Improvement for Statistical Machine Translation Applied to Hindi-English Cross-Lingual Information Retrieval

Vijay Kumar Sharma[1], Namita Mittal[1]

Malaviya National Institute of Technology Jaipur, India
{2014rcp9541, nmittal.cse}@mnit.ac.in

Abstract. Cross-Lingual Information Retrieval (CLIR) system incorporates a Machine Translation (MT) technique and an Information Retrieval module. The MT techniques are in growing state for Indian languages due to the unavailability of enough resources. In this paper, a Statistical Machine Translation (SMT) system is trained over the two parallel corpora separately. A large mono-lingual English language corpus is used to train the language modeling module in SMT. Different SMT experimental setups are prepared to translate the Hindi language queries for the experimental analysis of Hindi-English CLIR. The SMT systems don't deal with morphological variants while the proposed Translation Induction Algorithm (TIA) deals with that. The TIA outperforms the SMT systems in perspective of CLIR.

1 Introduction

Nowadays, the Internet has overwhelmed by the multi-lingual content. The classical Information Retrieval (IR) normally regards the documents and sentences in other languages as unwanted noise [1]. Global internet usage statistics shows that the numbers of web access by the non-English users are continuously increasing, but all of them are not able to express their queries in English [1]. The needs for handling multiple languages introduce a new area of IR that is Cross-Lingual Information Retrieval (CLIR). The CLIR provides the accessibility of relevant information in a language different than the query language [4]. The CLIR can be presumed as a translation mechanism followed by monolingual information retrieval. Two types of translation mechanism are followed in CLIR, namely, query translation and documents translation. A lot of computation time and space is elapsed in document translation approach, so query translation approach is preferred [3]. The Dictionary-Based Translation (DT), Corpus-Based Translation (CT) and Machine Translation (MT), are the conventional translation approaches[2]. Construction of manual dictionary is a cumbersome task, and the MT approaches internally use the parallel corpus, so the researchers put their efforts towards the development of effective and efficient MT systems and corresponding translation resources.

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

In this paper, Different SMT systems are trained with the different experimental parameters, which are evaluated by using the BLUE score in perspective of translation accuracy and Mean Average Precicion (MAP) in perspective of Hindi-English CLIR. Since the SMT system could not solve the problems of morphological variants, hence a Translation Induction Algorithm (TIA) is proposed. The literature survey is represented in section 2. Section 3 discusses an SMT system. TIA is proposed in section 4. The experimental results and discussions are represented in section 5. Section 6 represents the conclusion.

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 used for query translation in Cross-Lingual Information Retrieval (CLIR) systems [5, 13]. A manual dictionary is used for translation, and a transliteration mining algorithm is used to handle the Out Of Vocabulary (OOV) words which are not present in the dictionary [6]. The Term Frequency Model (TFM) includes the concept of a set of comparable sentences and cosine similarity [7]. The dual semantic space based translation models CL-LSI, CL-LDA are effective but not efficient [8]. A Statistical Machine Translation (SMT) system is trained on aligned parallel and comparable sentences [9]. The transliteration generation or mining techniques are used to handle the OOV words [10–12].

An open source machine translation toolkit *Moses*[2] was developed which is language-independent [14]. The phrasal translation technique enhances the power of MT [15]. 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 [16]. It is highly tedious to develop, train and evaluate the MT systems for Hindi-English language pair. A sentence-aligned parallel corpus *HindiEnCorp*[3] was developed and evaluated for MT system [17]. The recently developed sentence-aligned Hindi-English parallel corpus by IIT Bombay is a superset of the *HindiEnCorp*. The developed parallel corpus experimented for the SMT and NMT system, and they proved that the SMT system performs better than the NMT system for the same set of resources [18].

3 Statistical Machine Translation

An SMT system employs four components, i.e., word translation, phrasal translation, decoding, and language modeling [19].

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

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

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 different iterations, so an Expectation Maximization (EM) algorithm is used to eliminate this issue. It follows an Expectation step where the probabilities of alignments are computed and a Maximization step where the model is estimated from the data. The 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) in which 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.

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

3.2 Phrasal Translation

The phrase model is not limited to only linguistic phrases which can be a noun phrase, verb phrase, prepositional phrase etc. It includes two steps, extraction of phrase pairs and scoring phrase pairs. The phrase pairs are extracted such 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

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

3.3 Decoding

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

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

$$e_{best} = argmax_e \prod_{i=1}^{l} \phi(\bar{h_i}, \bar{e_i})\ d(start_i - end_{i-1} - 1)\ p_{LM}(E) \tag{7}$$

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 target language translation.

3.4 Language Modeling

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

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

$$p(w_1 w_2 ... w_n) = \prod_i p(w_i|w_1 w_2 ... w_{i-1}) \tag{8}$$

4 Proposed Approach

A Translation Induction Algorithm (TIA) is proposed in Figure 1, to compute the appropriate target language translation.

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 closer
          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 closer 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[$w_i$]
Step 3.3:         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
Step 3.4:         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[$w_i$]
Step 4.3:         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
Step 4.4:         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 $w_i$
```

Fig. 1. Translation Induction Algorithm.

Refined Stop-Words (RSW). The stop-words are considered as the noise in Mono-Lingual Information Retrieval (MoLiIR). In CLIR scenario, a source and target language stop word can have multiple meaningful target and source language translations respectively. The stop-words impart a significant role in CLIR process, as they represent meaningful target language translations, such stop-words examples are presented in Table 1. The meaningful stop-words are eliminated from the source and target language stop-words list, such meaningful stop-words are listed in Table 2.

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

Morphological Variants Solutions (MVS). The maximum Longest Common Subsequence Ratio (LCSR) score is used to select the approximate closer word if a source language query word is not found in Parallel Corpus (PC) by exact search. Only LCSR is not sufficient for morphologically rich language. Following MVS solutions are additionally added to trace the approximate closer word of source language query word.

- Equality of nukta character with the corresponding non-nukta character: The 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). The target documents contain many words with nukta and non-nukta characters, so an equality solution is applied where nukta and non-nukta characters are considered equal.
- Auto-correction of user query words: Query words are searched in the parallel corpus, as they appear to it. The correctness of the query words is not

checked. 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 compute the closest word's frequency cwf_i, of the query word with the help of LCSR. If $cwf_i > wf_i$, then query word is replaced by its closest word. The examples of such words are shown in Table 3.

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

- Equality of chandra-bindu with म् *and* न्: A query word with chandra-bindu can be equivalent to many other words, like a word "अंबानी"(Ambani) have similar LCSR score of 0.83 with these three words "अम्बानी"(Ambani), "अंबाजी"(A mbaji), "अल्बानी"(Albani). If Chandra bindu is considered as equivalent to "म्"then the word "अम्बानी"has the maximum LCSR score, and a correct sentence is selected from the PC.
- Auto-selection of the closest query word: An LCSR score is used to select the closest word if a word is exactly not found in the PC. A word can have multiple closest words with the similar LCSR score as shown in Table 4. The Compressed Word Format (CWF) algorithm [21] is used for auto-selection of the closest query words. So far, the researchers used the CWF algorithm for transliteration mining.

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

A set of parallel sentences is selected for each query word w_i from the PC, in a contextual manner such that each sentence contains either all three words of tri-gram and both of the words of bi-gram in any order, with the inclusion of w_i. If the numbers of selected parallel sentences are less than a threshold t, then z numbers of unigram based parallel sentences of minimum length are also included. The context-based selected sentences return appropriate translation.

5 Experiment Results and Discussion

5.1 Dataset and Resources

The FIRE[5] 2010 and 2011 datasets are used to evaluate the CLIR system, while the WMT [6] news 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. Three experimental setups of MT system are tuned and evaluated by using the common dev_set and test_set.

Table 5. The Characteristic of the MT Dataset and Resources

Training_set	**Language Modeling**	**Dev_set**	**Test_set**
HindiEnCorp	HindiEnCorp	WMT Dev_set (520 sentences)	WMT news test_set 2014 (2507 sentences), and Fire 2008,2010,2011, and 2012 query set (each have 50 sentences)
IIT Bombay[7] (1,492,827 sentences)	IIT Bombay		
	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

5.2 Evaluation Metrics

SMT Evaluation. BLEU score 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

$$precision = \frac{correct\ \ translation}{translation\ \ length}$$

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

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

BLEU score is computed for the entire corpus not for a single sentence [19].

$$BLEU = min(i, \frac{output\ -\ length}{reference\ -\ length})(\prod_{i=1}^{4} precision_i)^{\frac{1}{4}} \tag{9}$$

CLIR Evaluation. The 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 10. MAP for a set of queries is the mean of the average precision score of each query. Precision is the fraction of retrieved documents that are relevant to the query. Average precision of query is calculated in Equation 11.

$$Recall = \frac{|\{relevant\ documents\} \bigcap \{retrieved\ documents\}|}{|\{relevant\ documents\}|} \tag{10}$$

$$Average\ Precision = \frac{\sum_{k=1}^{n}(p(k) \times rel(k))}{Number\ of\ relevant\ documents} \tag{11}$$

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.

5.3 Experimental Setup

The SMT systems with three different experimental setups are trained to translate the user queries, which are given 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.
- 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.

All three experimental setups use 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.

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

5.4 Results and Discussions

The SMT experimental setups are evaluated by using the BLEU score. 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.

Table 7. SMT evaluation results

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

SMT_setup2 performs better than the SMT_setup1 for four test case. The performance of SMT_setup2 and SMT_setup3 are approximately similar, SMT_setup2 performs better for 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 the Recall and MAP, which is represented in Table 8.

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

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, but it is not so well-organized and mixes the noise in the translation, hence the translated output performance is poor in perspective of CLIR. The SMT_setup3 uses WMT news corpora 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 perspective of CLIR, the proposed approach outperforms the Hindi-English SMT systems which are based on the best available resources.

6 Conclusion

CLIR system follows an MT technique and a MoLiIR. The source language user queries are translated by using the different SMT setups and the proposed approach. HindiEnCorp is smaller than the parallel corpus developed by IIT Bombay, but it is more well-organized than the IIT Bombay corpus. The SMT_setup1 performance is a little poor in perspective of MT system, while in perspective of CLIR, its performance is better than the other SMT setups. Stop-words impart a significant role in information retrieval. The SMT systems don't deal with the stop-words and the morphological variants. The proposed approach improves the results and outperforms the other SMT systems, as it deals with the stop-words and morphological variants.

References

1. Mustafa A, Tait J, Oakes M. Literature review of cross-language information retrieval. In Transactions on Engineering, Computing and Technology, 2005.
2. Wang A, Li Y, Wang W. Cross language information retrieval based on lda. In International Conference on Intelligent Computing and Intelligent Systems, ICIS 2009. IEEE, vol. 3, p. 485-490. IEEE, 2009.
3. Nasharuddin NA, Abdullah MT. Cross-lingual Information Retrieval State-of-the-Art. In electronic Journal of Computer Science and Information Technology (EJCSIT). Vol. 2, No. 1, p.1-5, 2010.
4. Nagarathinam A, Saraswathi S. State of art: Cross Lingual Information Retrieval System for Indian Languages. In International Journal of computer application. Vol. 35, No. 13, p. 15-21, 2011.
5. Sharma VK, Mittal N. Cross Lingual Information Retrieval (CLIR): Review of Tools, Challenges and Translation Approaches. In Information System Design and Intelligent Ap-plication, p. 699-708, 2016.
6. Sharma VK, Mittal N. Cross Lingual Information Retrieval: A Dictionary Based Query Translation Approach. in Advances in Intelligent Systems and Computing , 2016.
7. Sharma VK, Mittal N. Exploiting Parallel Sentences and Cosine Similarity for Identifying Target Language Translation. Journal of Procedia Computer Science 89, p. 428-433, 2016.
8. Vulić I, De Smet W, Moens MF. Cross-language information retrieval models based on latent topic models trained with document-aligned comparable corpora. Information Retrieval, 16(3), p.331-368.
9. Jagarlamudi J, Kumaran A. Cross-Lingual Information Retrieval System for Indian Languages. In Advances in Multilingual and Multimodal Information Retrieval, Springer Berlin Heidelberg, p. 80-87, 2007.
10. Saravanan K, Udupa R, Kumaran A. Crosslingual information retrieval system enhanced with transliteration generation and mining. In Forum for Information Retrieval Evaluation (FIRE-2010) Workshop, 2010.

11. Surya G, Harsha S, Pingali P, Verma V. Statistical transliteration for cross language information retrieval using HMM alignment model and CRF. In Proceedings of the 2nd Workshop on Cross Lingual Information Access, 2008.
12. Shishtla P, Surya G, Sethuramalingam S, Varma V. A language-independent translit-eration schema using character aligned models at NEWS 2009. In Proceedings of the 2009 Named Entities Workshop: Shared Task on Transliteration, Association for Computational Linguistics, p. 40-43, 2009.
13. Zhou D, Truran M, Brailsford T, Wade V, Ashman H. Translation techniques in cross-language information retrieval. ACM Computing Surveys (CSUR), 45(1), p.1, 2012.
14. Koehn P, Hoang H, Birch A, Callison-Burch C, Federico M, Bertoldi N, Cowan B, Shen W, Moran C, Zens R. Moses: Open source toolkit for Statistical Machine Translation. In Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions, 2007.
15. Green S, Cer D, Manning C. Phrasal: A toolkit for new directions in statistical machine translation. In Proceedings of the Ninth Workshop on Statistical Machine Translation, p. 114-121, 2014.
16. Wu Y, Schuster M, Chen Z, Le QV, Norouzi M, Macherey W, Krikun M, Cao Y, Gao Q, Macherey K, Klingner J. Google's neural machine translation system: Bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144, 2016.
17. Bojar O, Diatka V, Rychlý P, Stranák P, Suchomel V, Tamchyna A, Zeman D. HindEnCorp-Hindi-English and Hindi-only Corpus for Machine Translation. In LREC, p. 3550-3555, 2014.
18. Kunchukuttan A, Mehta P, Bhattacharyya P. The IIT Bombay English-Hindi Parallel Corpus. arXiv preprint arXiv:1710.02855, 2017.
19. Koehn P. Statistical machine translation. Cambridge University Press, 2009.
20. Papineni K, Roukos S, Ward T, Zhu WJ. BLEU: a method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting on association for computational linguistics, Association for Computational Linguistics, p. 311-318, 2002.
21. Janarthanam SC, Sethuramalingam S, Nallasamy U. Named entity transliteration for cross-language information retrieval using compressed word format mapping algorithm. In Proceedings of the 2nd ACM workshop on Improving non english web searching, p. 33-38, ACM 2008.