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 one of the important activities in evolution of mankind. As the communities grew and became aware of other communities, the trading between them started. The traders, who went to far flung areas, were required to sell their product in a language which they did not know. 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 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.

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%.

Klapar 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 Allfogssprache (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]. Katayyan 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 parallely 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 parallely aligned sentences from tourism related documents developed under the project, “Development of Urdu to Indian Languages Machine Translation Systems”. We used 15,200 sentences parallely 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.

<table>
<thead>
<tr>
<th>Language Pair</th>
<th>Corpus</th>
<th>Training</th>
<th>Tuning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arabic-Urdu</td>
<td>QCRI Corpus</td>
<td>1,48,000</td>
<td>37,000</td>
</tr>
<tr>
<td>Urdu-Hindi</td>
<td>EILMT Corpus</td>
<td>12,160</td>
<td>3,040</td>
</tr>
</tbody>
</table>

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.

$$e_{best} = \arg\max_{e \in E} \prod_{1 \leq i \leq \overline{i}} \Theta(f_i|e_i) \cdot D(\text{start}_i, \text{end}_{i-1} - 1) \cdot P_L(e)$$  \hspace{1cm} (1)

Here, e is the target language and f is the source language. e\text{best} is the best translation in target language which is achieved using the computation of the equation 1. \(D\) multiplies all the available phrases in the translation. \(\Theta(f_i|e_i)\) is the translation model which identifies all the possible target translations for source phrases. \(D(\text{start}_i, \text{end}_{i-1} - 1)\) is the reordering model which identifies the sequence of words to be translated. \(P_L(e)\) is the language model which improves the fluency of the entire text.

The other is a factored model which used linguistic factors 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](image)

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.

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

Table 2: MT Systems Developed for this Study

Remarks
Impact of Related Languages as Pivot Language on Machine Translation

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 correlated the 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.
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.

<table>
<thead>
<tr>
<th></th>
<th>E1</th>
<th>E2</th>
<th>E3</th>
<th>E4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Doc1</td>
<td>0.212452</td>
<td>0.215052</td>
<td><strong>0.486217</strong></td>
<td>0.277207</td>
</tr>
<tr>
<td>Doc2</td>
<td>0.162512</td>
<td>0.184148</td>
<td><strong>0.430967</strong></td>
<td>0.227781</td>
</tr>
<tr>
<td>Doc3</td>
<td>0.100894</td>
<td>0.20893</td>
<td><strong>0.438678</strong></td>
<td>0.25133</td>
</tr>
<tr>
<td>Doc4</td>
<td>0.205436</td>
<td>0.203611</td>
<td><strong>0.493589</strong></td>
<td>0.26395</td>
</tr>
<tr>
<td>Doc5</td>
<td>0.205996</td>
<td>0.194285</td>
<td><strong>0.42791</strong></td>
<td>0.229436</td>
</tr>
</tbody>
</table>

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.

Table 7: Evaluation Results at System Level

<table>
<thead>
<tr>
<th>Engine</th>
<th>BLEU</th>
<th>HEval</th>
</tr>
</thead>
<tbody>
<tr>
<td>E1</td>
<td>0.144633</td>
<td>0.177458</td>
</tr>
<tr>
<td>E2</td>
<td>0.201205</td>
<td>0.201205</td>
</tr>
<tr>
<td>E3</td>
<td>0.45547</td>
<td>0.45547</td>
</tr>
<tr>
<td>E4</td>
<td>0.253995</td>
<td>0.249941</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Engine</th>
<th>Correlation Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>E1</td>
<td>0.2157</td>
</tr>
<tr>
<td>E2</td>
<td>0.2104</td>
</tr>
<tr>
<td>E3</td>
<td>0.2147</td>
</tr>
<tr>
<td>E4</td>
<td>0.2318</td>
</tr>
</tbody>
</table>

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.

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

Impact of Related Languages as Pivot Language on Machine Translation


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 Translation, Natural Language Processing and Internet of Things.

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

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

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