International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-7, Issue-6, March 2019

Impact of Related Languages as Pivot Language on Machine Translation

Syed Afroz Ahmed, Nisheeth Joshi, Iti Mathur, Pragya Katyayan

Abstract: ***In this paper we have explored a pivot-based approach in development of machine translation system whose parallel corpus is not available. For our study, we have taken Arabic-Hindi as the language pair for development of MT system and Urdu as the pivot language. We have developed 4 MT systems using this approach. These 4 MT systems work on different methodologies. Among them Hierarchical Phrase Based Machine Translation System produced better results.***

Index Terms: ***Machine Translation, Pivot Languages, Statistical Machine Translation, Arabic, Hindi.***

I. INTRODUCTION

Since the dawn of time, language has been the one of the important activities in evolution of mankind. As the communities grew and became aware of other communities, the trading between then started. The traders, who went to far flung areas, were required to sell their product in a language which they did not knew. Thus, emerge the importance of translation. Since the mid of 20th century, computing emerged as a forerunner in technology development. With increase in processing, the computer systems were able to do automatic translation. This was known as machine translation (MT) or Machine Aided Translation (MAT).

Over the years MT has improved many folds, but it is still yet to achieve fully automatic high-quality machine translation status. A lot of approaches have been introduced to implement the MT systems. The popular approaches are:

i. Dictionary Based MT
ii. Transfer Based MT
iii. Example Based MT
iv. Statistical MT
v. Neural MT
vi. Hybrid MT

All these approaches have their pros and cons. This is the reason why having a fully autonomous MT system is a distant dream.

In this paper we have we have tried to address this approach by using a Pivot language in between the source and the target language. For our study we have used Arabic-Hindi language pair. The reason for selecting this pair is due to its lack of availability of parallel corpus. Thus, in order or implement an MT system for this language pair, we need to implement it using an intermediate language. For our study we have used Urdu as the intermediate (pivot) language. In section 2, we have studied the related work done in area of pivot-based machine translation. In section 3, we have explained our proposed methodology. In section 4, we have discussed our evaluation results and in section 5, we have discussed the conclusion.

Revised Manuscript Received on 30 March 2019.
* Correspondence Author
Syed Afroz Ahmed*, Emirates Canadian University College, Umm Al Quwain, United Arab Emirates.
Nisheeth Joshi, Department of Computer Science, Banasthali Vidyapith, Rajasthan, India.
Iti Mathur, Department of Computer Science, Banasthali Vidyapith, Rajasthan, India.
Pragya Katyayan, Department of Computer Science, Banasthali Vidyapith, Rajasthan, India.

II. LITERATURE SURVEY

Tsunakawa et al. [1] described the process of generating bilingual lexicon by using other bilingual lexicons via pivot language. They further described the short comings of their approach. They explained that ambiguity and term mismatches are two major problems that occur in their approach.

De Belder et al [2] proposed a method to reduce the lexical complexity by performing text simplification. Lexical simplification aims to improve the readability of the text by substituting the original word from its simplified one. According to their approach, they have evaluated the source word by estimating probabilities using intersection of two sets which was generated using synonyms and using latent words language model (LWLM). They found that the accuracy of lexical simplification was 11.7% using LWLM.

Bott and Saggion [3] have used unsupervised alignment algorithm for collecting parallel corpus for text simplification. They have used 200 news articles in Spanish language which contain 110 source sentences and they processed them with the help of POS Tagging, NER and Parsing and produced 145 simplified sentences. There were cases when two adjacent simplified sentences correspond to single source sentence. This fact forced them to adopted Hidden Markov Model (HMM) for sequential classification. They evaluated the value of Recall and found it to be 61.1% using TF*IDF score based technique and 80.9% using HMM.

Candido Jr. et al [4] presented a rule based syntactic simplification system which produces text simplification for Brazilian Portuguese. They have used 7 operations which are used to simplify the structure of the sentences. In their paper, the authors followed the 3-phase architecture given by Siddharthan with a slight change in the regression phase. In the evaluation phase, they have used 104 news articles from Zero Hora newspaper. They have prioritized the sequence of

the simplification operations used and evaluated the value of precision, recall and f-measure as 51.64, 65.19 and 57.62 respectively when all the operations are applied. They have also developed a Simplifica system.

Cohn and Lapata [5] outlined a tree-based transduction attempt for text simplification. Their methodology was designed on synchronous tree substitution grammar (STSG). They made algorithm to implement decoding process. Using synchronous grammar, they analyzed their problem as tree to tree simplification. Their work was based on STSG framework and they made model to compress the sentences. Their evaluation was carried out on three compression corpora and they have concluded their work by establishing good results.

Some new stories are structured in complex format which may be difficult to understand for the people who suffered reading disabilities. Glavas and Stajner [6] attempted an event centered simplification-based approach. Their methodology involved extraction of factual events and extraction of argument events using sentence wise simplification and event wise simplification. Their objective was to improve the readability by removing the unnecessary information. They have used 100 news stories from EMM News Brief and they concluded their research by analyzing that event wise simplification produced more accurate results.

Jonnalagadda et al. [7] developed a text simplification process BioSimplify for simplifying biomedical text using Syntactic parsers. They have used BioInfer corpora. Their methodology consists of non-syntactic transformation and syntactic transformation. In non syntactic transformation approach, they have performed removal of unnecessary words, substitution of genes name and substitution of noun phrases. In syntactic transformation, they have reduced the sentence length by splitting the sentence. The value of f-measure using McClosky and Charniak (CM) was 78.86% and the value of absolute error reduction was 2.90% and using Link Grammar (LG), the value of absolute error reduction was evaluated as 4.23%.

Klaper et al. [8] have generated a monolingual parallel corpus using German/Simple German sentences. Statistical Machine Translation system for text simplification needs parallel corpus. They used 7000 sentences from German corpus. They have used monolingual sentence alignment algorithm to develop monolingual corpora. The monolingual sentence alignment algorithm consists of two stages, training and testing. In training phase, the formation of clusters takes place independently for Alltagssprache (AS) text (original text) and Leichte Sprache (LS) text (simplified text) and then mapping between the two sets of clusters are computed. Since, in German the nouns are capitalized so Name Entity Relationship (NER) becomes difficult. To resolve this problem, they need to aligned AS and LS text and applied mapping rules. In testing phase, each paragraph was assigned to its closest cluster using cosine similarity. Later, AS and LS text was combined and with the help of Boostexter they have identified whether the text was mapped or not.

Klebanov et al. [9] presented an approach of Natural language text simplification. They have developed an algorithm that produces easy access sentences (EAS) from which they obtained textual information. Their methodology involved BBN's Identifier to identify the person names and used MINIPAR to derive dependency structures and then perform verb-wise construction of EAS. They have used 123 sentences to evaluate the performance of EAS construction algorithm and found out that 68 sentences passed the EAS requirements (value of precision was 55%).

Klerke and Sogaard [10] presented DSim a Danish monolingual parallel corpus which contain 3701Danish sentences from news telegrams and their simplified sentences. While developing the DSim, the authors mainly focused of lexical simplification and syntactic simplification. Their methodology of sentence alignment involved mapping of each complex sentence with simplified sentence using cosine similarity. For evaluation, they have used TF*IDF weighted cosine similarity score and observed that the value of precision and recall were 90.8% and 84.7% respectively at a threshold of .35. Also, the value of precision and recall were 94.9% and 70.9% respectively at a threshold of 5.

In Indian context, some work has been done in this area. At lexical level, Ameta et al. [11] developed a rule-based stemmer for Gujarati which they used in a Gujarati-Hindi MT system. This used a transliteration scheme that was developed by Joshi et al. [12][13]. Paul et al. [14][15] developed a lemmatizer for Hindi which used a POS Tagger based on statistical learning [16]. Katyayan and Joshi [17] used this tagger for sarcasm detection. Some more POS Taggers were developed using this approach. Singh et al. [18][19][20] developed a POS Tagger for Marathi while Gupta et al. [21] developed a tagger for Urdu. Gupta et al. [22] further used their tagger in development of a stemmer [23] and a lemmatizer [24]. They also used this tagger for development of a multi-word expression system [25][26].

For Punjabi, Bhalla et al. [27] developed a name entity recognition (NER) and translation system. Chopra et al. [28][29] developed NER system for Hindi. Efforts have also been done in the area of parsing and chunking. Asopa et al. [30][31] developed a shallow parser for Hindi. Tyagi et al. [32][33] and Chopra et al. [34] developed techniques performing syntactic transfer for Urdu to Hindi MT. Chopra et al. [35] studied the drawbacks in the MT systems in Indian Languages and developed an MT system [36] which addressed most of the issues raised by them. Singh et al. [37] developed a transfer grammar system for Urdu-Hindi MT using parallel corpus. At semantic level, Kumar et al. [38] and Sharma and Joshi [39] developed mechanisms for word sense disambiguation for Hindi. Chopra at el. [40] reviewed various MT systems developed in India. This helped in developing our experiments in this study. Chopra at el. [41] also developed an approach of developing an MT system. In this approach they used several linguistic resources in development of their SMT system.

III. PROPOSED METHODOLOGY

We have implemented a MT system for Arabic-Hindi language pair. Since, we did not any parallel corpus. We incorporated a pivot-based approach to MT. We have used

Urdu as a pivot language for Arabic-Hindi MT. In this section we shall discuss the requirements of our experiment. Here, we describe the corpus used. Then we shall discuss the MT toolkits used for implementing our MT engines and NLP resources used. Then we shall discuss the MT systems used and the design methodology for them.

A. Corpus Used

For this study, we used two different corpora. The first was a QCRI [42] which had Arabic-Urdu language pair. This corpus had 1,85,000 parallelly aligned sentences, out of which, we used 1,48,000 sentences as our training corpus and 37,000 sentences as our tuning corpus. The second corpus was EILMT corpus [43] which had Urdu-Hindi language pair. This corpus has parallelly aligned sentences from tourism related documents developed under the project, "Development of Urdu to Indian Languages Machine Translation Systems". We used 15,200 sentences parallelly aligned Urdu-Hindi sentences. Out of this, we used 12,160 sentences for training and 3040 sentences as tuning corpus. Table 1 shows the statistics of this corpus.

Language Pair	Corpus	Training	Tuning
Arabic-Urdu	QCRI Corpus	1,48,000	37,000
Urdu-Hindi	EILMT Corpus	12,160	3,040

Table 1: Summary Statistics of Corpus

We choose Urdu as a pivot language because, we believe that Urdu is related to Arabic and at the same time is related to Hindi. This this language becomes our natural choice as a pivot language.

B. MT Toolkits and NLP Resources

We used 3 MT methodologies for this study. The first methodology used was based on statistical machine translation system which was implemented using Moses MT Toolkit [44]. Using this toolkit, we have trained 2 MT systems. The first is baseline system which uses simple phrase-based approach. It is based on the equation 1.

$$e_{best} = argmax_e \prod_{i=1}^{n} \emptyset(f_i|e_i)\, d(start_i - end_{i-1} - 1)\, p_{LM}(e) \quad (1)$$

Here, e is the target language and f is the source language. e_{best} is the best translation in target language which is achieved using the computation of the equation 1. Π multiplies all the available phrases in the translation. $\phi(f_i|e_i)$ is the translation model which identifies all the possible target translations for source phrases. $D(start_i - end_{i-1} - 1)$ is the reordering model which identifies the sequence of words to be translated. $P_{LM}(e)$ is the language model which improves the fluency of the entire text.

The other is a factored model which used linguistic factor for performing machine translation. This is based on figure 1. Here, instead of looking at just words/phrases, we look at its linguistic features as well. The composition of this model is the word, its root word and its corresponding linguistic features with its part of speech category. All these are mapped for both source and target language for each word. In order to implement this model, we were required to have linguistic resources for all three languages.

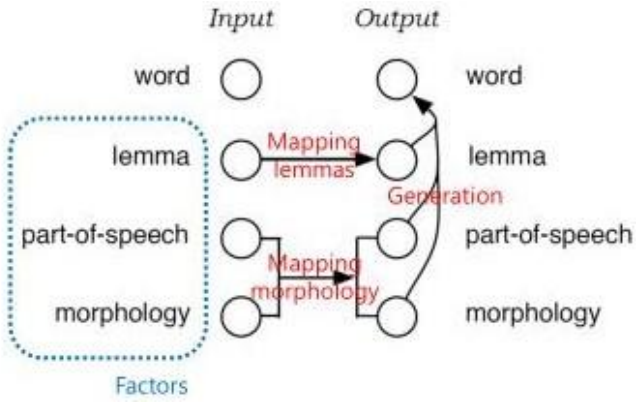

Figure 1: Factored Statistical Machine Translation

For Arabic, we used MADAMIRA [45] which was developed at Colombia University, for all the tasks viz morphological analysis, POS Tagging and chunking. The POS Tagset used for this was Mada POS Tagset which was then converted into Penn Arabic Tagset.

For Urdu, we used the statistical POS tagger used by Gupta et al. [21]. In this the Tagset used, or POS tagging and chunking was IL-POS Tagset. For Hindi, we used a lemmatizer developed by Paul et al. [46], a POS tagger developed by Joshi et al. [47] and a chunker developed by Asopa et al. [48]. The Tagset used for these was IL-POS Tagset.

The second MT methodology that we had used was based on hierarchical MT. for this we had used Joshua MT toolkit [49]. The third MT methodology was based on example-based approach. For this we have used the system developed at Banasthali Vidyapith [50]. Details of the MT systems developed is shown in table 2.

Engine No.	Machine Translation Model	Remarks
E1	Statistical Machine Translation Model	Translation of text from Arabic to Hindi using Urdu as Pivot Language for Moses Phrase-Based Model
E2	Factored (Statistical) Machine Translation Model	Translation of text from Arabic to Hindi using Urdu as Pivot Language for Moses Factored Model using Morphology and POS Tagging
E3	Hierarchical (Statistical) Machine Translation Model	Translation of text from Arabic to Hindi using Hindi as Pivot Language for Joshua Hierarchical Model

Engine No.	Machine Translation Model	Remarks
E4	Example Based Machine Translation Model	Translation of text from Arabic to Hindi using Hindi as Pivot Language for EBMT

Table 2: Machine Translation Systems Used

C. Working of MT Systems

The general working of these systems developed using this approach is shown in figure 2. Here, in all the cases, we took parallel corpus and trained a system. Since Arabic-Hindi corpus was not available. We trained a system on Arabic-Urdu and obtained the translations of Arabic text in Urdu. Next, we trained an Urdu-Hindi MT system which translated the Urdu text obtained from previous study into Hindi. The block diagram of this approach is shown in figure 3.

IV. Evaluation and Discussion

We evaluated our results using BLEU [51] automatic MT evaluation metric and corelated there results with human evaluation [52]. We performed the evaluation at sentence, document and system level. At sentence level, we have calculated the number of times an engine produced best translation. For BLEU MT evaluation metric, E3 scored the best translation. It scored the best results for 422 sentences. Rest all the engines roughly scored the same best rank except for E1 which was able to produce no best translations. Table 3 shows the results of this study.

We did the same study for Human Evaluation and found that the results do match with BLEU. E4 produced the best translations the maximum number of times and E1 was the second best. Table 4 shows these results.

At document level, we divided the 500 sentences into 5 documents of 100 sentences each and calculated the scores of each document. For BLEU, the results are shown in table 5. out of the 5 documents, E3 had the best results in all 5 documents and E4 had best in 2 documents. This is also shown in Figure 4.

MT Engine	No. of Times Scored the Highest
E1	0
E2	29
E3	15
E4	422

Table 3: Highest score of MT Engine for BLEU at Sentence Level

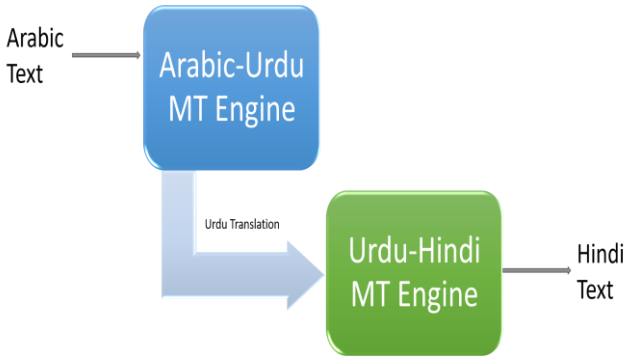

Figure 2: Block Diagram of Arabic-Hindi MT System using Urdu as Pivot Language

MT Engine	No. of Times Scored the Highest
E1	59
E2	17
E3	11
E4	386

Table 4: Highest score of MT Engine for HEval at Sentence Level

	E1	E2	E3	E4
Doc1	0.154196	0.215052	**0.486217**	0.290358
Doc2	0.13208	0.184148	**0.430967**	0.234898
Doc3	0.150576	0.20893	**0.438678**	0.25133
Doc4	0.145637	0.203611	**0.493589**	0.26395
Doc5	0.140675	0.194285	**0.42791**	0.229436

Table5: Evaluation Results of BLEU on Ar-Ur-Hi MT at Document Level

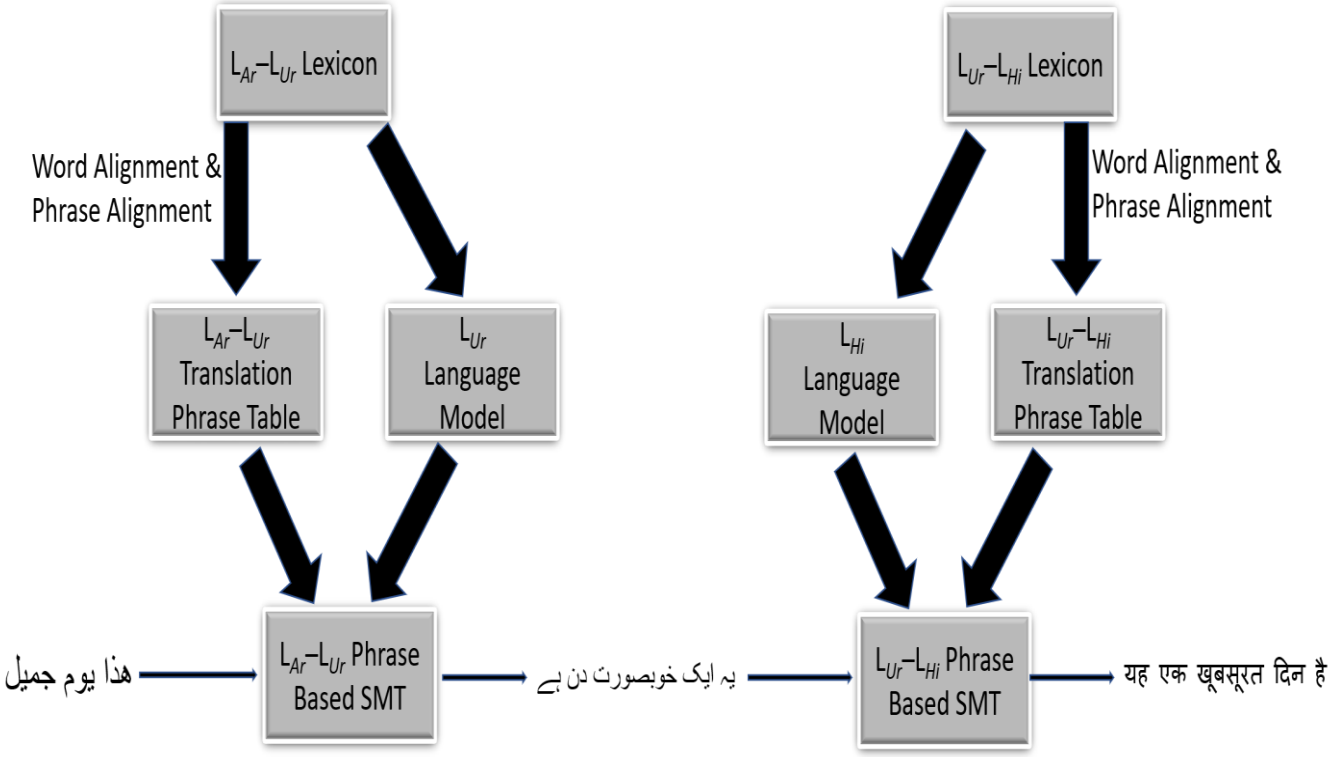

Figure 3: Working of Arabic-Hindi MT System using Urdu as Pivot Language

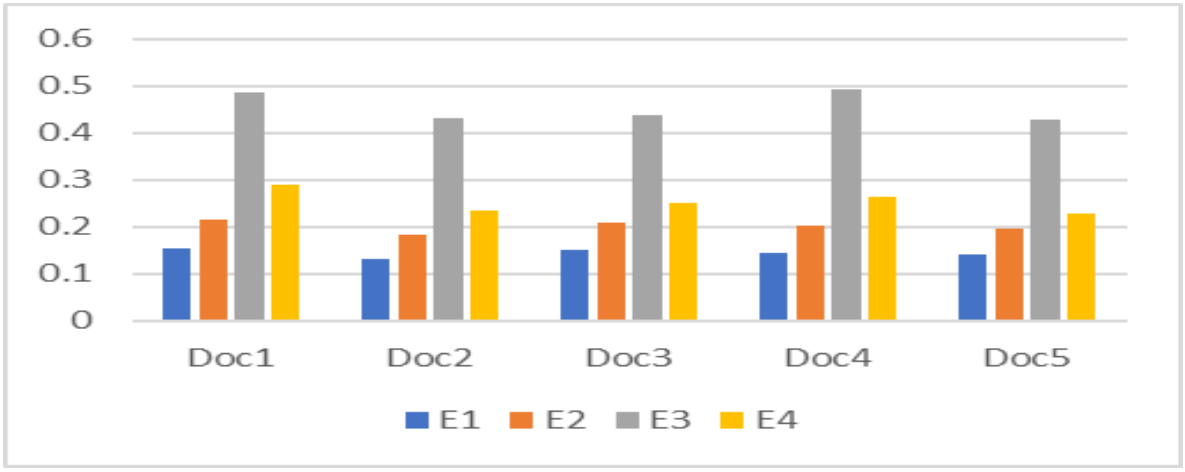

Figure 4: Arabic-Hindi MT Using Urdu as Pivot Language at Document Level for BLEU

For HEval, we did the same and the results of this evaluation is shown in table 6. Among the 5 documents, E3 again scored the best results in all 5 documents. This is also shown in figure 5.

	E1	E2	E3	E4
Doc1	0.212452	0.215052	**0.486217**	0.277207
Doc2	0.162512	0.184148	**0.430967**	0.227781
Doc3	0.100894	0.20893	**0.438678**	0.25133
Doc4	0.205436	0.203611	**0.493589**	0.26395
Doc5	0.205996	0.194285	**0.42791**	0.229436

Table 6: Evaluation Results of HEval on Ar-Ur-Hi MT at Document Level

At system level, for BLEU, E3 scored the best results overall. This was repeated in HEval also where again the best system level score was of E3. The result of this are shown in table 7. Figure 6 shows engine wise scores at system level and figure 7 shows metric wise scores at system level. In both the figures it is clearly shown that the results of all three-evaluation metrics are same for engines E3.

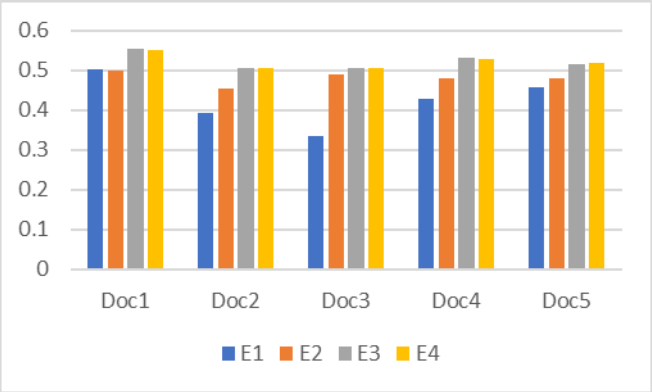

Figure 5: Arabic-Hindi MT Using Urdu as Pivot Language at Document Level for HEval

We performed statistical tests to analyze if the results were actual and were not processed as matter of chance. To establish this, we performed person correlation where we correlated the results of human evaluation with automatic evaluation metrics. This was done to establish that the BLEU evaluation metric was producing the same results as that of human evaluation.

Engine	BLEU	HEval
E1	0.144633	0.177458
E2	0.201205	0.201205
E3	**0.45547**	**0.45547**
E4	0.253995	0.249941

Table 7: Evaluation Results at System Level

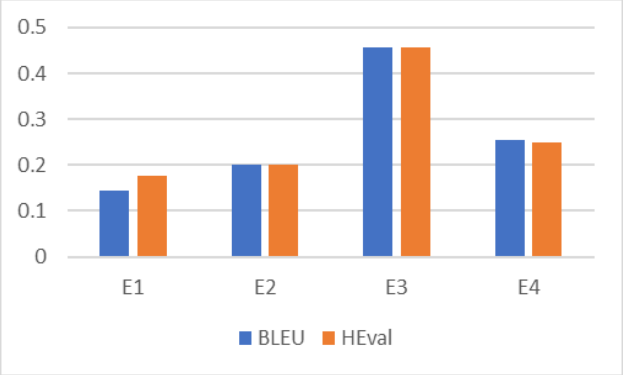

Figure 6: Engine-wise System Level Scores

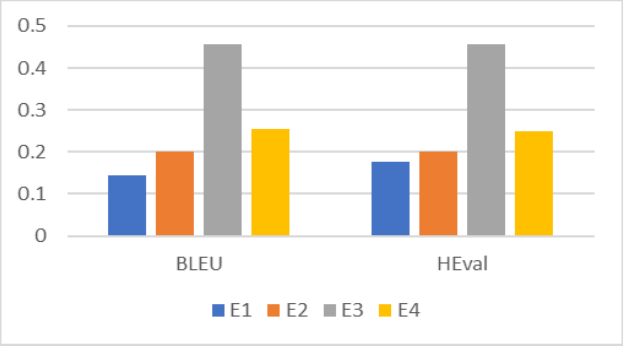

Figure 7: Metric-wise System Level Scores

The results of correlation between HEval and BLEU are shown in table 8. In all the case the results showed positive correlation between the two-evaluation metrics, for all the MT engines. In all the cases, the correlations of BLEU with human evaluation was significant for all engines. By this, we can assume that if we wish to incorporate Arabic-Hindi MT through Urdu as a pivot language then BLEU metric can be used for MT system development.

Engine	Correlation Score Human-BLEU
E1	0.2157
E2	0.2104
E3	0.2147
E4	0.2318

Table 8: Pearson Correlation Between Human and BLEU Evaluation Metrics for all Engines

I. CONCLUSION

In this paper, we shown the development of Arabic-Hindi MT using Urdu as a pivot language. We tested the developed system through 500 sentences across levels viz. sentence, document and system level. For this we did both human and automatic evaluation. In automatic evaluation, we found that, BLEU was producing very good results which were at par with human evaluation. In order to ascertain this, we correlated the results of BLEU with human evaluation, as human evaluation is considered as the benchmark in MT evaluation. The results produced good correlation between the two metrics. This confirms that we can use BLEU as a de-facto metric for development of Arabic-Hindi MT using Urdu as pivot language. Performance wise, engines E3 produced best results at all levels. Thus, we can safely say that this engine can be used for development of MT for this language pair using Urdu as pivot.

REFERENCES

1. Tsunakawa, Takashi, Naoaki Okazaki, and Jun'ichi Tsujii. "Building Bilingual Lexicons using Lexical Translation Probabilities via Pivot Languages." LREC. 2008.
2. De Belder, Jan, and Marie-Francine Moens. "Text simplification for children."Prroceedings of the SIGIR workshop on accessible search systems. 2010.
3. Bott, Stefan, and Horacio Saggion. "An unsupervised alignment algorithm for text simplification corpus construction." Proceedings of the Workshop on Monolingual Text-To-Text Generation. Association for Computational Linguistics, 2011.
4. Candido Jr, A., Maziero, E., Gasperin, C., Pardo, T. A., Specia, L., & Aluisio, S. M. "Supporting the adaptation of texts for poor literacy readers: a text simplification editor for Brazilian Portuguese." Proceedings of the Fourth Workshop on Innovative Use of NLP for Building Educational Applications. Association for Computational Linguistics, 2009.
5. Cohn, Trevor Anthony, and Mirella Lapata. "Sentence compression as tree transduction." arXiv preprint arXiv:1401.5693 (2014).
6. Glavaš, Goran, and Sanja Štajner. "Event-Centered Simplification of News Stories." Proceedings of the Student Workshop held in conjunction with RANLP. 2013.

7. Jonnalagadda, S., Tari, L., Hakenberg, J., Baral, C., & Gonzalez, G. "Towards effective sentence simplification for automatic processing of biomedical text." Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers. Association for Computational Linguistics, 2009.
8. Klaper, David, Sarah Ebling, and Martin Volk. "Building a german/simple german parallel corpus for automatic text simplification." Proc. of the Second Workshop on Predicting and Improving Text Readability for Target Reader Populations. 2013.
9. Klebanov, Beata Beigman, Kevin Knight, and Daniel Marcu. "Text simplification for information-seeking applications." On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE. Springer Berlin Heidelberg, 2004. 735-747.
10. Klerke, Sigrid, and Anders Søgaard. "DSim, a Danish Parallel Corpus for Text Simplification." LREC. 2012.Ameta, J., Joshi, N., & Mathur, I. (2012). "A lightweight stemmer for Gujarati." arXiv preprint arXiv:1210.5486.
11. Ameta, J., Joshi, N., & Mathur, I. (2013). "Improving the quality of Gujarati-Hindi Machine Translation through part-of-speech tagging and stemmer-assisted transliteration." arXiv preprint arXiv:1307.3310.
12. Joshi, N., & Mathur, I. (2012). "Input Scheme for Hindi Using Phonetic Mapping." arXiv preprint arXiv:1209.1300.
13. Joshi, N., Mathur, I., & Mathur, S. (2010). "Frequency-based predictive input system for Hindi." In Proceedings of the International Conference and Workshop on Emerging Trends in Technology (pp. 690-693). ACM.
14. Paul, S., Tandon, M., Joshi, N., & Mathur, I. (2013). "Design of a rule-based Hindi lemmatizer." In Proceedings of Third International Workshop on Artificial Intelligence, Soft Computing and Applications, Chennai, India (pp. 67-74).
15. Paul, S., Joshi, N., & Mathur, I. (2013). "Development of a Hindi lemmatizer." arXiv preprint arXiv:1305.6211.
16. Joshi, N., Darbari, H., & Mathur, I. (2013). "HMM-based POS tagger for Hindi." In Proceeding of 2013 International Conference on Artificial Intelligence, Soft Computing (AISC-2013).
17. Katyayan, P., & Joshi, N. (2019). "Sarcasm Detection Approaches for Urdu Language." In Smart Techniques for a Smarter Planet (pp. 167-183). Springer, Cham.
18. Singh, J., Joshi, N., & Mathur, I. (2013). "Development of Marathi part of speech tagger using statistical approach." In Advances in Computing, Communications and Informatics (ICACCI), 2013 International Conference on (pp. 1554-1559). IEEE.
19. Singh, J., Joshi, N., & Mathur, I. (2013). "Part of speech tagging of Marathi text using trigram method." arXiv preprint arXiv:1307.4299.
20. Singh, J., Joshi, N., & Mathur, I. (2014). "Marathi Parts-of-Speech Tagger Using Supervised Learning." In Intelligent Computing, Networking, and Informatics (pp. 251-257). Springer, New Delhi.
21. Gupta, V., Joshi, N., & Mathur, I. (2016). "POS tagger for Urdu using Stochastic approaches." In Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies (p. 56). ACM.
22. Gupta, V., Joshi, N., & Mathur, I. (2013). "Rule based stemmer in Urdu." In Computer and Communication Technology (ICCCT), 2013 4th International Conference on (pp. 129-132). IEEE.
23. Gupta, V., Joshi, N., & Mathur, I. (2015). "Design & development of rule based inflectional and derivational Urdu stemmer 'Usal'." In Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE), 2015 International Conference on (pp. 7-12). IEEE.
24. Gupta, V., Joshi, N., & Mathur, I. (2016). "Design and development of a rule-based Urdu lemmatizer." In Proceedings of International Conference on ICT for Sustainable Development (pp. 161-169). Springer, Singapore.
25. Gupta, V., Joshi, N., & Mathur, I. (2017). "Approach for multiword expression recognition & annotation in urdu corpora." In Image Information Processing (ICIIP), 2017 Fourth International Conference on (pp. 1-6). IEEE.
26. Gupta, V., Joshi, N., & Mathur, I. (2019). "Advanced Machine Learning Techniques in Natural Language Processing for Indian Languages." In Smart Techniques for a Smarter Planet (pp. 117-144). Springer, Cham.
27. Bhalla, D., Joshi, N., & Mathur, I. (2013). "Improving the quality of MT output using novel name entity translation scheme." In Advances in Computing, Communications and Informatics (ICACCI), 2013 International Conference on (pp. 1548-1553). IEEE.
28. Chopra, D., Joshi, N., & Mathur, I. (2016). "Named Entity Recognition in Hindi Using Conditional Random Fields." In Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies (p. 106). ACM.
29. Chopra, D., Joshi, N., & Mathur, I. (2016). "Named Entity Recognition in Hindi Using Hidden Markov Model." In Computational Intelligence & Communication Technology (CICT), 2016 Second International Conference on (pp. 581-586). IEEE.
30. Asopa, S., Asopa, P., Mathur, I., & Joshi, N. (2016). "Rule based chunker for Hindi." In Contemporary Computing and Informatics (IC3I), 2016 2nd International Conference on (pp. 442-445). IEEE.
31. Asopa, S., Asopa, P., Mathur, I., & Joshi, N. (2019). "A Shallow Parsing Model for Hindi Using Conditional Random Field." In International Conference on Innovative Computing and Communications (pp. 295-302). Springer, Singapore.
32. Tyagi, S., Chopra, D., Mathur, I., & Joshi, N. (2015). "Classifier based text simplification for improved machine translation." In Computer Engineering and Applications (ICACEA), 2015 International Conference on Advances in (pp. 46-50). IEEE.
33. Tyagi, S., Chopra, D., Mathur, I., & Joshi, N. (2015). "Comparison of classifier-based approach with baseline approach for Urdu-Hindi text simplification." In Computing, Communication & Automation (ICCCA), 2015 International Conference on (pp. 290-293). IEEE.
34. Chopra, D., Joshi, N., & Mathur, I. (2016). "Improving Quality of Machine Translation Using Text Rewriting." In Computational Intelligence & Communication Technology (CICT), 2016 Second International Conference on (pp. 22-27). IEEE.
35. Chopra, D., Joshi, N., & Mathur, I. (2018). "A Review on Machine Translation in Indian Languages." Engineering, Technology & Applied Science Research, 8(5), 3475-3478.
36. Chopra, D., Joshi, N., & Mathur, I. (2018). "Improving Translation Quality By Using Ensemble Approach." Engineering, Technology & Applied Science Research, 8(6), 3512-3514.
37. Singh, S. P., Kumar, A., Darbari, H., Singh, L., Joshi, N., Gupta, P., & Singh, S. (2017, March). Intelligent System for Automatic Transfer Grammar Creation Using Parallel Corpus. In International Conference on Information and Communication Technology for Intelligent Systems (pp. 519-528). Springer, Cham.
38. Kumar, A., Mathur, I., Darbari, H., Purohit, G. N., & Joshi, N. (2016). "Implications of Supervised Learning on Word Sense Disambiguation for Hindi." In Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies (p. 54). ACM.
39. Sharma, P., & Joshi, N. (2019), "Design and Development of a Knowledge Based Approach for Word Sense Disambiguation by Using WordNet for Hindi", International Journal of Innovative Technology and Exploring Engineering, pp 73-78, Vol 8(3).
40. Chopra, D., Joshi, N., & Mathur, I. (2016). "Improving Quality of Machine Translation Using Text Rewriting." In Computational Intelligence & Communication Technology (CICT), 2016 Second International Conference on (pp. 22-27). IEEE.
41. Chopra, D., Joshi, N., & Mathur, I. (2018). "A Review on Machine Translation in Indian Languages." Engineering, Technology & Applied Science Research, 8(5), 3475-3478.
42. Abdelali, A., Guzman, F., Sajjad, H., & Vogel, S. 2014. "The AMARA Corpus: Building Parallel Language Resources for the Educational Domain". In LREC Vol. 14.
43. Lata, S. and Somnath, C. V. K. (2010). Development of Linguistic Resources and Tools for Providing Multilingual Solutions in Indian Languages - A Report on National Initiative, In Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10), Valletta, Malta.
44. Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R. and Dyer, C. 2007. "Moses: Open source toolkit for statistical machine translation." In Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions (pp. 177-180). Association for Computational Linguistics.
45. Arfath Pasha, Mohamed Al-Badrashiny, Mona Diab, Ahmed El Kholy, Ramy Eskander, Nizar Habash, Manoj Pooleery, Owen Rambow, and Ryan M. Roth. 2016. "MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic." LREC. Vol. 14.
46. Paul, S., Tandon, M., Joshi, N. & Mathur, I. 2013. "Design of a rule based Hindi lemmatizer." Proceedings of Third International Workshop on Artificial Intelligence, Soft Computing and Applications, Chennai, India.

47. Joshi, N., Darbari, H. & Mathur, I. 2013. "HMM based POS tagger for Hindi." Proceeding of 2013 International Conference on Artificial Intelligence, Soft Computing (AISC-2013).
48. Asopa, S., Asopa, P., Mathur, I., & Joshi, N. 2016. "Rule based chunker for Hindi." Contemporary Computing and Informatics (IC3I), 2016 2nd International Conference on. IEEE.
49. Li, Z., Callison-Burch, C., Dyer, C., Ganitkevitch, J., Khudanpur, S., Schwartz, L., Thornton, W.N., Weese, J. and Zaidan, O.F., 2009. "Joshua: An open source toolkit for parsing-based machine translation." In Proceedings of the Fourth Workshop on Statistical Machine Translation (pp. 135-139). Association for Computational Linguistics.
50. Joshi, N., Mathur, I. and Mathur, S., 2011, February. "Translation memory for indian languages: an aid for human translators." In Proceedings of the International Conference & Workshop on Emerging Trends in Technology (pp. 711-714). ACM.
51. Papineni K., Roukos S., Ward T., & Zhu W.-J. 2001. Bleu: a method for automatic evaluation of machine translation, RC22176 Technical Report, IBM T.J. Watson Research Center.
52. Joshi, N. Mathur, M. Darbari, H. Kumar, A. 2013, HEval: Yet Another Human Evaluation Metric, International Journal of Natural Language Computing, pp 21-36, Vol 2(5).

AUTHORS PROFILE

Syed Afroz Ahmad is a Lecturer at Emirates Canadian University College, Umm Al Quawain, UAE. Her has over 18 years of teaching experience. His research area is Machine Translaion, Natural Languaeg Processing and Internet of Things.

Nisheeth Joshi is an Associate Professor at Bnasthali Vidyapith, India. He primarily works in Machine Translation, Information Retrieval, Cognitive Computing. He has over 12 years of teaching experimence.

Iti Mathur is an Associate Professor at Banasthali Vidyapith, India. She primarily works in Information Retrieval, Ontology Engnineering, Machine Translation. She has over 15 years experience in teching and research.

Pragya Katyayan is a research scholar at Banasthali Vidyapith. Before joining full-time PhD programme, she has worked as consultatnt on various project based on Natural Langauge Processing. Her areas of Inrerest are Machine Translation, Natural Langauge Processing, Information Retrieval and Deep Learning.