PUNJABI TEXT CLUSTERING BY SENTENCE STRUCTURE ANALYSIS

Saurabh Sharma and Vishal Gupta

[1]Department of Computer Science & Engineering, University Institute of Engineering and Technology, Panjab University, Chandigarh, India
saurabhsharma381@gmail.com
vishal@pu.ac.in

ABSTRACT

Punjabi Text Document Clustering is done by analyzing the sentence structure of similar documents sharing same topics and grouping them into clusters. The prevalent algorithms in this field utilize the vector space model which treats the documents as a bag of words. The meaning in natural language inherently depends on the word sequences which are overlooked and ignored while clustering. The current paper deals with a new Punjabi text clustering algorithm named Clustering by Sentence Structure Analysis(CSSA) which has been carried out on 221 Punjabi news articles available on news sites. The phrases are extracted for processing by a meticulous analysis of the structure of a sentence by applying the basic grammatical rules of Karaka. Sequences formed from phrases, are used to find the topic and for finding similarities among all documents which results in the formation of meaningful clusters.

KEYWORDS

Punjabi language, Text clustering, Sentence structure analysis, Karaka theory.

1. INTRODUCTION

The documentation in industry these days is prevalent in electronic form in addition to the normal documentation that exists on paper, which helps to provide a quick access to the documents. The documents are stored in a usually very large text database. Exploration and utilization of a vast text database like world wide web in the areas of information retrieval and text mining requires an effective solution [1].

Web Search Engines are immensely popular tools for search and retrieval from World Wide Web with Precision of retrieval. To achieve this precision many methods have been proposed. One of the techniques propagates, clustering the retrieval results before they are displayed to the user. The retrieval results usually cover a variety of topics and the user may be interested in just one of them[1][2].

Text document clustering is a clustering technique which is specifically used for clustering of material in text document format. The text documents are grouped together as clusters on the basis of their similarities and into different groups on the basis of dissimilarities between them, this concept forms the foundation of text document clustering[3].

Sundarapandian et al. (Eds): CoNeCo,WiMo, NLP, CRYPSIS, ICAIT, ICDIP, ITCSE, CS & IT 07,
pp. 237–244, 2012. © CS & IT-CSCP 2012 DOI : 10.5121/csit.2012.2420

2. RELATED WORK

2.1. Text Clustering techniques

Standard Researches done in the field of clustering of data are few and from amongst them, researchers[4] have given a lot of attention to frequent itemsets based text clustering. These researches have been reviewed here.

Xiangwei Liu and Pilian introduced Frequent Term Set based Clustering (Text clustering algorithm), which can lessen the dimension of text data for extremely large databases to enhance the speed and accuracy of the clustering algorithm.

Le Wang et al.[5] gave a Simple Hybrid Algorithm (SHDC). Their algorithm was based on the basis of top-k frequent term sets and k-means so as to overcome the main challenges of current web document clustering. To provide k initial means, Top-k frequent term sets were employed. A Clear description of clustering was given by k frequent term sets and the final optimal clustering was returned by k-means.

Zhitong Su et al [6] introduced a personalized e-learning based on maximal frequent itemsets for web text clustering. Vector Space Model was being used to represent the web documents initially. Finally, on the basis of a new similarity method measure of itemsets, clustering was done using maximal itemsets.

Yongheng Wang et al [7] used a frequent term based parallel clustering algorithm which was used to cluster short documents in a very large data base W.L.Liu and X.S.Zheng [8] propagated the document clustering algorithm on the basis of frequent term sets.

For discovering and unfolding the topics included in a text collection, Henry Anaya – Sanchez et al [9] proposed a clustering algorithm. Generation of most probable term pairs from the collection and estimation of the topic homogeneity related to these pairs was the foundation of the algorithm. Term Pairs, whose support sets were homogeneous for denoting collection topics, were used to generate topics and their descriptions. Experimental results, over three benchmark text collections thus obtained, showed efficacy and usefulness of the approach.

Florian Beil et al [10] employed frequent term sets for text clustering. Algorithm for association rule mining was used to determine such frequent sets. Mutual overlap of frequent sets was calibrated with regards to the sets of supporting documents to make clusters on the basis of frequent term sets. For frequent term based text clustering FTC & HFTC were given by them.
Review of literature in this respect indicates that the clustering algorithms unanimously treat the documents being clustered as a mere bag of words. As this model doesn't consider the information contained in the positioning of words, it isn't an effective approach for clustering text documents[1].

2.2. Similar work done in Indian languages

The prominent work in this category was done by Sudhir K Mishra [11] in 2007 whose work focused on the theory of Karaka, introduced by Panini in his work in Adhikara sutra[12], for analyzing the structure of a sentence in Sanskrit Language. Karaka denotes the relationship between a noun and a verb in a sentence and it literally means 'that which brings about' or the 'doer'[13]. Any factor that contributes to the accomplishment of any action. Punjabi language identifies eight sub types, like in Hindi and Sanskrit langugae[14][15][16].

3. ALGORITHM FOR PUNJABI TEXT CLUSTERING USING SENTENCE STRUCTURE (CSSA)

This algorithm starts with reading N number of text files. For each file repeat the steps from 1 to 7.

Step 1: For each word, If any sentence boundary or Karaka is found, break the sentence into phrase.

Step 2: For each phrase, repeat steps 3 and 4.

Step 3: Using Punjabi Noun Stemmer, check each word against stemming rules. If a word satisfies any rule, remove its suffix and stem word to its root word.

Step 4: Create sequences of words, having length of two, from a phrase. Calculate the TF of each sequence.

Step 5: Sort the Sequences in descending order. Make top 5 sequences as Key Sequences and the sequence with maximum TF, is the topic of the document.

Step 6: Check the Key Sequences for overlapping words to find the sequences of length greater than two words. If overlapping sequences are found, merge these sequences to form longer sequences.

Step 7: Find the similarity among all documents by matching topic and Key Sequences of each document. If any similar sequences found, group them into a cluster. Create cluster label using most common sequence.

3.1. Punjabi sentence boundary identification and Extracting phrases using list of Karaka

A text document is taken as input for extracting the phrases using a list of Karaka containing 234 karaka, created according to Punjabi grammar. One Phrase is counted when a Karaka is found or the boundary of a sentence is found. The generated outcome of this step consists of a number of phrases extracted from a text document. For example,

Input sentence: ਮਾਲਵਾ ਸੱਭਿਆਚਾਰਕ ਅਤੇ ਵੈਲਫੇਅਰ ਰਜਿ: ਬਰਨਾਲਾ ਵੱਲੋਂ ਚਲਾਏ ਜਾ ਰਹੇ ਸਿਲਾਈ ਸੈਂਟਰ ਦੀਆਂ ਦਸ ਵਿਦਿਆਰਥਣਾਂ ਨੂੰ ਸਿਖਲਾਈ ਸਰਟੀਫਿਕੇਟ ਵੰਡੇ ਗਏ

mālvā sabbhiācārak atē vailphēar raji: barnālā vallōṃ calāē jā rahē silāī saiṇṭar dīāṃ das vidiārthaṇāṃ nūṃ sikhlāī sarṭīphikēṭ vaṇḍē gaē

Phrases extracted: ਮਾਲਵਾ ਸੱਭਿਆਚਾਰਕ, ਵੈਲਫੇਅਰ ਸੱਥ ਰਜਿ: ਬਰਨਾਲਾ, ਚਲਾਏ ਜਾ ਰਹੇ ਸਿਲਾਈ ਸੈਂਟਰ, ਦਸ ਵਿਦਿਆਰਥਣਾਂ, ਸਿਖਲਾਈ ਸਰਟੀਫਿਕੇਟ ਵੰਡੇ ਗਏ

gaēmālvā sabbhiācārak, vailphēar satth raji: barnālā, calāē jā rahē silāī saiṇṭar, das vidiārthaṇāṃ, sikhlāī sarṭīphikēṭ vaṇḍē gaē

Karaka found in sentence: ਅਤੇ, ਵੱਲੋਂ, ਦੀਆਂ, ਨੂੰ (atē, vallōṃ, dīāṃ, nūṃ)

List of karaka contains ਦੀ, ਦੇ, ਵੱਲੋਂ, ਨੂੰ, ਦਾ, ਨੇ, ਤੇ, ਨਾਲ, ਵਿੱਚ, ਲਈ, ਇਸ, ਦੀਆਂ, ਦੌਰਾਨ, ਅਤੇ, ਜਦੋਂ, ਵੀ, ਕੋਈ, ਤਾਂ, ਉਹ etc. total 234 karaka.

3.2. Using Punjabi Noun Stemmer

In this step, check each and every word of the phrase against stemming rules of Punjabi noun stemmer. Total 18 rules are identified for stemming of words in Punjabi language. If a word matches the rules of stemming, they are stemmed to their root word [17][18][19]. For example,

ਖੇਡਾਂ -> ਖੇਡ, ਮੁਕਾਬਲਿਆਂ -> ਮੁਕਾਬਲਾ, ਲੜਕੇ -> ਲੜਕਾ

khēḍāṃ -> khēḍ, mukābliāṃ -> mukāblā, laṛkē -> laṛkā

3.3. Assign weight-age to each Sequence

This step involves the dividing the phrase into number of sequences each having length of two words. Calculate the TF (Term Frequency) of each sequence. For example,

Phrase: ਇੱਕ ਵੱਡੀ ਰਾਹਤ ਮਿਲਣ (ikk vaḍḍī rāhat milaṇ)

Sequences: ਇੱਕ ਵੱਡੀ (ikk vaḍḍī), ਵੱਡੀ ਰਾਹਤ (vaḍḍī rāhat), ਰਾਹਤ ਮਿਲਣ(rāhat milaṇ).

3.4. Find the topic of the document and Key Sequences

For In this step, all the sequences are sorted in descending order of their Term Frequency. The sequence with the maximum Term Frequency is treated as the Topic of the document and top 5 sequences are considered as key sequences, which will be used to find similar documents. Check the Key Sequences for overlapping words to find the sequences of length greater than two words. If overlapping sequences are found, merge these sequences to form longer sequences. For example,

Sequence 1: ਸ਼ਿਰੋਮਣੀ ਅਕਾਲੀ, (shirōmṇī akālī),

Sequence 2: ਅਕਾਲੀ ਦਲ, (akālī dal)

After merging overlapping words, we have resultant sequence of length 3 i.e. more than 2.

ਸ਼ਿਰੋਮਣੀ ਅਕਾਲੀ ਦਲ (shirōmṇī akālī dal)

At the end of this step, we have 5 key sequences and some longer sequences, if any, for finding similarities among all other documents. All the above steps will be repeated for each document.

3.5. Find the similarity among all documents using topic and key Sequences of each document

In this step, similar documents are grouped together by matching top 5 key sequences and longer sequences, if any, of each document. Two documents will be grouped together if they contain same sequences. If a document contains same number of matching sequences with more than one document i.e. if a document have more than one matching documents and it is overlapping in more than one cluster, then document will be grouped with that document which has the smallest total of matching sequences indexes.

For example, document d1 have 2 sequences common with documents d2 and d3. d1 and d2 have sequences in common at index {1, 3} and d1 and d3 have sequences in common at index {2, 4}. Now, number of matched sequences is same in both the cases, that is 2. For grouping of document d1, we will find the sum of indexes of sequences. For d1 and d2, score = 1 + 3 = 4, for

d1 and d3, score = 2 + 4 = 6 Document d1 will be grouped with the document which has the smallest score. In this case, document d1 will be grouped with d2.

3.6. Create the Clusters

Do The last step involves making of clusters and assigning them meaningful labels. The most common phrase from grouped documents will be the label of that cluster. For example,

1. ਪਤੰਜਲਿ ਯੋਗਪੀਠ ਵਿੱਚ ਰੌਣਕ ਪਰਤੀ। ਆਚਾਰਿਆ ਬਾਲਕ੍ਰਿਸ਼ਣ ਨੂੰ ਉਤਰਾਖੰਡ ਉੱਚ ਅਦਾਲਤ ਵਲੋਂ ਇੱਕ ਵੱਡੀ ਰਾਹਤ ਮਿਲਣ ਵਲੋਂ ਉਨ੍ਹਾਂ ਦੇ ਪਤੰਜਲਿ ਯੋਗਪੀਠ ਵਿੱਚ ਪਿਛਲੇ ਪੰਜ ਦਿਨਾਂ ਵਲੋਂ ਪਸਰੇ ਸੰਨਾਟੇ ਦੇ ਬਾਅਦ ਹੁਣ ਰੌਣਕ ਪਰਤ ਆਈ ਹੈ। ਸੀਬੀਆਈ ਦੁਆਰਾ ਉਨ੍ਹਾਂ ਦੀ ਸੰਭਾਵਿਕ ਗਿਰਫਤਾਰੀ ਦੀ ਸੰਦੇਹ ਵਲੋਂ ਪਤੰਜਲਿ ਯੋਗਪੀਠ ਵਿੱਚ ਸੱਨਾਟਾ ਛਾ ਗਿਆ ਸੀ।

patñjali yōgpīṭh vicc rauṇak partī. ācāriā bālkrishaṇ nūṃ utrākhaṇḍ ucc adālat valōṃ ikk vaḍḍī rāhat milaṇ valōṃ unhāṃ dē patñjali yōgpīṭh vicc pichlē pañj dināṃ valōṃ pasrē sannāṭē dē bāad huṇ rauṇak parat āī hai. sībīāī duārā unhāṃ dī sambhāvik girphatārī dī sandēh valōṃ patñjali yōgpīṭh vicc sannāṭā chā giā sī.

2: ਪਤੰਜਲਿ ਯੋਗਪੀਠ ਵਿੱਚ ਉਮੜਾ ਸਮਰਥਕਾਂ ਦਾ ਸੈਲਾਬ । ਬਾਬਾ ਰਾਮਦੇਵ ਦੇ ਆਉਣ ਦੀ ਖਬਰ ਸੁਣ ਕਰ ਹਰਦੁਆਰ ਸਥਿਤ ਪਤੰਜਲਿ ਯੋਗਪੀਠ ਵਿੱਚ ਵੱਡੀ ਗਿਣਤੀ ਵਿੱਚ ਉਨ੍ਹਾਂ ਦੇ ਸ਼ਰੱਧਾਲੁਆਂ ਦੇ ਪੁੱਜਣ ਦਾ ਸਿਲਸਿਲਾ ਸ਼ੁਰੂ ਹੋ ਚੁੱਕਿਆ ਹੈ। ਪਤੰਜਲਿ ਯੋਗਪੀਠ ਵਿੱਚ ਮੌਜੂਦ ਸਾਧੁ - ਸੰਤਾਂ ਨੇ ਬਾਬਾ ਰਾਮਦੇਵ ਦੇ ਖਿਲਾਫ ਕੀਤੀ ਗਈ ਕਾੱਰਵਾਈ ਦੀ ਆਲੋਚਨਾ ਕੀਤੀ।

patñjali yōgpīṭh vicc umṛā samrathkāṃ dā sailāb. bābā rāmdēv dē āuṇ dī khabar suṇ kar harduār sathit patñjali yōgpīṭh vicc vaḍḍī giṇtī vicc unhāṃ dē sharddhāluāṃ dē pujjaṇ dā silsilā shurū hō cukkiā hai. patñjali yōgpīṭh vicc maujūd sādhu-santāṃ nē bābā rāmdēv dē khilāph kītī gaī kārrvāī dī ālōcnā kītī.

Table 1. Creation of Clusters using Topic and key sequences

Doc	Top 3 Key Sequences	Topic	Cluster
Doc 1	ਪਤੰਜਲਿ ਯੋਗਪੀਠ (patñjali yōgpīṭh), ਰੌਣਕ ਪਰਤੀ (rauṇak partī), ਆਚਾਰਿਆ ਬਾਲਕ੍ਰਿਸ਼ਣ (ācāriā bālkrishaṇ)	ਪਤੰਜਲਿ ਯੋਗਪੀਠ (patñjali yōgpīṭh)	ਪਤੰਜਲਿ ਯੋਗਪੀਠ patñjali yōgpīṭh
Doc 2	ਪਤੰਜਲਿ ਯੋਗਪੀਠ (patñjali yōgpīṭh), ਉਮੜਾ ਸਮਰਥਕਾਂ (umṛā samrathkāṃ), ਬਾਬਾ ਰਾਮਦੇਵ (bābā rāmdēv)	ਪਤੰਜਲਿ ਯੋਗਪੀਠ (patñjali yōgpīṭh)	

4. RESULT ANALYSIS

To evaluate the accuracy of the clustering results generated by clustering algorithms, F-measure is employed. A commonly used external measurement method; it is a standard evaluation method for both flat and hierarchical clustering structures. Let us assume that each cluster is treated as if it were the result of a query and each natural class is treated as if it were the relevant set of

documents for a query. The recall, precision, and F-measure for natural class K_i and cluster C_j are calculated as follows:

$$\text{Precision}(K_i, C_j) = n_{ij} / |C_j| \quad (1)$$

$$\text{Recall}(K_i, C_j) = n_{ij} / |K_j| \quad (2)$$

$$\text{F-Measure}(K_i, C_j) = 2 * \frac{[\text{Precision}(K_i, C_j) * \text{Recall}(K_i, C_j)]}{[\text{Precision}(K_i, C_j) + \text{Recall}(K_i, C_j)]} \quad (3)$$

where n_{ij} is the number of members of natural class K_i in cluster C_j . Intuitively, $F(K_i;C_j)$ measures the quality of cluster C_j in describing the natural class K_i, by the harmonic mean of Recall and Precision for the "query results" C_j with respect to the "relevant documents" K_i.

In Fig 1, the graph plotted for Precision, Recall and F-Measure for two algorithms that were studied for clustering of Punjabi text documents. The standard Vector Space Model, is implemented for comparison with our proposed algorithm, shows a good precision but a very poor recall value. This leads to a very low value of F-Measure which is indicative of its overall poor performance. On the other hand, proposed CSSA shows good Precision, Recall and F-Measure, and hence generate better clustering results.

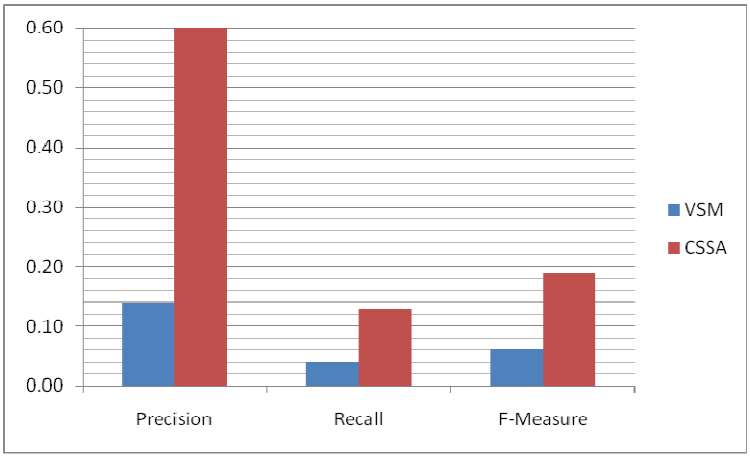

Figure 1. Spam traffic sample

5. CONCLUSIONS AND FUTURE WORK

The results were calculated for the proposed algorithm and comparisons were drawn for the results of the same data set by implementing VSM also. The results indicated that Proposed algorithm is logically feasible, efficient and practical for Punjabi text documents. Experimental results are indicative of the fact that proposed algorithm performs better than standard VSM with Punjabi text data sets. Proposed algorithm has better precision and recall hence , F Measure is better for it than VSM model. However, these results are drawn for the text data in Punjabi Language .

REFERENCES

[1] Yanjun Li, Soon M. Chung, John D. Holt. 2008. "Text document clustering based on frequent word meaning sequences". Data & Knowledge Engineering. 64, 1 (Jan.2008). Elsevier Science Publishers B. V. Amsterdam, The Netherlands, The Netherlands. 381-404. DOI: 10.1016/j.datak.2007.08.001

[2] Allan J., HARD track overview in TREC 2003. "High accuracy retrieval from documents". In *Proceedings of the 12th Text Retrieval Conference*, pp. 24–37, (2003)

[3] Tian Weixin, Zhu Fuxi. 2008. "Text Document Clustering Based On The Modifying Relations". In *Proceedings of IEEE International Conference on Computer Science and Software Engineering*. 1 (12-14 Dec. 2008), 256-259. ISBN: 9780769533360. DOI: 10.1109/CSSE.2008.1545

[4] S.Murali Krishna and S.Durga Bhavani. 2010. "An Efficient Approach for Text Clustering Based on Frequent Itemsets", European Journal of Scientific Research ISSN 1450-216X, 42, 3 (2010), EuroJournals Publishing, Inc. 2010, 399-410.

[5] Le Wang, Li Tian, Yan Jia and Weihong Han. 2007. "A Hybrid Algorithm for Web Document Clustering Based on Frequent Term Sets and k-Means". In *Proceedings APWeb/WAIM 2007 International Workshops: DBMAN 2007, WebETrends 2007, PAIS 2007 and ASWAN 2007*. 4537(Huang Shan, China, June 16-18, 2007). LNCS. Springer Berlin, 198-203, (2007). isbn = 978-3-540-72908-2. doi = http://dx.doi.org/10.1007/978-3-540-72909-9_20

[6] Zhitong Su, Wei Song, Manshan Lin, Jinhong Li. 2008. "Web Text Clustering for Personalized E-learning Based on Maximal Frequent Itemsets". In *Proceeding of the 2008 International Conference on Computer Science and Software Engineering, 06*. IEEE Computer Society Washington, DC, USA. 452-455 (2008) ISBN: 978-0-7695-3336-0. doi>10.1109/CSSE.2008.1639

[7] Yongheng Wang, Yan Jia and Shuqiang Yang. 2006. "Short Documents Clustering in Very Large Text Databases". In *Proceedings of Web Information Systems – WISE 2006 Workshops*. 4256. Lecture Notes in Computer Science, Springer Berlin. 83-93(2006) DOI: 10.1007/11906070_8

[8] W.-L. Liu and X.-S. Zheng. 2005. "Documents Clustering based on Frequent Term Sets". In *Proceeding of Intelligent Systems and Control, (2005).*

[9] Henry Anaya-Sánchez, Aurora Pons-Porrata, Rafael Berlanga-Llavori. 2010. "A document clustering algorithm for discovering and describing topics". Pattern Recognition Letters. 31, 6 (April, 2010) Elsevier Science Inc. New York, NY, USA. 502-510. doi>10.1016/j.patrec.2009.11.013

[10] Florian Beil, Martin Ester and Xiaowei Xu. 2002. "Frequent term-based text clustering". In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining* (Edmonton, Alberta, Canada, 2002). ACM New York, NY, USA. 436 - 442. ISBN:1-58113-567-X doi>10.1145/775047.775110

[11] Mishra, Sudhir Kumar. 2007. "Sanskrit Karaka Analyzer for Machine Translation", M.Phil dissertation submitted to SCSS, Jawaharlal Nehru University.

[12] Bharati, A., Sangal, R. 1990. "A Karaka Based Approach to Parsing of Indian Languages". In *Proceedings of the 13th conference on Computational linguistics.* 3(1990) Association for Computational Linguistics Stroudsburg, PA, USA. 25-29. ISBN:952-90-2028-7 doi>10.3115/991146.991151

[13] Bharati, A., Sangal, R. 1993. "Parsing free word order languages in the Paninian framework", In *Proceedings of the 31st annual meeting on Association for Computational Linguistics (1993)* Association for Computational Linguistics Stroudsburg, PA, USA. 105-111. doi>10.3115/981574.981589

[14] Bharati, A., Sangal, R., Reddy, P. 2002. "A Constraint Based Parser Using Integer Programming". In *Proceedings of 10th IEEE International Conference on Networks* (August 27-30, Grand Copthorne Waterfront, Singapore, 2002) Towards Network Superiority.

[15] Gupta, Vishal, Lehal, G. S. 2011. "Punjabi Language Stemmer for nouns and proper names". In *Proceedings of the 2nd Workshop on South and Southeast Asian Natural Language Processing* (WSSANLP), IJCNLP 2011 (Chiang Mai, Thailand, 2011) 35–39.

[16] Gupta Vishal, Lehal, G. S. 2011. "Preprocessing Phase of Punjabi Language Text Summarization", Information Systems for Indian Languages, Communications in Computer and Information Science, 139 (2011) Springer-Verlag. 250-253.

[17] Md. Zahurul Islam, Md. Nizam Uddin, Mumit Khan. 2007. "A light weight stemmer for Bengali and its Use in spelling Checker". In *Proceedings of 1st International Conference on Digital Communication and Computer Applications (DCCA 2007)*, (Irbid, Jordan, 2007)

[18] Gurmukh Singh, Gill, Mukhtiar Singh, Joshi S.S. 1999. "Punjabi to English Bilingual Dictionary". Punjabi University Patiala.

[19] Gurmukhi-Roman Transliterator GTrans 1.0, http://www.learnpunjabi.org/gtrans/index.asp

Authors

Saurabh Sharma is M.E. in Computer Science & Engineering from University Institute of Engineering & Technology, Panjab University Chandigarh. He has done B.Tech. in Computer Science & Engineering from Swami Devi Dayal Institute of Engineering & Technology, Barwala in 2007. He is devoting his research work in the field of Computational Linguistics and specifically to Text Mining. His research work has been published in reputed International journals.

Vishal Gupta is Assistant Professor in Computer Science & Engineering Department at University Institute of Engineering & Technology, Panjab University Chandigarh. He has done M.Tech. in computer science & engineering from Punjabi University Patiala in 2005. He secured 82% Marks in M.Tech. He did his B.Tech. in CSE fromGovt. Engineering College Ferozepur in 20 03. He is also pursuing his PhD in Computer Sc & Engg. Vishal is devoting his research work in field of Natural Language processing. He has developed a number of research projects in field of NLP including synonyms detection, automatic question answering and text summarization etc. One of his research papers on Punjabi language text processing was awarded as best research paper by Dr. V. Raja Raman at an International Conference at Panipat. He is also a merit holder in 10th and 12th classes of Punjab School education board.