Comparative Study on Punjabi Document Clustering

Iknoor Singh[1], Vijay Paul Singh[2], Naveen Aggarwal[3]

[1]Student, University Institute of Engineering and Technology, Panjab University
[2] Research Scholar, University Institute of Engineering and Technology, Panjab University
[3]Associate Professor, University Institute of Engineering and Technology, Panjab University
iknoor.ai@gmail.com

Abstract. The objective of clustering, a class of techniques that fall under the category of machine learning is to consequently isolate information into groups called clusters. Clustering of Punjabi documents finds numerous applications in the domain of natural language processing. Currently, not much work has been done for native languages such as Punjabi. This study presents the results of certain common document clustering techniques such as agglomerative and K-means experimented with different feature extraction methods to compare its performance using intrinsic and extrinsic measures. The recently released pre-trained Punjabi word vector model by Facebook has also been experimented as one of the feature extraction methods. This study is conducted to know which combination of clustering algorithm and feature extraction technique gives the most optimum results. This study also uses a supervised approach to evaluate the results of an unsupervised learning algorithm such as clustering.

Keywords: Document Clustering, Natural Language Processing, Text Mining, Punjabi Language;

1 Introduction

Document clustering is a technique for classifying a set of documents into a number of groups, based on certain distinguishing features, attributes and characteristic properties of the documents. The Internet contains a large number of high-dimensional data which must be categorized for certain reasons to enable efficient data processing and organization. For instance, E-commerce sites, blogs, social networking websites etc, can utilize various clustering techniques to this end. Text mining has become widely researched area with lots of applications [1]. Clustering techniques take advantage of the fact that the majority of the documents of a certain class contain similar kinds of words and frequency of these words can be used to predict the class to which the document belongs. Clustering of Punjabi documents finds numerous applications. For example in Plagiarism detection tool, documents can be checked in a particular cluster of similar documents instead of searching all the documents. This can confine the search to a limited number of

similar documents and hence speed up the search process. This study was embraced expressly for the clustering of Punjabi text documents as immensely less previous research has been done in this dialect as per the review of literature done to carry out this study [2]. All the open sourced text processing libraries are language-specific and generally work for English. Hence the main challenge is the preprocessing and vectorization of Punjabi documents. Corpus collection is another challenging task for the languages which are less available on the internet in abundance. Fortunately, many newspapers nowadays have their news content available in electronic format on the internet. Hence, a labelled Punjabi dataset is manually collected from various news sources. Most of the prior research work on clustering uses intrinsic measures such as silhouette score to evaluate the compactness of the cluster without the need of ground truth labels [3]. These measures lack the ability to evaluate the accuracy of the clustering model. For this study, the labelled dataset is collected to evaluate the performance of clustering using extrinsic measures such as Purity, F-measure and Adjusted Rand Index. Extrinsic measures will better evaluate the performance of clustering based on previous information about the dataset.

In the literature [4], many different text clustering algorithms have been proposed. All of these methods require several steps of initial preprocessing of the text. Initially, the normalization of documents is done which includes tokenization, elimination of stop word, stemming, punctuation removal, synonym replacement and format conversion. For document vectorisation, the statistical approach has been used which includes Bag-of-words vectorisation and TF-IDF vectorisation. Advanced word vectorisation such as average word vector and TF-IDF weighted average word vector are also implemented for the Punjabi text. These are done using FastText Punjabi word vector model [5]. Principal component analysis is used to reduce the dimensions of the feature vector and project the clustering results on a two dimensional graph. The cluster points are plotted with the actual ground truth labels represented with different shapes. This is done to know which cluster belongs to which label of the dataset, i.e. each cluster is allotted the category that is most frequent within the cluster[6]. This technique helps to identify the number of documents in each cluster that were correctly clustered. This was made possible because the manually created dataset is of small size with ground truth labels. Hence, the cluster predictions are evaluated against the true labels using metrics similar to ones used for classification evaluation (that is, based on true positive and negative and false positive and negative rates). Various performance metrics such as accuracy score, F1 Score and many more were used to test which combination of clustering algorithm and feature extraction technique performs best across the manually created dataset.

2 Related Work

In recent years, there has been extensive research going on the clustering of documents. It uses unsupervised machine learning algorithms that categorizes the documents into various clusters. The properties of these clusters are such that documents are similar and related within one cluster than documents of other

clusters[7]. Text clustering is very useful in various domains such as web page clustering, document summarization and sentiment analysis. Initially, a lot of attention was given to frequency of words for text clustering such as simple hybrid algorithm proposed by Le Wang et al.[8]. But now, more advanced approaches are being used for clustering. Tingting Wei et al.[9] proposed semantic approach using lexical chains and WordNet for text clustering. Amoli and Pedram Vahdani proposed a different technique approach clustering of scientific documents using summarisation [10]. This helps in reducing unnecessary data during clustering. Binyu Wang et al.[11], proposed a deep representation learning based algorithm for clustering using transfer learning domain adaptation. Recently, enhanced text clustering based on topic clusters was proposed by Fang Ji et al. [12]. It includes identification of the topic cluster relating to the set non-stop words found in the text and using polarity as the measure to differentiate between the clusters.

In this informative age, many documents are available in digital form in different Indian languages. According to India census, more than 19000 languages are spoken in India and mostly due to the proliferation of sources such as news, blogs, and other media sources, web data grow rapidly. Nidhi et al.[13], proposed normalization of Punjabi text documents and also a classification algorithm to classify the documents. The ontology and hybrid approach was also proposed for Punjabi text classification. Except these, independent research on various preprocessing tasks for Punjabi language have also been done. For instance, Vishal et al. [5] proposed a new approach on automatic stemming of Punjabi words using a statistical approach. In 2016, Jasleen et al. [14] presented a list of 184 stop words of Punjabi language, that is used in various applications related to NLP such as identification removal of stop words from the text. The recently released pre-trained Punjabi word embedding by Facebook's FastText library [5], provides standard vectorization tool for many NLP applications. The usage of these regional language embeddings is still scant and yet to be fully explored. Hence, this paper also uses FastText model to compare its performance. There has been some previous research work on domain based Punjabi text document clustering. For instance, Saurabh Gupta and Vishal Gupta [15,16] proposed a hybrid approach for Punjabi document clustering which takes into consideration the semantic relations between words. The most similar work to this paper is [17], which works on Punjabi documents clustering system where 221 Punjabi documents collected from various news websites were categorized into 7 Natural classes. The present study differs in two aspects. First, this study is conducted using different clustering algorithms with various feature extraction techniques to identify the best mechanism for clustering of Punjabi documents. Second, extrinsic measures are used which take into consideration the ground truth labels to evaluate the performance of clustering. The absence of real data with a correct ground truth label is a major challenge in the evaluation of clustering algorithms [18]. As a result, clustering research often relies on labelled data to evaluate and compare the results of clustering algorithms. Hence, the labelled data is collected and used with a different perspective by mapping the cluster results to a classification problem and thereby evaluating based on the ground truth labels.

3 Proposed Approach

3.1 Dataset

The dataset used in this study is created manually because the Punjabi language does not have any benchmark dataset available. Initially, the labelled dataset was collected from various Punjabi news websites such as *jagbani.com* and *ajitjalandhar.com*. A total of 150 documents were collected and this news dataset is categorized into five country classes called labels (see Table 1). The number of documents of each category is kept the same so as to evaluate the clustering performance accurately. The length of each document is also kept approximately the same. This raw set of documents is further processed to divide it into clusters.

Table 1. Punjabi News Dataset

Classes	Quantity
ਭਾਰਤ (India)	30
ਆਸਟ੍ਰੇਲੀਆ (Australia)	30
ਕਨੇਡਾ (Canada)	30
ਅਮਰੀਕਾ (America)	30
ਪਾਕਿਸਤਾਨ (Pakistan)	30
Total	150

3.2 Document Preprocessing

Document preprocessing includes the normalisation of the document text. Normalization of text consists of various steps such as wrangling, cleaning and standardizing of textual data into a form which could be used as an input to other NLP systems and analytical applications. The following pre-processing steps are performed.

Step 1. The punctuations are separated from the word by a space. This is done so that the punctuation do not remain attached with the word during the tokenization of documents. Example । , : , ? , (..etc. It is done with the help of *regex* using the substitute function.

Step 2. The digits and numbers are removed from the documents. Any special character is also removed such as @, $, %, &, ...etc.

Step 3. Word tokenization is done to split or segment sentences in documents into their constituent words. Tokenization process is important in information retrieval tasks as a token is the most basic unit of comparison.

Step 4. All the stopwords are removed from the documents. In general, stop words are the words most commonly found in a document and are of very little or no importance for various text processing tasks. The words such as ਦ , ਹ , ਦ , ਨ ◌ੰ , ◌ਿਸੰ ਘ, ਤ , ◌ਿਵਚ, ਦ , ਨੇ, ਅਤ , ਇਸ, ਤੋਂ etc are some of the most commonly used stop words in the Punjabi language [14]. Hence, the stopwords are removed by automatically checking the list of 143 Punjabi stopwords. Any remaining punctuations are also removed. This is also done by checking the list of Punjabi punctuations and removing them accordingly.

Step 5. Stemming of each tokenized word is done. Stemming can be defined as a process which combines the morphologically similar terms into one root word, which can be used to improve the process of information retrieval. [25]. Consider an example, where the word ਮੁੰਡੇ (munde) is mapped to stem word - ਮੁੰਡਾ (munda). This is done by truncating the suffix part and adding the relevant counterpart to reduce the word to its basic form.

Step 6. Synonym replacement is applied to the tokenized words. Punjabi synonyms are extracted from the Technology Development for Indian Languages (TDIL) synset dataset [23]. All the synonyms are kept in a list of lists. Now each tokenized word is checked. If any word is found in this list of lists, then that word is replaced by an index zero word in the list in which that particular word is found. This will help ensure some semantic similarity between documents at the time of clustering.

Step 7. Now after all the above steps, there remains some words with a single entity. Such words are of no significance. The length of such single words is three. Hence, these words are removed from the tokenized words. All the remaining tokenized words are converted into utf-8 format.

3.3 Feature Extraction Techniques

After the preprocessing phase, documents are vectorised to extract the relevant features. Although, there are different techniques of feature extraction for textual data, but this paper considers four principal strategies for the comparison purpose.

- *Bag of words(BoW):* For a BoW model, a sentence or document is seen as a bag that contain words. The words and their frequency in the sentence or the document is taken into consideration, regardless of any semantic relations present between them [24]. In eq.(1), *tf(d, t)* is the frequency of the term, $t \in T$ in document $d \in D$. The document is represented as an n-dimensional vector *td*.

$$td = (tf(d, t1), \ldots, tf(d, tn)) \quad (1)$$

- *Term Frequency - Inverse Document Frequency:* TF-IDF is a statistical approach to measure the importance of a word in a document or corpus [27].

Its significance increases based on number of times a specific word comes in the document but is compensated for by the occurrence of the word in the corpus. The general equation to measure TF-IDF is given in the eq.(2)

$$tfidf(d,t) = log(\frac{|D|}{df(t)}) \times tf(d,t) \quad (2)$$

In eq.(2), *t* and *d* are term and document respectively whereas, *df(t)* is the number of documents in which the term *t* appears.

- *Average Word Vectorisation Model:* Average word vectorization uses the Punjabi word vector model to compute the average of vectors of all words in a document. The word vector model is obtained from Facebook's FastText library [29]. This can be represented mathematically by the equation

$$AWV(D) = \frac{\sum_{1}^{n} wv(w)}{n} \quad (3)$$

In eq.(3), *AWV(D)* represents the Average word vector for a document *D*, containing words *w1, w2,…, wn* and *wv(w)* is the word vector representation for the word w.

- *TF-IDF Weighted Average Vectorisation Models:* TF-IDF weighted average word vector also uses the Punjabi word vector model, but in this each word vector is multiplied by its TF-IDF score and then average is taken. This can be represented mathematically by the equation

$$TW(D_\circ) = \frac{\sum_{1}^{n} wv(w) \times tfidf(w)}{n} \quad (4)$$

In eq. (4), *TW(D)* represents the TF-IDF weighted averaged word vector for $D_\circ$ document, containing words *w1, w2,…, wn,* where *tfidf(w)* represent the TF-IDF weight for the word *w* and *wv(w)* is the word vector representation.

3.4 Clustering Algorithms

A number of techniques and methods are available to cluster the given data. In this paper, some of the most common document clustering techniques such as hierarchical and centroid-based methods are used [7]. Hierarchical clustering models are also referred to as connectivity-based clustering methods. These are based on the idea that similar objects in the vector space are closer to the related objects than unrelated objects which are more distant. The clusters are formed based on their distance from the connected objects. These are primarily subdivided into agglomerative and divisive clustering models [19]. In case of centroid-based clustering models, clusters are constructed such that each cluster contains a central representative member, representing each cluster and having the characteristics that distinguish this particular

cluster from others. There are various centroid based algorithms, like k-medoids, k-means and so on, which requires initial declaration of number of clusters 'k' and minimizing the distance metrics like squares of distances from each data point to the centroid.

3.5 Principal Component Analysis

Semantically similar documents are grouped together in the clustering of documents. The dimensionality of documents to be clustered is often very large, containing thousands of terms. Therefore, the original dimensions is usually reduced to project the cluster results onto two dimensional graph [28]. Principal component analysis is one of the techniques used for dimensionality reduction. It provides a linear mapping of the data to a lower-dimensional space so as to maximize data variance in a low-dimensional representation [26]. In practice, eigenvectors are computed by constructing the covariance matrix of the data. In addition, the first few eigenvectors may often be interpreted in terms of the system's extensive physical behaviour. In our study, this technique is used to visualise the results of clustering on a two dimensional graph i.e. high dimensional data is reduced to a two dimensional data using Principal component analysis. Cluster points are plotted with different shapes according to their ground truth labels. This is done to know which cluster belongs to which label of dataset, that is, each cluster is assigned to the class which occurs most frequently in a cluster [22]. This is used to evaluate and compare the results with manually created labelled dataset using extrinsic measures discussed in the next section.

3.6 Testing Strategy

There are two types of performance evaluation strategies that are used for clustering techniques. First one is external evaluation in which there is some previous information about dataset such as the ground truth labels and second one is internal evaluation in which the evaluation is done in the absence of ground truth labels [20]. The typical objective of clustering is to attain high intra-cluster similarity and low inter-cluster similarity. This is known as internal criterion or intrinsic measures for evaluating the quality of a clustering. However, fine results on an internal criterion do not necessarily mean that clustering is effective. Hence there is a need of external evaluation metric or extrinsic measures which evaluate the clustering performance by taking into consideration the ground truth labels [21]. For external evaluation Purity score, F-measure, Normalized Mutual Information and Rand Index are used whereas for internal evaluation Silhouette score is used.

- Purity is used to measure the extent to which a cluster contains the cluster points of a single class. Formally, given some set of clusters C and some set of classes N, both partitioning D data points, purity is given by the eq.(5) as follows,

$$\frac{1}{D} \sum_{m \in C} max_{\, n \in N} |m \cap n| \tag{5}$$

- Mutual information is a theoretical measure of the extent to which information is shared between clustering and a ground truth labels which may observe non-linear likeness between two clusters. In eq.(6), Normalized Mutual Information (NMI) is calculated by normalizing the Mutual Information (MI) score. It gives the results from 0 to 1, where 0 means no mutual information and 1 indicates perfect correlation.

$$NMI(X, C) = \frac{2 \times I(X;C)}{[H(Y)+H(C)]} \tag{6}$$

  In eq (6), X is class labels, C is cluster labels, H represents Entropy and I(X;C) denotes the mutual information between Y and C.
- Rand index measures the proportion of correct decisions made by the clustering algorithm. Rand index is calculated using the following formula:

$$RI = \frac{TP + TN}{TP+FP+FN+TN} \tag{7}$$

  In eq (7), where TP, FP, TN & FN are the true positives, false positives, true negatives and false negatives respectively.
- F-measure is the weighted harmonic mean of recall and precision, In eq (8), β parameter determines the weight of precision in the resultant. If R is the recall rate and P is the precision rate, then F-measure is calculated using the following formula:

$$F_{\beta} = \frac{(\beta^2+1) \times P \times R}{\beta^2 \times P + R} \tag{8}$$

- Silhouette coefficient compares with the average distance between elements in the same cluster to the elements in other clusters. Its value varies from -1 and 1. Objects with a higher silhouette value are well clustered, whereas the objects with a lower value will be thought of as outliers. This metric works well with k-means clustering, and is also used to identify the optimal cluster counts.

4 Results

The results are obtained by running the clustering algorithm ten times by randomly shuffling the documents to be clustered. The following results are obtained by taking the average of all evaluation metrics. Table 2 and Table 3 shows the evaluation results of k-means and agglomerative clustering respectively. The graphical representation of the results can be seen in Fig. 1 and Fig. 2.

Table 2. Metrics Evaluation for K-means algorithm

K-means	Bag of words	TF-IDF	Average WordVec	TF-IDF weighted WordVec
Purity Score	0.4600	0.7100	0.6000	0.4100
NMI	0.1985	0.3885	0.3800	0.1225
Adjusted Rand Score	0.0681	0.3717	0.3067	0.0640
Precision	0.4600	0.7100	0.5900	0.4100
Recall	0.4600	0.7100	0.5900	0.4100
F-measure	0.4600	0.7100	0.5900	0.4100
Silhouette Score	0.0046	0.0094	0.0614	0.0578

Table 3. Metrics Evaluation for Agglomerative algorithm

Agglomerative	Bag of words	TF-IDF	Average WordVec	TF-IDF weighted WordVec
Purity Score	0.5700	0.4700	0.6100	0.4600
NMI	0.4151	0.3085	0.3499	0.2043
Adjusted Rand Score	0.1805	0.0532	0.2726	0.1083
Precision	0.7898	0.8301	0.6058	0.5428
Recall	0.5700	0.4700	0.5800	0.4500
F-measure	0.5740	0.4583	0.5897	0.4014
Silhouette Score	0.0326	0.0123	0.0558	0.0636

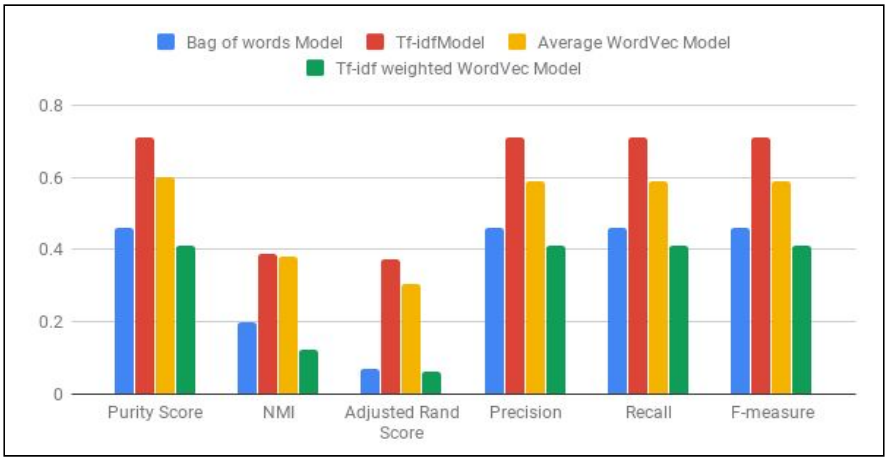

Fig. 1. Results of K-means clustering

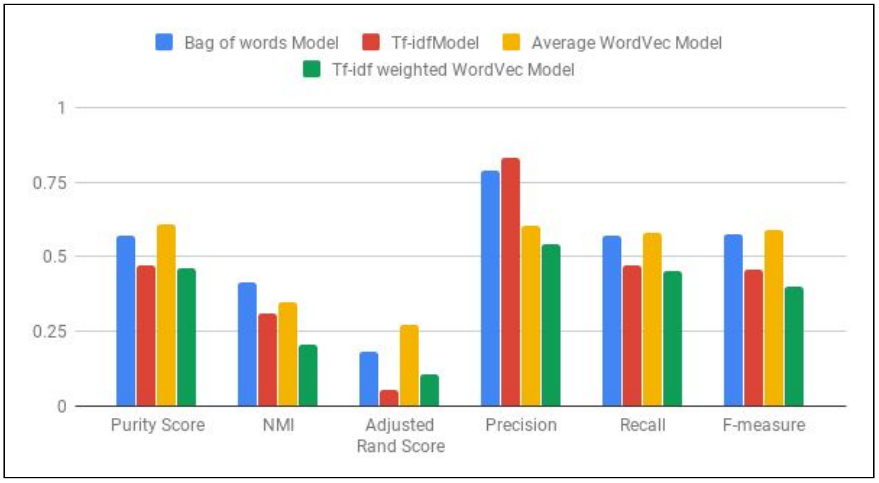

Fig. 2. Results of Agglomerative clustering

5 Conclusion and Future work

In this paper, clustering of Punjabi documents using different clustering algorithm with various feature extraction techniques was done. Proposed work shows that out of various feature extraction methods, TF-IDF with K-means algorithm outperforms all others with a purity score of 71%. In the case of agglomerative clustering, average word vector gives the best results with a purity score of 61%. Also, FastText

pre-trained model seems to give consistent results as compared to other feature extraction techniques. In case of F-measure score, K-means with TF-IDF and Agglomerative clustering with average word vectorisation gives the best results. As far as the intrinsic measure is concerned, agglomerative clustering seems to give consistently high scores as compared to K-means clustering. This study is conducted independently of the domain of the documents as the dataset consists of generic news articles of different countries. Therefore, the results are expected to be similar when tried on documents from other domains. These results can be enhanced through more efficient preprocessing and some better feature extraction and vectorization methods. In this study, some semantic similarity is ensured with the synonym replacement in pre-processing phase and the use of word vector model in feature extraction. For future, deep learning can be used with larger datasets for further enhancements and better results.

References

1. Gupta, Vishal, and Gurpreet S. Lehal. "A survey of text mining techniques and applications." Journal of emerging technologies in web intelligence 1.1 (2009): 60-76.
2. Kaur, Gagandeep, and Kamaldeep Kaur. "Sentiment Analysis on Punjabi News Articles Using SVM." Int. J. Sci. Res. 6.8 (2015): 414-421.
3. Baarsch, Jonathan, and M. Emre Celebi. "Investigation of internal validity measures for K-means clustering." Proceedings of the international multiconference of engineers and computer scientists. Vol. 1. sn, 2012.
4. Steinbach, Michael, George Karypis, and Vipin Kumar. "A comparison of document clustering techniques." KDD workshop on text mining. Vol. 400. No. 1. 2000.
5. Grave, Edouard, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. "Learning word vectors for 157 languages." arXiv preprint arXiv:1802.06893 (2018).
6. Schütze, Hinrich, Christopher D. Manning, and Prabhakar Raghavan. "Introduction to information retrieval." Proceedings of the international communication of association for computing machinery conference. 2008.
7. Aggarwal, Charu C., and ChengXiang Zhai. "A survey of text clustering algorithms." Mining text data. Springer, Boston, MA, 2012. 77-128.
8. Wang, Le, Li Tian, Yan Jia, and Weihong Han. "A hybrid algorithm for web document clustering based on frequent term sets and k-means." In Advances in web and network technologies, and information management, pp. 198-203. Springer, Berlin, Heidelberg, 2007.
9. Wei, Tingting, Yonghe Lu, Huiyou Chang, Qiang Zhou, and Xianyu Bao. "A semantic approach for text clustering using WordNet and lexical chains." Expert Systems with Applications 42, no. 4 (2015): 2264-2275.
10. Amoli, Pedram Vahdani. "Scientific Documents Clustering Based on Text Summarization." International Journal of Electrical & Computer Engineering (2088-8708) 5.4 (2015).
11. Wang, Binyu, Wenfen Liu, Zijie Lin, Xuexian Hu, Jianghong Wei, and Chun Liu. "Text clustering algorithm based on deep representation learning." The Journal of Engineering 2018, no. 16 (2018): 1407-1414.
12. Fang, Ji, Pablo Zivic, Yibin Lin, and Andrew Ko. "Enhanced text clustering based on topic clusters." U.S. Patent Application 10/049,148, filed August 14, 2018.

13. Gupta, Vishal, and V. Gupta. "Algorithm for punjabi text classification." International Journal of Computer Applications 37.11 (2012): 30-35.
14. Kaur, Jasleen, and Jatinderkumar R. Saini. "Punjabi Stop Words: A Gurmukhi, Shahmukhi and Roman Scripted Chronicle." Proceedings of the ACM Symposium on Women in Research 2016. ACM, 2016.
15. Gupta, Vishal, and V. Gupta. "Algorithm for punjabi text classification." International Journal of Computer Applications 37.11 (2012): 30-35.
16. Gupta, Vishal. "Punjabi text classification using Naive Bayes, centroid and hybrid approach." (2012).
17. Sharma, Saurabh, and Vishal Gupta. "Domain Based Punjabi Text Document Clustering." Proceedings of COLING 2012: Demonstration Papers. 2012.
18. Luu, Tuong. Approach to evaluating clustering using classification labelled data. MS thesis. University of Waterloo, 2011.
19. Berkhin, Pavel. "A survey of clustering data mining techniques." Grouping multidimensional data. Springer, Berlin, Heidelberg, 2006. 25-71.
20. Amigó, Enrique, Julio Gonzalo, Javier Artiles, and Felisa Verdejo. "A comparison of extrinsic clustering evaluation metrics based on formal constraints." Information retrieval 12, no. 4 (2009): 461-486.
21. Hossin, Mohammad, and M. N. Sulaiman. "A review on evaluation metrics for data classification evaluations." International Journal of Data Mining & Knowledge Management Process 5.2 (2015): 1.
22. Ding, Chris, and Xiaofeng He. "K-means clustering via principal component analysis." Proceedings of the twenty-first international conference on Machine learning. ACM, 2004.
23. Jha, Girish Nath. "The TDIL program and the Indian language corpora initiative." Language Resources and Evaluation Conference. 2012.
24. Joulin, Armand, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. "Bag of tricks for efficient text classification." arXiv preprint arXiv:1607.01759 (2016).
25. Gupta, Vishal, and Gurpreet Singh Lehal. "A survey of common stemming techniques and existing stemmers for indian languages." Journal of Emerging Technologies in Web Intelligence 5.2 (2013): 157-161.
26. Wold, Svante, Kim Esbensen, and Paul Geladi. "Principal component analysis." Chemometrics and intelligent laboratory systems 2.1-3 (1987): 37-52.
27. Ramos, Juan. "Using tf-idf to determine word relevance in document queries." Proceedings of the first instructional conference on machine learning. Vol. 242. 2003.
28. Napoleon, D., and S. Pavalakodi. "A new method for dimensionality reduction using k-means clustering algorithm for high dimensional data set." International Journal of Computer Applications 13.7 (2011): 41-46.
29. Gupta, Vishal. "Automatic stemming of words for Punjabi language." Advances in Signal Processing and Intelligent Recognition Systems. Springer, Cham, 2014. 73-84.