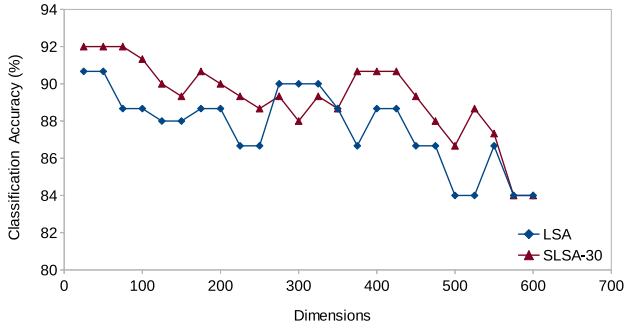

Figure 2: Classification accuracies using SLSA with 30% document summaries

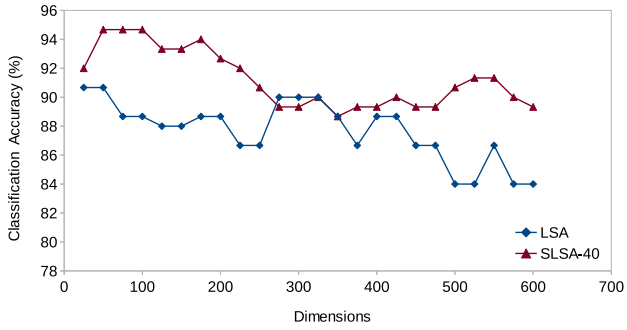

Figure 3: Classification accuracies using SLSA with 40% document summaries

6.2 Summaries in SLSA with Domain Information

For including domain information to the 600 document summaries in training set, the summaries of extra 150 documents – 50 from each of the categories science, sports and entertainment are included into the initial term-by-document matrix. This results in increasing the order of the initial matrices to 7446×750

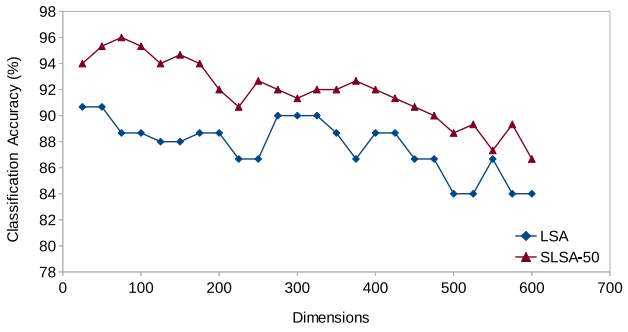

Figure 4: Classification accuracies using SLSA with 50% document summaries

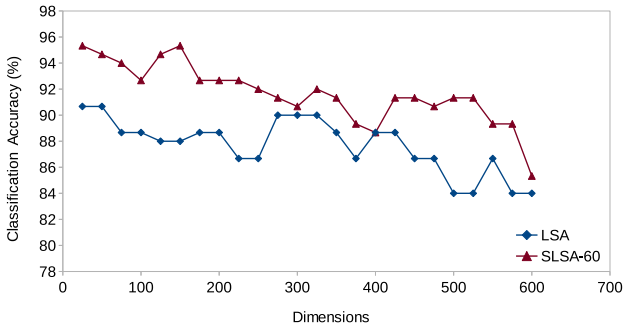

Figure 5: Classification accuracies using SLSA with 60% document summaries

for summaries of 20%, 8648×750 for summaries of 30%, 9463×750 for summaries of 40%, 10041×750 for summaries of 50% and 10564×750 for summaries of 60% of the full-length documents. For classification, the summaries of 150 test documents are folded into the SLSA semantic space reconstructed along with the added domain information and then compared with the initial 600 training documents. The average accuracy of classifying the full-length test documents using plain LSA is 87.6%. The comparative results of SLSA with summaries against the baseline are shown in Fig. 6 to 10. SLSA with summaries is found to perform better than the baseline across majority of dimensions of the semantic space. Table 3 gives the average classification accuracies obtained in the experiments. It is observed that there is an overall increase in performance by 2.1% to 4.8% by using summaries in SLSA.

So far very little work is done for text classification with respect to Indian languages due to non-availability of resources like standard corpus and tools. Text classification tasks for a few Indian languages like Bengali, Punjabi, Assamese and Marathi

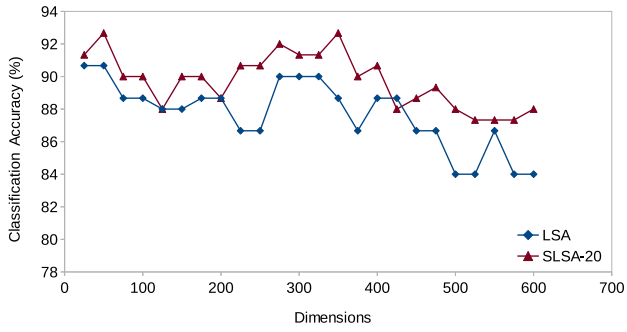

Figure 6: Classification accuracies using SLSA with 20% document summaries

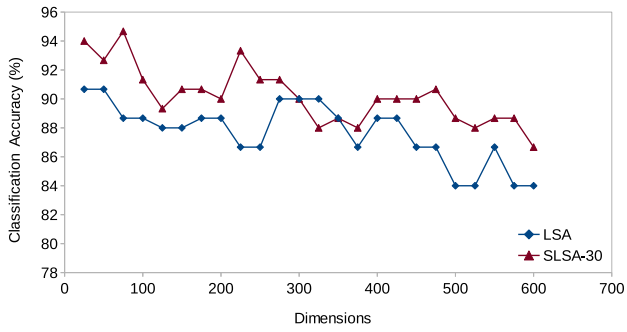

Figure 7: Classification accuracies using SLSA with 30% document summaries

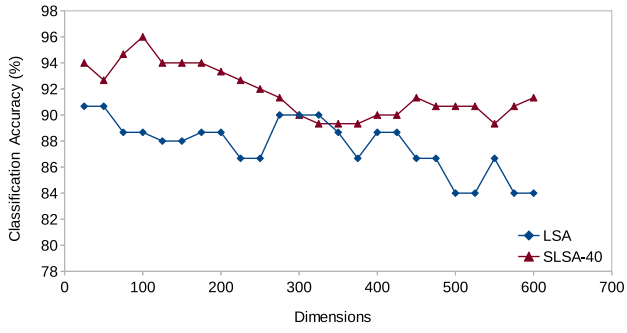

Figure 8: Classification accuracies using SLSA with 40% document summaries

are found in the literature. For text classification on Bengali documents, an n-gram based algorithm was used by Mansur [34] resulting in 90% classification accuracy. Nidhi [35] classified Punjabi text documents using ontology based classifier. The work gave a classification accuracy of 85%. Sarmah [36] presented an approach for classification of Assamese documents using Assamese WordNet. This approach

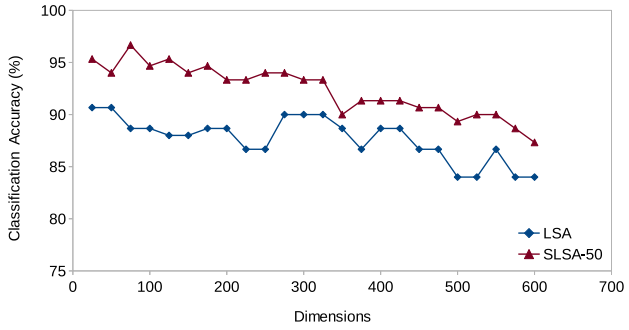

Figure 9: Classification accuracies using SLSA with 50% document summaries

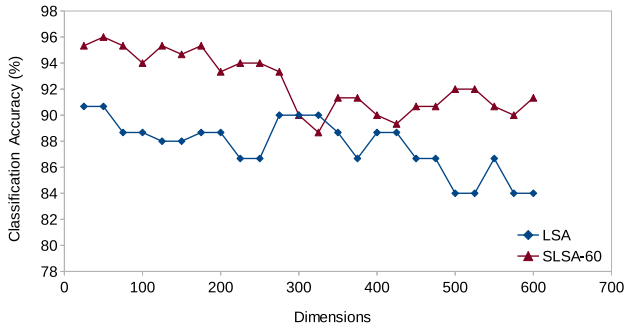

Figure 10: Classification accuracies using SLSA with 60% document summaries

Table 3: Average classification accuracies with summaries in SLSA

Model	Accuracy(%)	Improvement(%)
LSA (Baseline)	87.6	-
SLSA-20	89.7	2.1
SLSA-30	90.2	2.6
SLSA-40	91.7	4.1
SLSA-50	92.3	4.7
SLSA-60	92.4	4.8

gave an accuracy of 90.27% on Assamese documents. The work of Vispute [37] showed that the performance of a VSM based clustering algorithm is good for categorizing Marathi text documents. For Marathi documents the overall accuracy of the system was 91.10%. The present work on Hindi documents performs better than the previous techniques in the field with an accuracy of 92.4%. To the best of the authors' knowledge, this work is the first of its kind in Hindi to use LSA for classification.

7 Conclusions and Future Scope

Summaries of documents contain words that actually contribute towards the concepts of the document. In the present work, summaries are used as inputs to LSA instead of entire full-length documents to capture the semantic structure of documents. Further, the model is supplemented with extra information in two forms – document category and domain information. Supplements are added to LSA by adding extra rows and/or columns to the initial term-by-document matrix from where LSA's processing starts. This enhancement is referred as Supplemented Latent Semantic Analysis (SLSA) in the present work. This input matrix to SLSA results in a high quality semantic structure of document summaries which is used for classifying Hindi texts. The classification performances of SLSA on summaries of various proportions of the full-length documents have been compared with those of plain LSA on full-length documents for the two forms of supplements across various reduced dimensions of the semantic structure. Considerable improvements in performance is achieved using extractive summaries in SLSA rather than entire full-length documents in plain LSA.

The average classification accuracy of LSA using full-length documents is 87.6%. With document category as a supplement in SLSA, the classification experiments using summaries of 20, 30, 40, 50 and 60 percentages resulted in average classification accuracies of 88.4%, 89.1%, 91.2%, 91.8% and 91.6% respectively. With domain information as a supplement in SLSA, summaries of 20%, 30%, 40%, 50% and 60% resulted in average classification accuracies of 89.7%, 90.2%, 91.7%, 92.3% and 92.4% respectively. On the whole, it is observed that for various percentages of summaries of both training and test documents as inputs to SLSA, there is an overall improvement in the classification accuracies by 0.8% to 4.8%. By achieving better classification accuracies using extractive summaries rather than full-length documents, it is concluded that using summaries to understand documents indeed help in capturing better conceptual correlations within texts.

The present work is carried out using term frequency as the term weighting measure in the vector space model. As an extension to this work, experimental evaluations are to be carried out to study the influence on the document structure by considering various unsupervised and supervised term weighting measures in SLSA along with summaries across different supplements in the process.

References:

[1] Deerwester, S., Dumais, S., Furnas G. and Landauer T. K. (1990) Indexing by latent semantic analysis. *American Society for Information Science*, 391–407.

[2] Landauer, T. K. and Foltz, P. W. (1998) An Introduction to Latent Semantic Analysis. *Discourse Processes*, 259–284.

[3] Berry, M. and Dumais, S. (1995) Using linear algebra for intelligent information retrieval. *SIAM Review*, 573–595.

[4] Soboroff, I., Nicholas, C. K., Kukla, J. M. and Ebert, D. S. (1997) Visualizing Document Authorship Using n-grams and Latent Semantic Indexing. *Workshop on New Paradigms in Information Visualization and Manipulation*, 43–48.

[5] Foltz, P. W., Kintsch, W. and Landauer, T. K. (1998) The measurement of textual coherence with Latent Semantic Analysis. *Discourse Processes*, 285–308.

[6] Gordon, M. D. and Dumais, S. (1998) Using latent semantic indexing for literature based discovery. *Journal of the American Society for Information Science*, 674–685.

[7] Wolfe, M. B. and Schreiner, M. E. (1998) Learning from text: Matching readers and text by Latent Semantic Analysis. *Discourse Processes*, 309–336.

[8] Choi, F. Y. Y., Hastings, P. W. and Moore, J. D. (2001) Latent Semantic Analysis for Text Segmentation. *Proceedings of Empirical Methods in Natural Language Processing*, 109–117.

[9] Bradford, R. (2006) Relationship Discovery in Large Text Collections Using Latent Semantic Indexing. *Workshop on Link Analysis, Counter Terrorism and Security*, 20–22.

[10] Gansterer, W. and Janecek, A. (2008) Spam Filtering Based on Latent Semantic Indexing. *Survey of Text Mining: Clustering, Classification, and Retrieval*, 165–183.

[11] Botana, G. J. and Leo, J. A. (2010) Latent Semantic Analysis Parameters for Essay Evaluation using Small-Scale Corpora. *Journal of Quantitative Linguistics*, 1–29.

[12] Khan, N. A. and Yegnanarayana, B. (2004) Latent Semantic Analysis for Speaker Recognition. *International Conference of Spoken Language Processing*, 2589–2592.

[13] Thorleuchter, D. and Van Den Poel, D. (2012) Improved Multilevel Security with Latent Semantic Indexing. *Expert Systems with Applications*, 13462–13471.

[14] Souvannavong, F. and Merialdo, B. (2004) Latent semantic indexing for semantic content detection of video shots. *IEEE Int. Conference on Multimedia and Expo*, 1783–1786.

[15] Yang, J., Luo, M. and Jiao, Y. (2013) Face Recognition Based on Image Latent Semantic Analysis Model and SVM. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, 101–110.

[16] Pulla, C., Karthik, S. and Jawahar, C. V. (2010) Efficient Semantic Indexing for Image Retrieval. *International Conference on Pattern Recognition*, 3276–3279.

[17] Hofmann, T. (1999) Probabilistic latent semantic indexing. *International ACM SIGIR conference on research and development in information retrieval*, 50–57.

[18] Blei, D. M., Ng, A. Y. and Jordan, M. I. (2003) Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 993–1022.

[19] Baker, K. (2005) *Singular Value Decomposition Tutorial*. Electronic document.

[20] Wiemer-Hastings, P. and Zipitria, I. (2001) Rules for Syntax, Vectors for Semantics. *Proceedings of the Annual Conference of the Cognitive Science Society*, 1112–1117.

[21] Zelikovitz, S. (2001) Using LSI for Text Classification in the Presence of Background Text. *ACM International Conference on Information and Knowledge Management*, 113–118.

[22] Serafin R., Eugenio B. D. and Glass M. (2003) Latent semantic analysis for dialogue act classification. *North American Chapter of the Association for Computational Linguistics on Human Language Technology*, 94–96.

[23] Kanejiya D., Kumar A. and Prasad S. (2003) Automatic Evaluation of Students' Answers using Syntactically Enhanced LSA. *Workshop on Building Educational Applications using Natural Language Processing*, 53–60.

[24] Rishel, T., Perkins, A. L. and Yenduri S. (2006) Augmentation of a Term-Document Matrix with Part-of-Speech Tags to Improve Accuracy of Latent Semantic Analysis. *Int. Conference on Applied Computer Science*, 573–578.

[25] Ker, S. J. and Chen, J. (2000) A text categorization based on summarization technique. *ACL workshop on Recent advances in natural language processing and information retrieval*, 79–83.

[26] Ko, Y., Park, J. and Seo, J. (2004) Improving Text Categorization Using the Importance of Sentences. *Information Processing and Management*, 65–79.

[27] Mihalcea, R. and Hassan, S. (2005) Using the essence of texts to improve document classification. *Proceedings of the Conference on Recent Advances in Natural Language Processing*, 150–160.

[28] Hulth, A. and Megyesi, B. B. (2006) A study on automatically extracted keywords in text categorization. *Int. Conference on Computational Linguistics and Annual Meeting of the Association for Computational Linguistics*, 537–544.

[29] Ogura, Y. and Kobayashi, I. (2013) Text Classification based on the Latent Topics of Important Sentences extracted by the PageRank Algorithm. *ACL Student Research Workshop*, 46–51.

[30] Krishnamurthi, K., Panuganti, V. R. and Bulusu, V. V. (2013) An Empirical Evaluation of Dimensionality Reduction using Latent Semantic Analysis on Hindi Text. *IEEE Int. Conference on Asian Language Processing*, 21–24.

[31] http://www.bbc.co.uk/hindi/

[32] http://members.unine.ch/jacques.savoy/clef/hindiST.txt

[33] Ramanathan, A. (2003) A Lightweight Stemmer for Hindi. *Workshop of Computational Linguistics for South Asian Languages Expanding Synergies with Europe*, 42–48.

[34] Mansur, M., Uzzaman, N. and Khan, M. (2006) Analysis of N-Gram Based Text Categorization for Bangla in a Newspaper Corpus. *Center for Research on Bangla Language Processing, BRAC University, Dhaka, Bangladesh*.

[35] Nidhi and Gupta, V. (2012) Domain Based Classification of Punjabi Text Documents using Ontology and Hybrid Based Approach. *Workshop on South and Southeast Asian Natural Language Processing*, 109–122.

[36] Sarmah, J., Saharia, N. and Sarma, S.K. (2012) A Novel Approach for Document Classification using Assamese WordNet. *International Global Wordnet Conference*, 324–329.

[37] Vispute, S.R. and Potey, M.A. (2013) Automatic text categorization of Marathi documents using clustering technique. *International Conference on Advanced Computing Technologies*, 1–5