ISSN 2007-9737

An Improvement in Statistical Machine Translation in Perspective of Hindi-English Cross-Lingual Information Retrieval

Vijay Kumar Sharma, Namita Mittal

Malaviya National Institute of Technology, Jaipur,
India

{2014rcp9541, nmittal.cse}@mnit.ac.in

Abstract. Cross-Lingual Information Retrieval (CLIR) enables a user to query to the different language target documents. CLIR incorporates a Machine Translation (MT) technique which is in growing state for Indian languages due to the unavailability of enough resources. In this paper, a Statistical Machine Translation (SMT) system is trained on two parallel corpora separately. A large English language corpus is used for language modeling in SMT. Experiments are evaluated by using BLEU score, further, these experimental setups are used to translate the Hindi language queries for the experimental analysis of Hindi-English CLIR. Since SMT does not deal with morphological variants while the proposed Translation Induction Algorithm (TIA) deals with that, therefore, TIA outperforms the SMT systems in perspective of CLIR.

Keywords. Cross-lingual information retrieval, parallel corpus, statistical machine translation, morphological variants.

1 Introduction

Nowadays, the Internet has overwhelmed by the multi-lingual content. Classical Information Retrieval (IR) considers the documents and sentences in other languages as unwanted noise [1]. Global internet usage statistics shows that the number of web access by the non-English users is tremendously increased, but all the non-English users are not able to express their queries in English[1], so there is a need for handling multiple languages arises which introduces a new area of IR that is Cross-Lingual Information Retrieval (CLIR).

CLIR provides the accessibility of relevant information in a language different than the query language [10], it can be presumed as a translation technique followed by monolingual information retrieval. CLIR follows two types of translation techniques, namely, query translation and documents translations.

A lot of computation time and space is elapsed in document translation, so query translation is preferred [11], where Dictionary-based Translation (DT), Corpus-based Translation (CT) and Machine Translation (MT), are the conventional translation techniques [19]. Manual construction of a dictionary is a cumbersome task and the MT internally uses a parallel corpus, therefore, the researchers put their efforts towards the development of effective and efficient MT system and corresponding translation resources.

In this paper, an SMT system is trained with the different experimental parameters to translate the Hindi language queries, which are evaluated by using BLEU score for MT and Mean Average Precision (MAP) for the Hindi-English CLIR. Since the SMT is not able to resolve the issue of morphological variants, therefore, a Translation Induction Algorithm (TIA) is proposed which incorporates morphological variants solutions.

The literature survey is represented in section 2. Section 3 discusses an SMT system. The proposed TIA is discussed in section 4. Experimental results and discussions are represented in section 5 and section 6 concludes the paper.

[1] http://www.internetworldstats.com

doi: 10.13053/CyS-22-4-3069

2 Literature Survey

The direct translation approaches, DT, CT, and MT, and the indirect translation approaches, Cross-Lingual Latent Semantic Indexing (CL-LSI), Cross-Lingual Latent Dirichlet Allocation (CL-LDA), Cross-Lingual Explicit Semantic Analysis (CL-ESA) are the CLIR approaches [14, 21]. A manual dictionary is used for translation and a transliteration mining algorithm is used to handle Out Of Vocabulary (OOV) words which are not present in the dictionary [16]. The transliteration generation or mining techniques are used to handle the OOV words [3, 13, 17].

Term Frequency Model (TFM) includes the concept of a set of parallel sentences and cosine similarity [15]. The dual semantic space based translation models CL-LSI, CL-LDA are effective but not efficient [18]. A Statistical Machine Translation (SMT) system is trained on a parallel corpus [5].

An open source machine translation toolkit *Moses*[2] was developed which was language-independent [8] where the phrasal translation technique enhances the power of MT [4]. Neural networks impart a significant role in the field of data mining. A Neural Machine Translation (NMT) system was developed and evaluated for the various foreign language but not for Hindi [20].

It is very tedious to develop and evaluate MT systems for Hindi-English language pair, a sentence-aligned parallel corpus *HindiEnCorp*[3] was developed and evaluated for MT system [2]. A recently developed sentence-aligned Hindi-English parallel corpus by IIT Bombay which is a superset of the *HindiEnCorp*, is experimented for the SMT and NMT system, and it is concluded that the SMT performs better than the NMT [9].

3 Statistical Machine Translation

SMT employs four components, i.e., word translation, phrasal translation, decoding, and language modeling [7].

3.1 Word Translation

An IBM model is used to generate the word alignment table from the sentence aligned parallel corpus. The Hindi and English language sentences are given as $h = \{h_1, h_2, ..., hm\}$ of length m, and $e = \{e_1, e_2, ..., en\}$ of length n.

An alignment function $a : j \rightarrow i$ for an English word e_j to a Hindi language word h_i is given as:

$$p(e,a|h) = \frac{\epsilon}{(m+1)^n}\prod_{j=1}^{n} t(e_j|h_{a(j)}), \tag{1}$$

where ϵ represents the normalization constant and $t(e_j|h_{a(j)})$ represents the translation probability.

A source language word is likely to be aligned with different target language words in the different iterations, so an Expectation Maximization (EM) algorithm is used to eliminate this issue. It follows an Expectation step where the probability of alignment is computed and a Maximization step where the model is estimated from the data. EM step is continuously applied until the convergence.

Expectation Step: The probability of alignment $p(a|e,h)$ is computed as:

$$p(a|e,h) = \frac{p(e,a|h)}{p(e|h)}, \tag{2}$$

where $p(e,a|h)$ computed by using equation 1, and $p(e|h)$ is calculated as:

$$p(e|h) = p(e,a|h), \tag{3}$$

$$p(e|h) = \frac{\epsilon}{(m+1)^n}\prod_{j=1}^{n}\sum_{i=1}^{m} t(e_j|h_i), \tag{4}$$

Maximization Step: It includes the collection count step where the sentence pairs (e,h), e is a translation of h, are counted:

$$c(e|h;e,h) = \sum_{a} p(a|e,h)\sum_{j=1}^{n}\delta(e,e_j)\delta(h,h_{a(j)}), \tag{5}$$

The different variation of IBM model and Hidden Markov Model (HMM) are used for word alignment. The GIZA++[4] tool implements an IBM Model 5 and HMM alignment model.

[2] http://www.statmt.org/moses/

[3] https://lindat.mff.cuni.cz/repository/

[4] https://github.com/moses-smt/giza-pp/blob/master/GIZA%2B%2B-v2/README

doi: 10.13053/CyS-22-4-3069

3.2 Phrasal Translation

Phrase model is not limited to only linguistic phrases which are noun phrases, verb phrases, prepositional phrases etc. It includes two steps, extraction of phrase pairs and scoring phrase pairs. The phrase pairs are extracted in such a way that they should be consistent with the word alignment. A phrase pair $(\bar{e}, \bar{h})$ is consistent with an alignment A, if all words $h_1, h_2, ..., h_l$ in $\bar{h}$, and $e_1, e_2, ..., e_l$ in $\bar{e}$ have the same alignment points in A and vice versa:

$$\begin{aligned} (\bar{e}, \bar{h})\ consistent\ with\ A \Leftrightarrow, \\ \forall e_i \in \bar{e} : (e_i, h_j) \in A \rightarrow h_j \in \bar{h}, \\ AND\ \forall h_j \in \bar{h} : (e_i, h_j) \in A \rightarrow e_i \in \bar{e}, \\ AND\ \exists\ e_i \in \bar{e}, h_j \in \bar{h} : (e_i, h_j) \in A. \end{aligned}$$

A translation probability is assigned to each phrase pair by calculating the relative frequency:

$$\phi(\bar{h}, \bar{e}) = \frac{count(\bar{e}, \bar{h})}{\sum_{h_i} count(\bar{e}, \bar{h_i})}. \qquad (6)$$

3.3 Decoding

The best target language translation e_{best} with the highest translation probability is identified at the decoding stage:

$$e_{best} = argmax_e\ p(e|h), \qquad (7)$$

$$e_{best} = argmax_e \prod_{i=1}^{l} \phi(\bar{h_i}, \bar{e_i})\ d(start_i - end_{i-1} - 1) p_{LM}(E), \qquad (8)$$

where $\phi(\bar{h_i}, \bar{e_i})$ represents the translation probability, $d(start_i - end_{i-1} - 1)$ represents the reordering component, and $p_{LM}(E)$ represents a n-gram language model to generate a fluent translation.

3.4 Language Modeling

A n-gram Language Model (LM) is used to generate a fluent translation output. LM follows n^{th} order Markov chain property:

$$p(w_1 w_2 w_3 ... w_n) = p(w_1) p(w_2|w_1) p(w_3|w_2 w_1) ...... p(w_n|w_{n-1} w_{n-2} ... w_1), \qquad (9)$$

$$p(w_1 w_2 ... w_n) = \prod_i p(w_i|w_1 w_2 ... w_{i-1}). \qquad (10)$$

4 Proposed Algorithm

A Translation Induction Algorithm (TIA) is proposed in Algorithm 1, which incorporates Refined Stop-Words and Morphological Variants Solutions.

Refined Stop-Words (RSW): Stop-words are the frequently occurring words which are considered as the noise in Mono-Lingual Information Retrieval (MoLiIR). In CLIR scenario, a source and target language stop word may have multiple meaningful target and source language translations respectively, hence, the stop-words impart a significant role in CLIR, such stop-words examples are presented in Table 1. The meaningful stop-words are eliminated from the source and target language standard stop-words lists, such meaningful stop-words are listed in Table 2.

Morphological Variants Solutions (MVS): The maximum Longest Common Subsequence Ratio (LCSR) score is used to select the approximate nearer word if a source language query word is not present in the exact form in parallel corpus, but the LCSR is not sufficient for morphologically rich language, therefore, following MVS solutions are additionally added to trace the approximate nearer word. An LCSR score between two strings a and b is computed as follows:

$$LCSR(a, b) = \frac{LCS(a, b)}{max(len(a), len(b))}, \qquad (11)$$

where LCS(a,b) returns longest common subsequence string between the strings a and b.

Table 1. List of stop-words and their translations

Stop-words	Translations
Against	खिलाफ (khilaf), विरुद्ध (Virudh), विपरीत (Vipreet), प्रतिकूल (Pratikool)
During	दौरान (Dauran), की अवधि में (Ki Avadhi Me), कालावधि तक (Kalavadhi Tak), पर्यन्त (Paryant)
बिल्कुल (Bilkul)	All, Completely, Perfectly, Quite
पूरा (Poora)	Complete , Finished, Total, Overall, Through

Table 2. List of meaningful stop-words for Hindi and English language

Hindi Stop - Words	English Stop - words
बिल्कुल (bilkul), निहायत (nihayat), वर्ग (varg), रखें (rakhen), काफी (kaffi), निचे (niche), पहले (pahle), अंदर (andar), भीतर (bheetar), पूरा (poora), गया (gaya), बनी (bani), बही (bahi), बीच (bich)	About, above, after, again, against, all, because, before, below, between, but, down, during, few, more, most, off, only, ought, out, over, own, some, than, through, too, under, up

Equality of nukta character with the corresponding non-nukta character: LCSR is unable to detect the equality between the nukta and non-nukta characters. The words with nukta characters are like सड़क (sadak), लड़ाई (ladai), परवेज़ (parvez). Target documents contain many words with nukta and non-nukta characters, so an equality solution is applied where nukta character and non-nukta characters are equally considered.

Auto-correction of query words: Query words are searched in the parallel corpus as they appear. The correctness of the query words is not verified. A word's popularity based correctness solution is applied, where a query word's frequency wf_i is computed over the corpus and compares it against the empirically defined threshold T. If wf_i is less than T, then the nearest word's frequency cwf_i, of the query word is computed with the help of LCSR. If $cwf_i > wf_i$, then query word is replaced by its nearest word. The examples of such words are shown in Table 3.

Equality of chandra-bindu with *म् and न्*: A query word with chandra-bindu may be equivalent to many other words, like a word "अंबानी"(Ambani) has similar LCSR score of 0.83 with these three words "अम्बानी"(Ambani), "अंबाजी"(Ambaji), "अल्बानी"(Albani). If chandra bindu is considered as equivalent to "म्" then the word "अम्बानी" has the maximum LCSR score.

Auto-selection of the nearest query word: An LCSR score is used to select the nearest word if a word is exactly not found in the PC. A word may have multiple nearest words with the similar LCSR score as shown in Table 4. A Compressed Word Format (CWF) algorithm [6] is used for auto-selection of the nearest query words, so far, the CWF algorithm is used for transliteration mining. Further, a set of parallel sentences is selected for each query word w_i from the parallel corpus, in a contextual manner such that each sentence contains either all three words of tri-gram and both of the words of bi-gram independent of word order, with the inclusion of w_i.

5 Experiment Results and Discussion

If the number of selected parallel sentences is less than a threshold t, then z number of unigram based parallel sentences of minimum length is also included. These context-based selected sentences return the appropriate translation.

FIRE[6] 2010 and 2011 datasets are used to evaluate the CLIR system, while the WMT [7] news

[6] http://fire.irsi.res.in/fire/static/data

[7] http://www.statmt.org/wmt15/translation-task.html

Table 3. Auto-corrected words

Query Word	Frequency	Closest Word	Frequency
मसजिद (Masjid)	4	मस्जिद (Masjid)	229
सियाचिन (Siachen)	2	सियाचीन (Siachen)	6
मुसलिम (Muslim)	3	मुस्लिम (Muslim)	947

Table 4. Multiple closest words with same LCSR score

Query Word	Corpus Word	LCSR Score
गुटखा (Gutkha)	गुइटा (Guita)	0.8
	गुटखे (Gutkhe)	0.8
	गुरखा (Gurkha)	0.8

test-set 2014 is used to evaluate the MT system. The dataset and resources which are used for MT and CLIR, are represented in Table 5 and 6. All three experimental setups of MT system are tuned and evaluated by using the common dev_set and test_set.

An SMT system is evaluated by using the BLEU score which computes the N-gram overlap between the MT output and the referenced translation. It computes precision for N-grams of size 1 to 4, which is given as:

$$precision = \frac{correct\ \ translation}{translation\ \ length}, \tag{12}$$

BLEU score is computed for the entire corpus not for a single sentence [7]:

$$BLEU = \\ min(i, \frac{output\ -\ length}{reference\ -\ length})(\prod_{i=1}^{4} precision_i)^{\frac{1}{4}}. \tag{13}$$

A CLIR system is evaluated by using Recall and Mean Average Precision (MAP). The Recall is the fraction of relevant documents that are retrieved as shown in Equation 14. MAP for a set of queries is the mean of the average precision score of the queries. Precision is the fraction of retrieved documents that are relevant to the query.

Average precision of a query is calculated in Equation 15:

$$Recall = \\ \frac{|\{relevant\ documents\} \bigcap \{retrieved\ documents\}|}{|\{relevant\ documents\}|}, \tag{14}$$

$$Average\ Precision = \\ \frac{\sum_{k=1}^{n}(p(k) \times rel(k))}{Number\ of\ relevant\ documents}. \tag{15}$$

Where k is the rank in the sequence of retrieved documents, n is the number of retrieved documents, $p(k)$ is the precision at rank k, $rel(k)$ is equal to 1 if the document at rank k is relevant otherwise 0.

5.1 Experimental Setup

User queries are translated by using an SMT system which is trained with three different experimental setups as follows.

— SMT_setup1, HindiEnCorp is used for both of the purposes of training and language modeling.

— SMT_setup2, A Hindi-English parallel corpus developed by IIT Bombay is used for both of the purposes of training and language modeling.

Algorithm 1: Translation Induction Algorithm

```
Input:  Source Language Query $SLQ[w_1, w_2, ..., w_m]$ and a Parallel Corpus  $PC[e_1, e_2, ..., e_n]$ where each
entry of the parallel corpus $e_i$ contains the Source Language Sentence (SLS) and corresponding Target
Language Sentence (TLS)
Output:  Best Target Language Translation (TLT) for each query word

Step 1: Remove source language stop-words (Refined Stop-Words) from the SLQ and initialize a Temporary
        Corpus TC=[ ]
Step 2: SLQ term selection: Verify that each SLQ term is available in PC
Step 2.1: If a SLQ term is exactly not found in PC then approximate nearer term is chosen from the PC by
          Longest Common Sub-sequence Ratio (LCSR) and replace the SLQ term
Step 2.2: If a SLQ term is exactly not found by LCSR, then Morphological Variants Solution are used to
          select the approximate nearer word and replace the SLQ term
Step 3: For each term $w_i$ , Generate distinct Tri-Gram Pairs TGP[$w_i$] from the SLQ
Step 3.1: Tri-Gram Count TGC[$w_i$]=0
Step 3.2: For each TGP[$w_i$]
            Select the sentence $e_i$ from the PC such that the corresponding SLS contain all the three words,
            order independently and add the selected sentence to TC
            TGC[$w_i$]+=1
Step 4: For each term $w_i$ , Generate distinct Bi-Gram Pairs BGP[$w_i$] from the SLQ
Step 4.1: Bi-Gram Count BGC[$w_i$]=0
Step 4.2: For each BGP[$w_i$]
            Select the sentence $e_i$ from the PC such that the corresponding SLS contain both the words,
            order independently and add the selected sentence to TC
            BGC[$w_i$]+=1
Step 5: For each SLQ term $w_i$, if TGC[$w_i$]+BGC[$w_i$] $<$ t, where t is a threshold
Step 5.1: For each SLQ term $w_i$, select the z number of minimum length sentences which contain term $w_i$,
            from the PC and add the selected sentence to TC
Step 6: Construct a Term Frequency – Inverse Document Frequency (TF-IDF) matrix for the TC which have
            the vectors only for target language terms and source language query words
Step 7: Cosine Similarity Score is used to select the best TLT for each query word $w_i$
```

— SMT_setup3, A parallel corpus developed by the IIT Bombay is used for training, while the WMT news corpus 2015 is used for language modeling.

These experimental setups are tuned by using the common dev_set and test_set, which is shown in Table 5.

Fire 2010 and 2011 Hindi language query sets are translated by using the different SMT setups and the proposed approach, further, these translated queries are used to retrieve the target English language documents. HindiEnCorp is utilized as a parallel corpus in the proposed approach.

The Terrier[8] open source search engine is used for indexing and retrieval. In our experiments, Terrier uses Term Frequency-Inverse Document Frequency (TF-IDF) for indexing and cosine similarity for retrieval.

5.2 Results and Discussions

An SMT system is trained in three ways, which are evaluated by using the BLEU score [12]. These trained SMT systems are evaluated for five different test_set, their BLEU scores are represented in Table 7. The News test_set 2014, Fire 2008, 2010, and 2011 test sets are evaluated against the corresponding human translated text, while Fire 2012 test set is evaluated against the Google translated text because the human translated text for Fire 2012 is not available.

SMT_setup2 performs better than the SMT_setup1 for all five test cases. The

[8] http://terrier.org/download/

Table 5. The Characteristic of the MT Dataset and Resources

Training_set	Language Modeling	Dev_set	Test_set
HindiEnCorp	HindiEnCorp		WMT news test_set 2014 (2507 sentences), and Fire 2008,2010,2011, and 2012 query set (each have 50 sentences)
IIT Bombay[5] (1,492,827 sentences)	IIT Bombay	WMT Dev_set (520 sentences)	
	WMT News 2015 Corpus (3.3 GB)		

Table 6. CLIR Dataset characteristic

Dataset	Fire 2010		Fire 2011		HindiEnCorp
Characteristic	Query	Document	Query	Document	Parallel Corpus
Number of queries/sentence/documents	50	125586	50	392577	273886
Average length (Number of Tokens) of query/sentence/document	6	264	3	245	20

performance of SMT_setup2 and SMT_setup3 are approximately similar, SMT_setup2 performs better for the first three test cases while in the last two test cases, the performance of SMT_setup3 is better.

Now, these SMT systems and the proposed TIA are evaluated for CLIR by using Recall and MAP, which is represented in Table 8.

The SMT_setup1 performs better than the SMT_setup2 and SMT_setup3 in perspective of CLIR. The SMT_setup1 is trained on HindiEnCorp which is smaller than the IIT Bombay parallel corpus, used in SMT_setup2 and SMT_ setup3. Although the parallel corpus developed by IIT Bombay is a superset of HindiEnCorp, it is not so well-organized and mixes the noise in the translation, hence, the translation performance is poor in perspective of CLIR. The SMT_setup3 uses WMT news corpus 2015 for language modeling, so it performs a little better than the SMT_setup2.

The proposed approach utilizes the well-organized HindiEnCorp as a parallel corpus. Refined stop-words and Morphological Variants Solutions improve the Recall and MAP for both of the Fire 2010 and 2011 datasets. In the perspective of CLIR, the proposed approach outperforms the Hindi-English SMT system which is trained on the best available resources.

6 Conclusion

CLIR retrieves the target documents which are in a language different than the query language, with the help of an MT technique. Source language user queries are translated by using different SMT setups and the proposed approach. HindiEnCorp is smaller than the parallel corpus developed by IIT Bombay, but it is better organized than the IIT Bombay corpus. SMT_setup1 performance is a little poor in perspective of MT system, but in perspective of CLIR, its performance is better than the other SMT setups.

Stop-words impart a significant role in CLIR, SMT does not deal with the stop-word and the morphological variant. The proposed approach improves the results and outperforms the SMT systems, as it deals with the stop-words and morphological variants.

Table 7. SMT evaluation results (BLEU)

Setups	News test_set 2014	Fire 2008	Fire 2010	Fire 2011	Fire 2012
SMT_setup1	7.05	10.76	4.48	8.13	17.11
SMT_setup2	9.70	11.72	6.75	6.53	17.59
SMT_setup3	8.95	11.45	5.13	8.77	17.75

Table 8. CLIR evaluation results

Setups	Fire 2010		Fire 2011	
	Recall	MAP	Recall	MAP
SMT_setup1	0.8575	0.2382	0.7088	0.1885
SMT_setup2	0.7718	0.2075	0.6602	0.1608
SMT_setup3	0.7978	0.1994	0.6602	0.1767
Proposed Approach	0.8685	0.2818	0.7195	0.1816

References

1. **Abusalah, M., Tait, J., & Oakes, M. (2005).** Literature review of cross-language information retrieval. *Transactions on Engineering, Computing and Technology, ISSN*, Citeseer.

2. **Bojar, O., Diatka, V., Rychlỳ, P., Stranák, P., Suchomel, V., Tamchyna, A., & Zeman, D. (2014).** HindEnCorp-Hindi-English and Hindi-only corpus for machine translation. *LREC*, pp. 3550–3555.

3. **Ganesh, S., Harsha, S., Pingali, P., & Verma, V. (2008).** Statistical transliteration for cross language information retrieval using HMM alignment model and CRF. *Proceedings of the 2nd Workshop on Cross Lingual Information Access*.

4. **Green, S., Cer, D., & Manning, C. (2014).** Phrasal: A toolkit for new directions in statistical machine translation. *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pp. 114–121.

5. **Jagarlamudi, J. & Kumaran, A. (2007).** Cross-lingual information retrieval system for indian languages. *Workshop of the Cross-Language Evaluation Forum for European Languages*, Springer, pp. 80–87.

6. **Janarthanam, S. C., Subramaniam, S., & Nallasamy, U. (2008).** Named entity transliteration for cross-language information retrieval using compressed word format mapping algorithm. *Proceedings of the 2nd ACM workshop on Improving non english web searching*, ACM, pp. 33–38.

7. **Koehn, P. (2009).** *Statistical machine translation*. Cambridge University Press.

8. **Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al. (2007).** Moses: Open source toolkit for statistical machine translation. *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, Association for Computational Linguistics, pp. 177–180.

9. **Kunchukuttan, A., Mehta, P., & Bhattacharyya, P. (2017).** The IIT Bombay English-Hindi parallel corpus. *arXiv preprint arXiv:1710.02855*.

10. **Nagarathinam, A. & Saraswathi, S. (2011).** State of art: Cross lingual information retrieval system for Indian languages. *International Journal of Computer Applications*, Vol. 35, No. 13.

11. **Nasharuddin, N. A. & Abdullah, M. T. (2010).** Cross-lingual information retrieval: State-of-the-art. *Electronic Journal of Computer Science and Information Technology: eJCIST*, Vol. 2, No. 1.

12. **Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002).** BLEU: a method for automatic evaluation of machine translation. *Proceedings of the 40th annual meeting on association for computational linguistics*, Association for Computational Linguistics, pp. 311–318.

13. **Saravanan, K., Udupa, R., & Kumaran, A. (2010).** *Crosslingual information retrieval system enhanced with transliteration generation and mining*. Citeseer.

14. **Sharma, V. K. & Mittal, N. (2016).** Cross lingual information retrieval (CLIR): Review of tools, challenges and translation approaches. In *Information Systems Design and Intelligent Applications*. Springer, pp. 699–708.

15. **Sharma, V. K. & Mittal, N. (2016).** Exploiting parallel sentences and cosine similarity for identifying target language translation. *Procedia Computer Science*, Vol. 89, pp. 428–433.

16. **Sharma, V. K. & Mittal, N. (2018).** Cross-lingual information retrieval: A dictionary-based query translation approach. In *Advances in Computer and Computational Sciences*. Springer, pp. 611–618.

17. **Shishtla, P., Ganesh, V. S., Subramaniam, S., & Varma, V. (2009).** A language-independent transliteration schema using character aligned models at NEWS 2009. *Proceedings of the 2009 Named Entities Workshop: Shared Task on Transliteration*, Association for Computational Linguistics, pp. 40–43.

18. **Vulić, I., De Smet, W., & Moens, M.-F. (2013).** Cross-language information retrieval models based on latent topic models trained with document-aligned comparable corpora. *Information Retrieval*, Vol. 16, No. 3, pp. 331–368.

19. **Wang, A., Li, Y., & Wang, W. (2009).** Cross language information retrieval based on lda. *Intelligent Computing and Intelligent Systems, 2009. ICIS 2009. IEEE International Conference on*, volume 3, IEEE, pp. 485–490.

20. **Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al. (2016).** Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

21. **Zhou, D., Truran, M., Brailsford, T., Wade, V., & Ashman, H. (2012).** Translation techniques in cross-language information retrieval. *ACM Computing Surveys (CSUR)*, Vol. 45, No. 1, pp. 1.

Article received on 12/12/2017; accepted on 15/02/2018.
Corresponding author is Vijay Kumar Sharma.