Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 125 (2018) 59–67

Procedia Computer Science

www.elsevier.com/locate/procedia

6th International Conference on Smart Computing and Communications, ICSCC 2017, 7-8 December 2017, Kurukshetra, India

Hindi Roman Linguistic Framework for Retrieving Transliteration Variants using Bootstrapping

Shashi Shekhar[a*],Dilip Kumar Sharma[a],M.M. Sufyan Beg[b]

[a]*GLA University,Mathura,281406,India*
[b]*Aligarh Muslim University,Aligarh,202002,India*

Abstract

Worldwide and colossal development of Web clients utilizing all web based applications in their local dialects for looking and separating data is a rising examination issue in the field of transliterated data retrieval. There is a developing need to help neighbourhood dialects in all web based applications by utilizing Machine Transliteration. There is a gigantic measure of client produced content in Roman content almost for each dialects which are composed in indigenous contents for some reasons. In the light of this wonder, the web crawlers confront a non-inconsequential issue of coordinating questions and reports in transliterated space where transliterated content contain broad spelling variety. This paper portrays our proposed technique to deal with such variety through non-straight dimensionality lessening. The assessment of the proposed framework and the outcome got connotes the change in giving the likely varieties to a term which are further valuable in assessing the word variations with use of different character sets. This paper describes the proposed method to handle such variation through non-linear dimensionality reduction techniques which enhances the possibility of use of variations for a term giving flexibility to end user to represent any word in other possible versions other than specified standards

Peer-review under responsibility of the scientific committee of the 6th International Conference on Smart Computing and Communications.

* Corresponding author
E-mail address: shashi.shekhar@gla.ac.in

1877-0509
Peer-review under responsibility of the scientific committee of the 6th International Conference on Smart Computing and Communications
10.1016/j.procs.2017.12.010

Keywords: Bootstrapping,Transliteration;Character mapping;Roman script;Term Variations

1. Introduction

The greater part of the Indian dialects like Bengali, Malayalam, Gujarati and other Indian dialects, are composed utilizing indigenous contents. In current situation on web numerous client created substance posted on facebook , tweets and websites are composed utilizing Roman content. This procedure of phonetically speaking to the expressions of a dialect in a non-local content is called transliteration. Presently a days, transliteration in Roman content, is utilized broadly Online for seeking reports and substance, yet additionally for client planned questions searching for these records or substance. All the more regularly there exist different Roman content transliterations for the local terms; for instance, the word Paani ("Water" in Hindi) can be composed in Roman content as paanii, paanie, paanee, paanei et cetera. This roman portrayal of a word in various arrangements and in various varieties makes a term coordinating issue for web indexes to coordinate the vernacular content question with the reports in numerous contents considering broad spelling variety. Roman Transliteration is by and by utilized wherever Online for archive hunting , as well as down detailing client questions for seeking reports on the web. As there is no standard strategy exists for composing a specific word utilizing roman in this manner transliterated substance will dependably have the issue of broad spelling varieties as any local term can be transliterated into Roman content from various perspectives. Transliteration process end up noticeably troublesome in nearness of loud words and out of vocabulary words which are available in the archives. Out of vocabulary (OOV) words are tricky in bilingual data recovery. The OOV words are named elements, similar to number, specialized terms and any acronyms. Late writing depicts that a bigger piece of web content is accessible in roman shape and is accessible just when a client utilizes roman based inquiry for looking through these information.

As the volume of data accessible in neighborhood or territorial dialects increments, there exists a requirement for keen apparatuses to perform productive seeking utilizing nearby dialects or utilizing blended content Nonetheless, as we find in this paper, blended content present different difficulties that the current methodologies for unraveling bilingual spelling variety and its transliteration in IR can't address consummately, particularly on the grounds that the vast majority of the transliterated inquiries have inquiry definition issue in setting to spelling varieties as far as composing a question. The paper focusses on issues of inquiry definition utilizing roman content and its related research challenges. The exploration says upwards of 9% of the inquiries utilized on the web for looking through any report utilizes Hindi roman transliteration, of which just 32% questions are unadulterated Named Elements (name of individuals, name of area and name of association and so on.). Then again, 33% of the sought inquiries are of amusement space (parts of tune verses and film discoursed). This legitimizes the utilization of utilizing transliterated questions on the web. In setting to India, Hindi melodies are additionally a standout amongst the most looked things, in this manner giving a knowledge to chip away at the zone of blended content recovery or cross dialect recovery. This inspired us to lead our exploration and proposed structure for separating data on Hindi melody verses by giving a model to standardize the varieties in composing any word while looking through the tune verses.

The paper proposes an all-inclusive answer to handle the spelling varieties if there should arise an occurrence of roman content utilized for retrieving any information. The proposed structure utilizes the blended content components for inquiry on learning condition where it can be contrasted and closest coordinating words that can be utilized as a part of place of that word as conceivable word variations.

The examination goal of this paper is to propose a system for looking through the substance utilizing roman transliteration by dealing with the varieties of a word written in roman transliterated shape. The remaining section provides the detailing about following: segment 2 depicts the related research in the field of transliteration considering Indian Dialects. Segment 3 diagrams the requirement for proposed structure and transliteration engineering for looking, sifting and positioning outcomes recovered from melody's verses. Segment 4 shows the execution particular subtle elements of the structure that we have worked to assess the execution of spelling variety standardization procedure using bootstrapping with Apriori lastly segment 5 talks about the conclusion alongside specific pointers towards future bearings in the field of effective transliterated retrieval.

2. Related Work

Transliteration particularly alludes to phonetics safeguarding translation of a word or a name from one dialect into the content of another dialect [1]. Transliteration now a days is vital for speaking to out-of-vocabulary words and for speaking to named elements as interpretation bombs in these spaces. Transliteration is a rising exploration zone in light of its broad applications in data recovery and in regions of outlining Input Method Editors (IME) for local dialects particularly for Indian dialects [2]. The unmistakable test additionally exists , as Hindi-English dataset for preparing and testing any transliteration framework barely accessible for open utilize. A significant number of the current transliteration motors depend on machine learning approaches (Linguistic based or SMT (Statistical Machine Transliteration) approaches [3] and subsequently the execution of the framework exceedingly relies on amount and nature of the preparation dataset. The underneath segment portrays the exploration embraced for transliteration in setting to Indian dialects.

The paper give a model answer for handling the spelling standardization of any roman word upto seven probable words by taking care spelling normalization regarding any inquiry word. The paper talks about the utilization of mixed script in the domain of self-learning environment to deliver likely seven word varieties. The proposed strategy can have the capacity to deliver equal words as of the inquiry word. The question word is then coordinated against the delivered counterparts. Tests have been done on blended content dataset of FIRE. The investigation result deliver fundamentally enhanced outcomes contrasted with the aggressive baselines with 17% change in MRR and 34% change in MAP over the current pattern.

This section, describes the notion of mixed script retrieval along the lines of cross lingual information retrieval. We also ponder current research issues in the area of mixed script information retrieval. To state these kind of existing situation in mixed script query formulation. This area, portrays the idea of mixed content recovery along the lines of cross lingual data recovery. To express these sort of existing circumstance in blended content question detailing, the accompanying idea has been utilized. Accept L be an arrangement of various local dialects (l1,l2,l3 and.... ln). The theory expresses that each dialect is composed utilizing a specific content, which we are alluding here as local content. Let Language li is utilizing local content si. Following this, content S = (s1,s2,s3 and.. sn) has a total arrangement of balanced mapping to Language L. The presence of any word composed utilizing local dialect made out of two traits, first the dialect the word has a place with and the content in which the word has been composed. We use the notation w $\in$ {li,sj} to mainly signify that w is in language li, written with the help of the script sj . When i = j, we can be able to justify that w exists in its native script else, we can conclude that w is available in its transliterated form, As discussed transliteration process is defined as the process of representing the phonetics of a word of one language into another script. The below section discusses certain literature review considering the domain of roman transliteration.

Ahmed et al. [4] describes the existence of spelling variation problems due to a lack of proper standardization for mapping local languages to the Roman script. The paper describes the scope to minimize the existence of variation in spellings for effective retrieval using roman transliteration. The work on searching Bollywood songs [5] also focussed on the identifying Hindi word variations written in Roman script. Related work by [6] goes into the details of word variations handling issues while extracting transliterated forms of Hindi song lyric. The use of Edit-distance approach significantly improved the results for the generation of such pairs [7] for English-Telugu, [8] for Tamil-English. The research in [9] describes a method for transliteration normalization of any text that by combining two techniques: a stemmer based method that focusses on deleting suffixes [10] and applying hand crafted rules for mapping the different word variants into a single word. A similar approach that uses stemming and grapheme-to-phoneme conversion is used by [11] to design a multilingual search engine for 10 different Indian languages. This gives enough research space in the area of normalizing the spelling variations during transliterated searching. Joshi H et al [12] have discussed syllabification approach for transliteration for Roman script to Devanagari script. They developed a system that retrieved the Hindi song lyrics written in the Roman script . They used the approach of statistical machine learning for transliteration and TF-IDF model for information extraction. Few rules were crafted for auto syllabification. Bhalla D. and Joshi N [13] presented the rule based transliteration system for English-Punjabi language pair [4]. The author uses the syllabification approach for converting English input to equivalent Punjabi output. The domain used for this is (NE) Named Entity. Below section describes the proposed framework

along with algorithm for finding probable variations of a roman transliterated word in probability of higher to lower existence in a given query term. The author [14] classify that words available in the query can be either content or intent. The major focus of this paper is to segment intent word from the content word. This classification is done assuming the content word must be able to represent the central topic of the query. Here users can add intent words to make their requirements more clear while searching. This seems to be one of the important research gap of identifying intent words from the content in regard to any query formulated in transliterated form for searching any document. This requires intelligent processing of intent words which can further be helpful in improving the quality of searched result. RishiRaj et al. [15] described about the Offline Script Identification (OSI) that promotes many important applications including automatic archiving of multilingual documents, examining online/offline archives of document images etc. Feature extraction and classification techniques are accompanied with the OSI to accomplish the different Indic scripts together. Script identification was used for reading multi-script documents in which different paragraphs, text lines, text-blocks written in different scripts are chiefly significant for use in a multiscript document. Subsequently while dealing with multi-script environment the script recognition also supports in text area identification, document sorting in digital libraries and video indexing and retrieval. The preliminary step in automatic processing of document images was the script or language identification.

3. Proposed Framework

The proposed and introduced structure is the expansion of [14], The created crawler conjure part web indexes and the client passes the input parameter written in roman transliteration to retrieve results based on input. The framework gathers the verses set into an accessible rundown store and would pre-process the verses content as indicated by characterized pre-preparing rules. Bootstrapping based Handcrafted rules are connected to standardize the likelihood of term varieties. The procedure of word arrangement is additionally taken care utilizing the character n-gram mapping strategy to extricate conceivable word varieties in roman form.

4. Framework Description

We have designed a crawling framework to crawl Hindi-English transliteration pairs from online Bollywood song lyrics. The crawled data is further used for training in the form of Backward and Forward transliteration. The figure 1 describes the schematic of our proposed approach. There are basically four major steps involved in this framework, First and foremost crawling the lyrics domain for creating data corpus of Bollywood Roman song lyrics is done. Secondly we have applied the pre-processing on the dataset by removing noisy elements from the crawled data, Thirdly we have performed the spelling variation normalization for creating an environment to ease the users to use any keywords as per their choice for representing any word as the proposed system can recognize a single word with its six different variants. Finally the work towards word level alignment of song lyrics and its roman transliteration is handled to generate Hindi-English transliteration pairs.

The proposed framework consists of component search engine along with following components. Engine selection module, Query execution module, Duplicacy elimination module and finally result aggregator module. The complete architecture of proposed system framework is illustrated in Figure 1.

The engine selection module provides the flexibility to select search engines. The role of engine selection module is more effective when we have a long list of options for selecting lyrics domain for searching. The query execution module is capable of managing multithreaded connection with selected engines. The duplicacy elimination module keep a check on lyrics url whether it has been crawler earlier or not by the means of Flag system and finally result aggregator module manages to perform intersection over the results to get unique lyrics set crawled from different domains.

The system work like this, In regard to a Query “q” and a Document “d”. The major task of an Information retrieval system is to present the ranking of a document “d” in such a way that the document relevant to query must be available at the top irrespective of the script of the query. To explain the same, let us assume that q $\in$ {l1,s1}. In

monolingual IR, D = {d1,d2,d3 and dn}contains those documents that are available in same script, i.e., for all k, dk $\in$ {l1, s1}. This scenario has been modified in regard to use of mixed script.

$$D = \bigcup D_i$$

Where $i=1$ to N

In this case $D_i = \{D_{i1}, D_{i2}$ and .. $D_{iN}\}$ denotes different documents available in language li, i.e., for all k, $D_{ik} \in \{l_i, s_i\}$.The below algorithm justifies the steps of Figure 1.

4.1 Word Alignment & Matching Algorithm

Step 1: Let no. of search engine used is S_e. Here value of S_e=3
Step 2: Let No. of URLs retrieved by S_e is T_e.
Set initial value of T_e=0
Step 3: Generate three threads for each S_e and execute Lyrics query.
For each S_e generate Thread T_e.
If $\mathbf{T_e \varepsilon S_e}$ then
Extract URLs and increment T_e by 1;
$T_e t \leftarrow T_e+1$
Step 4: Add Retrieved URLs to the .Lst (Repository for storing Lyrics)
Step 5: Check duplicate URLs in .Lst file
If duplicacy exists in .Lst **then** remove URLs from T_e.
$\mathbf{T_e \leftarrow T_e - 1}$
Step 6: Repeat Step 3 through Step 5 until t required number of URLs with each URL the downloaded lyrics is Retrieved (Taken 1000 words as max value).
Step 7: Retrieve URLs and add to $\mathbf{T_e}$ till $\mathbf{T_e}$=1000.
Step 8: Perform preprocessing on .lst
Step 9: Perform Linguistic operation on .lst (Handcrafted rules designed for variation normalization)
Step 10: Word level alignment on retrieved lyrics
Step 11: Train CRF Engine for MLM(Maximum Likelihood Matching)
Step 12: Produce Lyrics with different variations in writing pattern

The Apriori calculation with the negative bootstrapping is executed in this manner after the hash table arrangement as in [17,18]. At the point when the contribution for the Apriori with negative bootstrapping is gotten at first the word is handled to check whether it is a positive specimen or negative example. The positive example and the negative specimen is portrayed in view of the likeness score. On the off chance that it is a negative specimen then the yield is disposed of and it is not recorded in the database or the OOV (Out Of Vocabulary) and the database incorporates just the vocabulary of the diverse positive examples in the preparation stage which is used over the relating testing areas. The frequent matrix Apriori (FMA) algorithm is implemented which includes a set of association rules to associate the first word from the input to the corresponding second and thus identifying the song. The association rules includes the estimation of the support, confidence and the lift comprises of two steps such as

1. *Find all frequent item sets.*
2. *Then generate association rules with confidence above a minimum confidence threshold.*

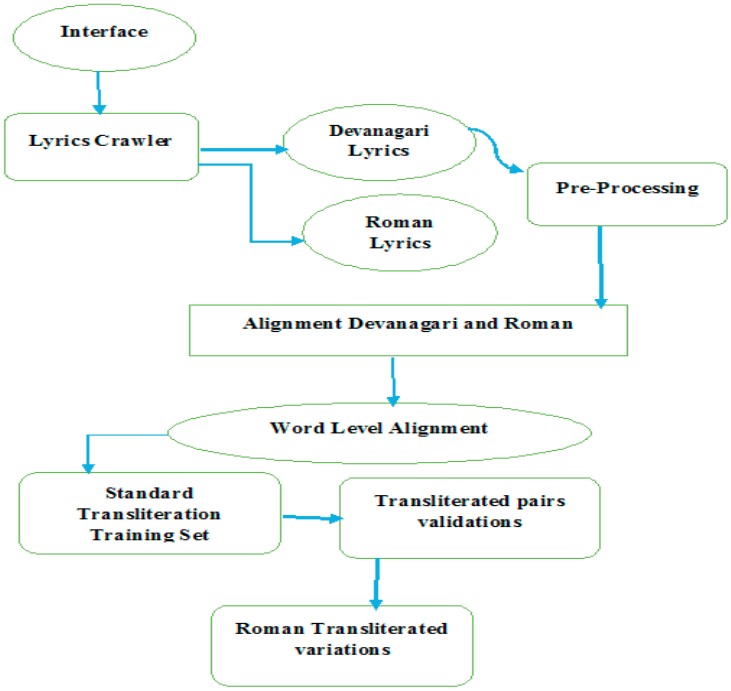

Fig 1. Schematic of the Lyrics Crawling Architecture

5. Experimental Evaluation

The proposed system's evaluation is performed on roman words considering different parameters in context to named entity and entertainment domain on MSIR data set of FIRE(Forum of Information retrieval and Evaluation[16]. The proposed system and its URL retrieval efficiency is evaluated using TREC Style Average Precision (TSAP) mechanism. In the experiment the precision value is calculated considering different queries. The average precision value for the queries is computes and is compared with the standard baseline. The next level of evaluation is done considering relevance of the retrieved results. Here relevance ration is computed for different query sets .Lastly the proposed lyrics retrieval performance is evaluated on the basis of following qualifiers: *Variations length, Variations accuracy.* These qualifiers act as an important evaluation factor to evaluate the efficiency of the proposed system to judge the performance on the basis of word variations normalization. At the end the performance evaluation is done on the basis of effective word variations retrieval.
The evaluation principle is based on following parameters:

- *Every document must contain words from Hindi and English languages.*
- *All the documents must be in Roman script.*
- *The length of the query word vary from 4 to 12 words.*

The baseline used here for evaluating the proposal is by assigning the label of the language classifier having the highest confidence score. In this case from available set of labels we randomly select a label and is assigned to the Wi. The set of available labels is obtained by setting a threshold value t on the confidence scores of the classifiers.

$$\delta(W_i=\text{argMax}\sum_{j-1}^{n}(Conf\_Score(Wi,Lj)=\frac{DOCn}{Wn} \quad (1)$$

5.1 Retrieval Effectiveness

The TSAP results considering different engines for different query levels have been computed and compared with proposed system as illustrated in Figure 2. The result graph is designed on the precision value calculated using the equation 1. The result suggests the retrieval effectiveness based on precision value outperforms when compared against other engines.

$$\text{TSAP@N} = \frac{\sum_{i-1}^{N} ri}{N} \quad (2)$$

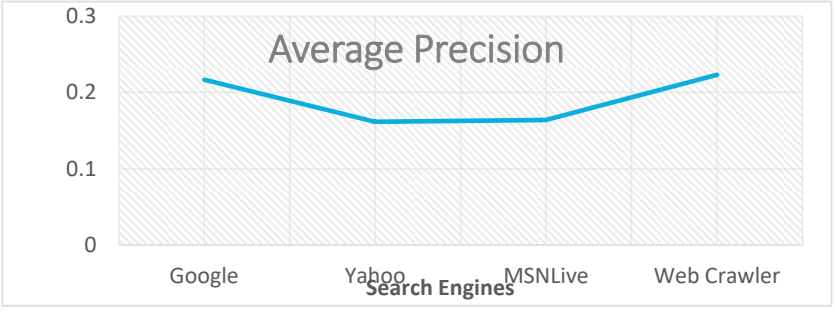

Fig 2. Average Precision Comparison

Table 1. Hash table representation for variation Accuracy

Words	**Possibilities**
ham	hum
	hamm
	humm
aise	ise
	aisae
	isae
	iseh
	aiseh
Paani	Paenee
	Paanie
	Paanei
	Paanii

Table 2. Word Accuracy on FIRE Data

System	Hindi	English
Random	0.179	0.832
MaxWeighted	0.423	0.782
Coverset	0.579	0.814
Optimal	0.782	0.864

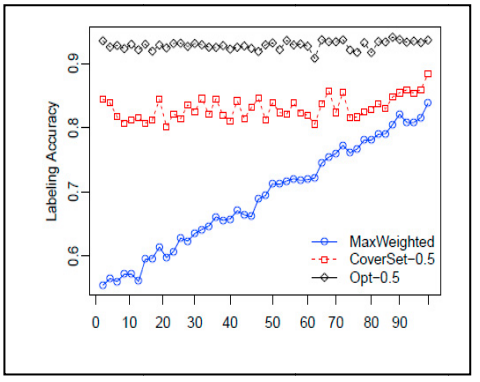

Fig 3. Loading accuracy

The Figure 4 below describes the implementation snapshot of word normalization module considering the case of roman transliteration variations. The word pani taken as word comes out with seven possible variations of the same word written differently using the concept of normalizing the variations.

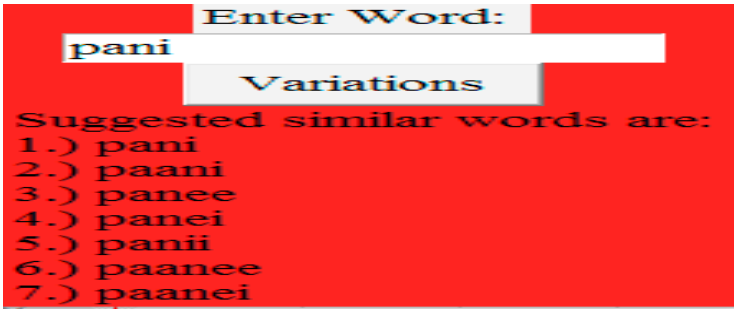

Fig 4. Variations Sample

Table 3. Result Comparison

Method	**Precision**	**Recall**	**F_1-sore**
Proposed Bootstrapping	0.94	0.75	0.83
Bi-directional mapping	0.80	0.73	0.76
RNNLM	0.9	0.7	0.7

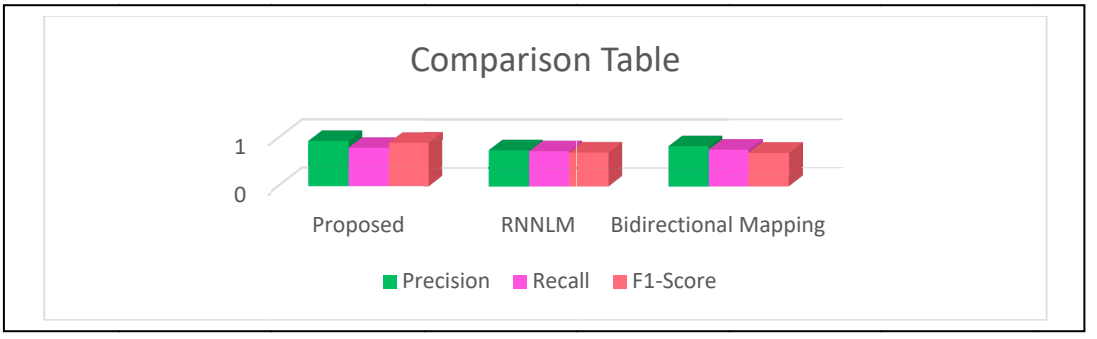

Fig 5. Comparison Result

6. Conclusion and Future work

The paper talks about the framework design alongside various segments of verses extraction and data set collection. The created framework incorporates the Apriori with negative bootstrapping to extract probable word variations. An exploratory model has been executed utilizing Python for recovering word varieties for a current word written in roman form. The outcome demonstrate our expansion outflank the current frameworks altogether. An imperative zone of research toward this path can be dialect identification in the field of linguistic based information mining in regard to transliteration.

References

1. Karimi S, Scholer F, & Turpin, (2011) “Machine Transliteration Survey”, ACM Computing Surveys, Vol. 43, No. 3, Article 17, pp.1-46.
2. Sowmya V. B., Monojit Choudhury, Kalika Bali, Tirthankar Dasgupta, and Anupam Basu.(2010). Resource Creation for Training and Testing of Transliteration Systems for Indian Languages. Proceedings of the Language Resource and Evaluation Conference (LREC) 2010.
3. Jong-Hoon Oh, Key-Sun Choi & Hitoshi Isahara, (2006) “A Comparison of Different Machine Transliteration Models”, Journal of Artificial Intelligence Research, pp. 119-151.
4. U. Z. Ahmed, K. Bali, M. Choudhury, and S. VB (2011).” Challenges in designing input method editors for indian languages: The role of Word origin and context. In Proceedings of the WTIM
5. N. Dua, K. Gupta, M. Choudhury, (2011)” Query completion without query logs for song search. In Proceedings of WWW (Companion Volume)
6. K. Gupta, M. Choudhury, and K. Bali. (2012) “Mining hindi-english transliteration pairs from online hindi lyrics. In Proceedings of LREC, pp 2459-2465.
7. V. B. Sowmya and V. Varma.(2009) “Transliteration based text input methods for telugu. In Proceedings of ICCPOL, pp. 122-132..
8. S. C. Janarthanam, S. Subramaniam, and U. Nallasamy,(2008) Named entity transliteration for cross-language information retrieval using compressed word format mapping algorithm. In Proceedings of iNEWS,pp. 33-38.
9. D. W. Oard, G.-A. Levow, and C. I. Cabezas,(2000) Clef experiments at maryland: Statistical stemming and backoff translation. In Proceedings of CLEF, pp. 176-187.
10. D. Pal, P. Majumder, M. Mitra, S. Mitra, and A. Sen, (2008)” Issues in searching for indian language web content. In Proceedings of iNEWS, pp. 93-96.
11. A. A. Raj and H. Maganti.,(2011) Transliteration based search engine for multilingual information access. In Proceedings of CLIAWS3, pp. 12-20.
12. Joshi H et al. (2013) “Transliterated Search using Syllabification Approach”, Forum for Information Retrieval Evaluation.
13. Bhalla, D. and Joshi, N, (2013) “Rule Based Transliteration Scheme For English To Punjabi”, International Journal on Natural Language Computing, Vol. 2, No. 2, pp. 67-73.
14. Shashi Shekhar et. Al, (2010) “An Architectural Framework of a Crawler for Retrieving Highly Relevant Web Documents by Filtering Replicated Web Collections” IEEE International Conference on Advances in Computer Engineering.
15. Rishiraj,Rahul, (2015)"Discovering and understanding word level user intent in Web search queries”In proceedings of Journal of Web Semantics on Semantic Search” Elsevier, 2015.
16. http://www. fire.irsi.res.in
17. Li X, Snoek CM, Worring M, Koelma D and Smeulders AW,“ Bootstrapping visual categorization with relevant negatives” IEEE Transactions on Multimedia,vol. 15, no. 4, pp. 933-45, Jun 2013.
18. Rehman, Shabnum, and Anil Sharma. "Privacy Preserving Data Mining Using Association Rule Based on Apriori Algorithm." In Advanced Informatics for Computing Research, pp. 218-226. Springer, Singapore, 2017.