Algorithm for Punjabi Text Classification

Nidhi
M.E, Computer Science & Engineering
University Institute of Engineering & Technology,
Panjab University, Chandigarh, India

Vishal Gupta
Assistant Professor, Computer Science & Engineering
University Institute of Engineering & Technology,
Panjab University, Chandigarh, India

ABSTRACT
Text Mining is a field that extracts hidden, not yet discovered, useful information from the text document according to user's query. And Text Classification is one of the text mining tasks to manage the information efficiently, by classifying the documents into classes using classification algorithms. Any text classification method uses a set of features to characterize each text document, where these features should be relevant to the task at hand. Not much work has been done for Punjabi text classification. Adequate annotated corpora are not yet available in Punjabi. This paper introduces preprocessing techniques, features selection methods for Punjabi and classification algorithm to classify the Punjabi Text documents.

General Terms
Natural language processing, Text classification

Keywords
NLP, Text mining, Text Classification, Features extraction

1. INTRODUCTION
Text Mining refers to the process of deriving high-quality information from text. 'High quality' in text mining means that information extracted should be relevant to the user, and according to the interest of the user. The text document may be a plain text document (e.g. ASCII) or a tagged text document (e.g. HTML/XML). And typical text mining tasks include text classification (text categorization), text clustering, concept/entity extraction, topic tracking, information visualization, question answering, document summarization etc. The field of text mining has received a lot of attention due to the always increasing need for managing the information that resides in the vast amount of available documents [1] [2] [3].

Text classification is one of the important research issues in the field of text mining, where the documents are classified with supervised knowledge. It assigns a text document to one of a set of predefined classes. The dramatic increase in the amount of content available in digital forms gives rise to large-scale digital libraries, targeted at millions of users. As a result, it has become a necessary to categorize large texts (documents) [4]. Mostly-text documents include business letters, forms, newspapers, technical reports, proceedings, and journal papers, etc. Text classification is used for document filtering, detecting a document's encoding (ASCII, Unicode UTF-8 etc), sentiment detection, email sorting, topic specific search, language detection of the text document etc.

Feature extraction is a text mining task that is covered under the concept of Natural Language Processing. Features are the index terms that contain the most important information about the contents of the document. Feature selection is the task to identify a subset of relevant features from a document so as to achieve classification accuracy. [4]

Punjabi text classification process divides into three phases:

1. Preprocessing Phase: This phase include, stop words elimination, stemming, punctuation marks and special symbols removal.
2. Feature Extraction: It consist of statistical approach and linguistic approach to extract relevant features from the documents to perform classification.
3. Processing Phase: The last phase of the Punjabi text classification, apply text classification algorithms to the extracted features to classify the documents into classes.

With the increasing importance of the Web and other text-heavy application areas, the demands for and interest in both Text Mining and Natural Language Processing (NLP) have been rising. Researchers in text mining have hoped that NLP—the attempt to extract a fuller meaning representation from free text—can provide useful improvements to text mining applications of all kinds [5].

In this paper, we concentrate on different features to be selected to classify Punjabi text documents. And also proposes an algorithm for the Punjabi Text documents. Section 2 gives an overview of the various Text Classification Techniques used by different Researchers to classify the text documents. Section 3 provides techniques to extract the features from the Punjabi Text Document to make classification efficient. Section 4 includes Text Classification Algorithm for the Punjabi Text Documents.

2. RELATED WORK
Text Classification is an active research area of text mining, with increasing demand of managing large textual databases. With increase in the availability of the digital textual data, it becomes necessary to manage this textual data efficiently to make the access, search of any particular document faster [3] [6].

Earlier many systems support Standing Query, which is like any other query except that it is periodically executed on a collection to which new documents are added over a time. But to achieve good recall (efficiency and accuracy), standing queries have to be refined over time and can gradually become complex. E.g. if standing query is multicore AND computer AND chips, the results will miss many new relevant articles containing terms such as multicore processor. To overcome this problem, text classification was

proposed, given a set of classes, we seek to which class (es) a given object belongs to. In this example, the two classes will be: documents about multicore computer chips and documents not about multicore computer chips. This is known as two-class classification [7].

Text Classification assigns class to the unlabelled text documents from a set of predefined classes using certain rules. Traditionally, many text classification tasks been solved manually, but such manual classification is expensive to scale and also labor intensive since rules are created manually for the efficient classification. Therefore, another approach to classification is machine learning-based text classification that uses automatic generation of rules to classify the documents [7].

The various techniques to classification are: Nearest Neighbour (KNN) [8] [9] [10], Bayesian classification [7] [10] [11], Support Vector Machine [12] [13], Association based classification [14] [15] [16], Term Graph Model [17] [18], Decision Tree [10], Neural Networks [19] [20] etc.

Text Classification for Indian Languages: With exponential increase in the information in Indian languages on the web, automatic information processing and retrieval become an urgent need. So far very little work has been done for text classification with respect to Indian languages. The problems faced by many Indian Languages are: No capitalization, non-availability of large gazetteer lists, lack of standardization and spelling, scarcity of resources and tools, free word order language. The only corpus available in most languages is an EMILLE/CIIL corpus that contains about 3 million words. These corpus documents are classified manually; hence they are used as Training Set. Indian Languages, especially Dravidian languages (Tamil, Telugu, Kannada, and Malayalam) are highly inflectional and derivational language, leading to a very large number of word forms for each root word. This makes the classification task more difficult [21].

Many Text Classification Techniques used for Southern Indian Languages are, e.g. Naive Bayes classifier has been applied to Telugu news articles in four major classes to about 800 documents. In this, Category-wise normalized $TF \times IDF$ are used to extract the features from the document [22]; Semantic based classification using Sanskrit wordnet used to classify Sanskrit Text Document [23]; for Urdu language, statistical techniques using Naïve Bayes and Support Vector Machine used to classify subjective sentences from objective sentences, in this language specific preprocessing used to extract the features [24]; for Hindi language, Support Vector Machine used for classification [25]; for Tamil Documents, Vector Space Model, Artificial Neural network is used, for extracting features weights are assigned to the terms[26].

But for Punjabi Text Document, not much work has been done to classify the documents due to lack of resources, annotated corpora, name dictionaries, good morphological analyzers, POS taggers are not yet available in the required measure.

3. FEATURE EXTRACTION

When the input data to an algorithm is too large to be processed and it is suspected to be notoriously redundant (much data, but not much information) then the input data will be transformed into a reduced representation set of features. Transforming the input data into the set of features is called feature extraction. If the features extracted are carefully chosen it is expected that the features set will extract the relevant information from the input data in order to perform the desired task using this reduced representation instead of the full size input. Due to the high dimensionality of feature sets, feature extraction can be performed to reduce the dimensionality of the feature space and improve the efficiency and accuracy of the classifiers. [27]

Feature extraction also helps to acquire better understanding about the contents of the documents by telling which the important features are and how they are related with each other.

A good feature set contains discriminating information, which can distinguish one object from other objects. It must also be as robust as possible in order to prevent generating different feature codes for the objects in the same class. The selected set of features should be a small set. [28]

But before extracting features from the documents, we have to do some preprocessing. In preprocessing phase we represent each original text document as "Bag of words". Then following operations are done on each document:

1. Remove punctuation marks, special symbols (<,>, :,{,},[,],^,&,*,(,) etc.) from the Punjabi text documents. Also, if a document contains excess use of spaces, tabs, shift, remove them.
2. Stop words Removal: Stop words are the frequently occurring set of words which do not aggregate relevant information to the text classification task. Therefore, we have to remove these words from the text documents. We have made a list of Punjabi language stop words from the dataset. This corpus contains around 47,029 words [29]. We manually analyzed and identified 2,141 stop words. The total stop words removed from 150 Punjabi text documents is 21,081 in 1 min 29 secs. Some commonly occurring stop words are: ਦੇ (dē), ਦਾ (dā), ਵਿੱਚ (vicc), ਦੀ (dī), ਹੈ (hai), ਇਹ (ih), ਅਤੇ (atē), ਵਲੋਂ (valōṃ),, ਹਨ (han), ਨੂੰ (nūṃ) etc.
3. Stemming: It is the process for reducing inflected (or sometimes derived) words to their stem, base or root form. E.g. ਖਿਡਾਰੀਆਂ (khiḍārīāṃ) → ਖਿਡਾਰੀ (khiḍārī), ਟੀਮਾਂ (ṭīmāṃà)→ ਟੀਮ (ṭīm) etc. An analysis of corpus was made and various possible noun suffixes were identified like ੀਆਂ (īāṃ), ਾਂ (āṃ), ੇ (ē), ੇਂ (ēṃ) etc. Stemming can be done using Regular Expressions Functions, but this can be used if we have small dataset, also it is programming language dependent. Therefore, for large dataset, as in the given corpus we have 150 Punjabi text documents, we use Punjabi Language Stemmer given in [30].

   Input: ਖੇਡਾਂ (khēḍāṃ); Output: ਖੇਡ (khēḍ)

After preprocessing phase, words that do not contain relevant information to the classification tasks are removed. Now, we have documents with less numbers of words, and extracting features from these words now become easier. Now, there are two approaches to extract features from the text document that are following:

3.1 Statistical Approach

These methods are simple, language independent and do not require training data. Following are the different methods [6] [31]:

3.1.1 Term Frequency Weighting (TF)

In this simple method, the weight of a term in a document is equal to the number of times the term appear in the document, i.e. to the raw frequency of the term in the document.

$$w_i = tf_i \text{ Where } i = i^{th} \text{ term in the document}$$

*3.1.2 Term Frequency * Inverse Document Frequency Weighting (TF*IDF)*

TF weighting do not consider the frequency of the term throughout all the documents in the document corpus. Term Frequency * Inverse Document Frequency ($TF \times IDF$) weighting is the most common method used for term weighting that takes into account this property. In this approach, the weight of term i in document d assigned proportionally to the number of times the term appears in the document, and in inverse proportion to the number of documents in the corpus in which the term appears.

$$w_i = tf_i . \log\left(\frac{N}{N_i}\right)$$

$TF \times IDF$ Weighting approach weights the frequency of a term in a document with a factor that discounts its importance if it appears in most of the documents, as in this case the term is assumed to have little discriminating power.

3.1.3 $TF \times IDF$ Weighting with Length Normalization

In this approach, to account for documents of different lengths each document vector is normalized so that it is of unit length.

$$w_i = \frac{tf_i . \log\left(\frac{N}{N_i}\right)}{\sqrt{\sum_{j=1}^{|T|}\left[tfi . \log\left(\frac{N}{Ni}\right)\right]^2}}$$

This method performs better than other methods.

3.1.4 Mutual Information (MI)

This statistical method is used to determine whether a genuine association exists between two text features or not. In text classification, MI has been broadly employed in a variety of approaches to select the most significant text-features that serve to classify documents. MI between term t and class c is calculated by:

$$MI(t,c) \approx \log \frac{A_{ct} \times N}{(A_{ct} + C_{ct}) \times (A_{ct} + B_{ct})}$$

Here, A_{ct} is the number of times t and c co-occur, B_{ct} is the number of times t occurs without c, C_{ct} is the number of times c occurs without t, and N is the total number of documents. When t and c are independent MI (t,c) is equal to zero. E.g. in the given sports domain corpus, if we want to know is there any association between term ਵਿਕਟ (vikaṭ) and class ਕ੍ਰਿਕਟ (krikaṭ), MI plays an important role here.

3.2 Linguistic Approach

A number of rules specific for Punjabi language are formed to extract the language dependent features.

3.2.1 Name Rules

3.2.1.1 Person- Prefix

- If current word is prefix, its next word is taken as First Name.

3.2.1.2 Middle Name and last Name

- Word next to first name is checked for middle name or last name.
- If it is middle name, its next word is checked whether it is last name or not.
- If not, middle name is considered as last name.

3.2.2 Location Rules

- If Punjabi word ਵਿਖੇ (vikhē) is found, its previous word is extracted as location name.
- If Punjabi word ਪਿੰਡ (piṇḍ) is found, its next word is extracted as location name.
- If Punjabi word ਜ਼ਿਲ੍ਹੇ (zilhē) is found, its previous word is extracted as location name.

3.2.3 Date/Time Rule

- If month is found, it is extracted.
- If week day is found, it is extracted.
- If Punjabi words ਅੱਜ (ajj), ਕੱਲ (kall), ਸਵੇਰ (savēr), ਸ਼ਾਮ (shāmm), ਦੁਪਹਿਰ (duphir) etc. are found, they are extracted.

3.2.4 Numbers/Counting

- If any numeric character is found, it is extracted.
- If Punjabi words ਇੱਕ (ikk), ਦੂਜਾ (dūjā), ਦੋ (dō), ਪਹਿਲਾ (pahilā), ਛੇਵੀਂ (chēvīṃ) etc. are found, they are extracted.

3.2.5 Designation Rule

- If designation name found e.g. ਕਪਤਾਨ (kaptān), ਕੋਚ (kōc), ਕੈਪਟਨ (kaipṭan), it is extracted.

3.2.6 Abbreviation

- If words like ਆਈ (āī), ਸੀ (sī), ਐਲ (ail), ਪੀ (pī), ਬੀ (bī) etc. are found, they are extracted.

3.3 Gazetteer Lists

Due to the scarcity of resources in electronic format for Punjabi language, so the gazetteer lists are prepared manually from corpus and web.

- Person- Prefix
- Middle Names
- Last names
- Location Names
- Month Names
- Day Names
- Designation names
- Abbreviations
- Stop words
- Class-wise lists (e.g. preparing list for class ਕ੍ਰਿਕਟ (Cricket) that contain all of its related terms like ਬੱਲੇਬਾਜ਼ੀ (ballēbāzī), ਗੇਂਦਬਾਜ਼ੀ (gēndbāzī), ਫੀਲਡਿੰਗ (phīlḍiṅg), ਵਿਕਟ (vikaṭ), ਸਪਿਨ (sapin), ਆਊਟ (āūṭ), ਵਿਕਟਕੀਪਰ (vikṭakīpar) etc.

4. PROPOSED CLASSIFICATION ALGORITHM

Following is the proposed algorithm using Ontology Based Classification [32] to classify Punjabi text documents into eight predefined classes. These classes are: ਕ੍ਰਿਕਟ (krikaṭ), ਹਾਕੀ (hākī), ਕਬੱਡੀ (kabḍḍī), ਫੁਟਬਾਲ (phuṭbāl), ਟੈਨਿਸ (ṭainis), ਬੈਡਮਿੰਟਨ (baiḍmiṇṭan), ਓਲੰਪਿਕ (ōlmpik) and Others.

Step1: Remove all special symbols e.g. <,>, :,{,},[,],^,&,*,(,), extra tabs, spaces, shifts from the text documents.

Step2: Remove stopwords e.g. ਦੇ (dē), ਦਾ (dā), ਵਿੱਚ (vicc), ਦੀ (dī), ਹੈ (hai), ਇਹ (ih), ਅਤੇ (atē), ਵਲੋਂ (valōṃ),, ਹਨ (han), ਨੂੰ (nūṃ) using Punjabi Stopwords List.

Step3: Extract names, places, dates, months name etc. from the text document using Gazetteer lists.

Step4: Calculate term frequency (TF) for each remaining word.

Step5: Eliminate terms whose term frequency is below the threshold value.

Step6: Calculate Inverse Document Frequency (IDF) of each word from the document after pre-processing step.

Step7: Calculate $TF \times IDF$ of each word, and removing those words that are having $TF \times IDF$ value less than threshold value. This step will further help in reducing dimensionality.

Step8: Create ontology for each class that consists of terms related to its class. E.g. for Cricket Class Ontology, we have terms such as ਬੱਲੇਬਾਜ਼ੀ (ballēbāzī), ਗੇਂਦਬਾਜ਼ੀ (gēndbāzī), ਫੀਲਡਿੰਗ (phīlḍiṅg), ਵਿਕਟ (vikaṭ), ਸਪਿਨ (sapin), ਆਊਟ (āūṭ), ਵਿਕਟਕੀਪਰ (vikṭakīpar) etc. This results in Class-wise lists.

Step9: Remaining terms from Step7 is matched with each Class-wise list, and if maximum terms are matched with one class, assign that class to the unlabelled document.

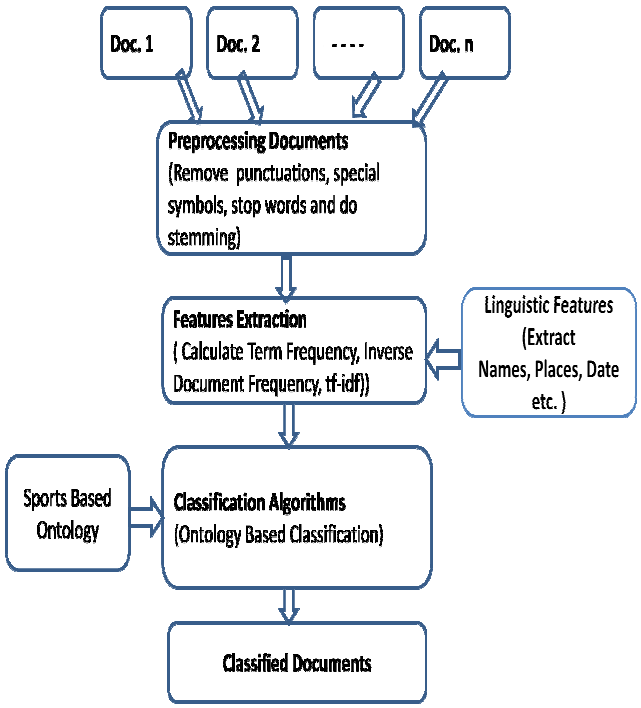

Figure 1: Algorithm for the Punjabi Text Document

INPUT: 150 Punjabi Text Documents (related to Sports only)

CLASSES: ਕ੍ਰਿਕਟ (krikaṭ), ਹਾਕੀ (hākī), ਕਬੱਡੀ (kabḍḍī), ਫੁਟਬਾਲ (phuṭbāl), ਟੈਨਿਸ (ṭainis), ਬੈਡਮਿੰਟਨ (baiḍmiṇṭan), ਓਲੰਪਿਕ (ōlmpik) and Others.

OUTPUT: This is the proposed results where each unlabelled document is classified into its class. Fig2 shows the distribution of unlabelled documents.

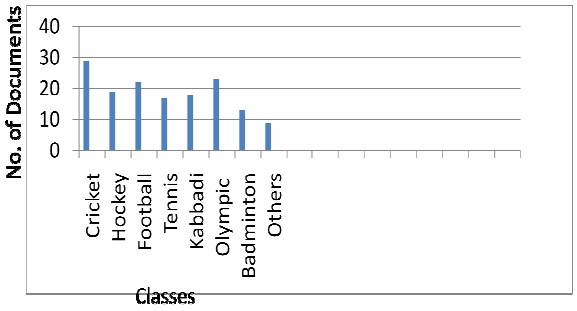

Figure2: Distribution of Unlabelled Documents into its Classes (Proposed Results)

5. DATASET

A corpus we used for text classification contains 150 Punjabi text documents. These documents are taken from the Punjabi news web sources such as likhari.org, jagbani.com, ajitweekly.com, punjabispectrum.com, europevichpunjabi.com, quamiekta.com, sahitkar.com, onlineindian.com, europesamachar.com, parvasi.com etc, but all the documents are Sports related.

As classification is a supervised learning, meaning we have predefined classes, so we have 8 classes for this corpus, these are: ਕ੍ਰਿਕਟ (krikaṭ), ਹਾਕੀ (hākī), ਕਬੱਡੀ (kabḍḍī), ਫੁਟਬਾਲ (phuṭbāl), ਟੈਨਿਸ (ṭainis), ਬੈਡਮਿੰਟਨ (baiḍmiṇṭan), ਉਲੰਪਿਕ (ōlmpik) and Others. Each class contains approx. 20-25 documents.

6. CONCLUSION

With the dramatic rise in the use of the internet, there has been an explosion in the volume of online documents. Text Classification (Text Categorization), the assignment of text documents to one or more predefined classes based on their content, is an important component in many information management tasks.

In this paper, we have used Domain (Sports) Based Ontology for the Classification of Punjabi Text Documents (related to Sports only).This is the proposed algorithm for the classification of Punjabi Text Documents where we have used C#.net (front end) and Microsoft Access 2007 (back end) to implement the algorithm. As not much work has been done in Punjabi Language, so in this approach we have made an initiation to create an ontology for Punjabi Language by creating Sports Based Ontology in Punjabi that consist of class related terms. E.g. in Cricket Class Ontology, it consists of words like wicket, bowler, bat etc. One of the major advantages of this ontology based classification is that we do not need Training Data i.e. Labeled Documents to classify the documents, whereas other Classification Techniques such as KNN technique, Naïve Bayes Algorithm, Association Based Classification etc. need Training Set or Labeled Documents to train the classifier to do the classification of the unlabelled documents.

As we have created only Sports ontology in Punjabi Language to classify Punjabi Sports Documents only. Therefore, in future, we can create Ontology for others domains too, to classify the documents from all other domains.

7. REFERENCES

[1] J.H. Kroeze, M.C. Matthee and T.J.D. Bothma**,** July 2007**,** "Differentiating between data-mining and text-mining terminology", "doi: *10.1.1.95.7062"*.

[2] F. Sebastiani, 2002 "Machine learning in automated text categorization", ACM Computer Surveys 34(1), 1–47.

[3] Nawei Chen and Dorothea Blostein, 2006, "A survey of document image classification: problem statement, classifier architecture and performance evaluation", Springer-Verlag, "doi: 10.1007/s10032-006-0020-2".

[4] Christoph Goller, Joachim Löning, Thilo Will and Werner Wolff*, 2009,* "Automatic Document Classification: A thorough Evaluation of various Methods", **"**doi=10.1.1.90.966".

[5] Kao, Anne, Poleet, R. Steve, "Natural Language Processing and Text Mining", (Eds.), 1st edition, 2007, XII, 265p, 655illus.

[6] Vishal Gupta, Gurpreet S. Lehal, August 2009 "A Survey of Text Mining Techniques and Applications", Journal of Emerging Technologies in Web Intelligence, VOL. 1, NO. 1.

[7] Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, "Introduction to Information Retrieval", Cambridge University Press. 2008.

[8] Yu Wang and Zheng-Ou Wang, 2007, " A Fast KNN Algorithm for Text Classification", Machine Learning and Cybernetics, International Conference on, Vol. 6, pp. 3436-3441, doi **:** 10.1109/ICMLC.2007.4370742, Hong Kong, IEEE.

[9] Wei Wang, Sujian Li and Chen Wang, 2008, "ICL at NTCIR-7: An Improved KNN Algorithm for Text Categorization", Proceedings of NTCIR-7 Workshop Meeting, December 16–19, Tokyo, Japan.

[10] Jiawei Han, Michelin Kamber, 2001, "Data Mining Concepts and Techniques", Morgan Kaufmann publishers, USA, 70-181.

[11] Jingnian Chen, Houkuan Huang, Shengfeng Tian and Youli Qu, 2009, "Feature selection for text classification with Naïve Bayes", Expert Systems with Applications: An International Journal, Volume 36 Issue 3, and Elsevier.

[12] Wen Zhang, Taketoshi Yoshida and Xijin Tang, 2008, "Text classification based on multi-word with support vector machine", Journal: Knowledge Based Systems – KBS, vol. 21, no. 8, pp. 879-886, doi: 10.1016/j.knosys.2008.03.044, Elsevier.

[13] Steve R. Gunn, 1998, "Support Vector Machines for Classification and Regression", University of Southampton.

[14] Wenmin Li, Jiawei Han and Jian Pei, 2001, "CMAR: Accurate and Efficient Classification Based on Multiple Class-Association Rules", IEEE International Conference on Data Mining - ICDM, pp. 369-376, DOI: 10.1109/ICDM.2001.989541.

[15] Xiaoxin Yin, Jiawei Han. CPAR*,* 2003," Classification based on Predictive Association Rules", in Proceedings of SDM, *doi=10.1.1.12.7268*.

[16] Fernando Berzal, Juan-Carlos Cubero, Nicolás Marín, Daniel Sánchez, Jose-María Serrano, Amparo Vila, *"*Association rule evaluation for classification purposes*"*.

[17] Chuntao Jiang, Frans Coenen, Robert Sanderson, Michele Zito, May 2010, "Text classification using graph mining-based feature extraction", Journal Knowledge-Based Systems Volume 23 Issue 4, Elsevier.

[18] Dat Huynh, Dat Tran, Wanli Ma, Dharmendra Sharma, 2011, "A New Term Ranking Method Based on Relation Extraction and Graph Model for Text Classification", Faculty of Information Sciences and Engineering, University of Canberra ACT 2601, Australia.

[19] Guoqiang Peter Zhang, November 2000, "Neural Networks for Classification: A Survey", IEEE Transactions on systems, man and cybernetics-Part C, Applications and Reviews, Vol. 30, NO. 4.

[20] Larry Manevitz, Malik Yousef, 2007, "One-class document classification via Neural Networks", Neurocomputing 70, 1466–1481, Elsevier.

[21] Darvinder kaur, Vishal Gupta, "A survey of Named Entity Recognition in English and other Indian Languages", Published in IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 6, November 2010.

[22] Kavi Narayana Murthy, "Automatic Categorization of Telugu News Articles", Department of Computer and Information Sciences, University of Hyderabad, Hyderabad, doi: 202.41.85.68.

[23] S. Mohanty , P. K. Santi , Ranjeeta Mishra , R. N. Mohapatra , Sabyasachi Swain, " Semantic Based Text Classification Using WordNets: Indian Language Perspective", doi=10.1.1.134.866.

[24] Abbas Raza Ali, Maliha Ijaz, "Urdu Text Classification", Published in FIT '09 Proceedings of the 7th International Conference on Frontiers of Information Technology, ACM New York, USA, 2009. ISBN: 978-1-60558-642-7 doi: 10.1145/1838002.1838025.

[25] P.Singh, A.Verma, N.S Chaudari, " Performance Analysis of flexible zone based features to classify Hindi numerals", Published in Electronics Computer Technology (ICECT), 3rd International Conference on 8-10 April 2011 on page 292-296, doi: 10.1109/ICECTECH.2011.5942101.

[26] K.Rajan, V. Ramalingam, M.Ganesan, S.Palanivel, B. Palaniappan, " Automatic Classification of Tamil documents using Vector Space Model and Artificial Neural Network", Published in: · Journal Expert Systems with Applications: An International Journal, Volume 36 Issue 8, October, doi: 10.1016/j.eswa.2009.02.010, 2009.

[27] George Forman, Evan Kirshenbaum, "Extremely Fast Text Feature Extraction for Classification and Indexing", Published in: Proceeding CIKM '08 Proceedings of the 17th ACM conference on Information and knowledge management ACM New York, NY, USA, 2008 ISBN: 978-1-59593-991-3, doi :10.1145/1458082.1458243.

[28] G S Lehal and Chandan Singh, "Feature extraction and classification for OCR of Gurmukhi script", Vivek, Vol. 12, No. 2, pp. 2-12 (1999).

[29] Punajbi Corpus

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

[31] Yanbo J. Wang , Frans Coenen , Robert Sanderson, "A Hybrid Statistical Data Pre-processing Approach for Language-Independent Text Classification", doi=10.1.1.157.6558, 2009.

[32] Guoshi Wu, Kaiping Liu, "Research on Text Classification Algorithm by Combining Statistical and Ontology Methods, IEEE International Conference on Computational Intelligence and Software Engineering, 11-13 Dec. 2009 , Pages 1-4, doi: 10.1109./CISE.2009.5363406.