Hindi to English Machine Translation: Using Effective Selection in Multi-Model SMT

Kunal Sachdeva, Rishabh Srivastava, Sambhav Jain, Dipti Misra Sharma
Language Technologies Research Center,
International Institute of Information Technology, Hyderabad
{kunal.sachdeva, rishabh.srivastava, sambhav.jain}@research.iiit.ac.in, dipti@iiit.ac.in

Abstract

Recent studies in machine translation support the fact that multi-model systems perform better than the individual models. In this paper, we describe a Hindi to English statistical machine translation system and improve over the baseline using multiple translation models. We have considered phrase based as well as hierarchical models and enhanced over both these baselines using a regression model. The system is trained over textual as well as syntactic features extracted from source and target of the aforementioned translations. Our system shows significant improvement over the baseline systems for both automatic as well as human evaluations. The proposed methodology is quite generic and can easily be extended to other language pairs as well.

Keywords: Machine Translation, Multi-Model, Hindi-English

1. Introduction

English and Hindi are respectively reported to be the 3^{rd} and 4^{th} largest spoken languages in the world[1]. The fact that one of these language is known by 10%(approx.) of the world population, makes it an ideal pair for *translation studies*. In the past, nearly a dozen systems are proposed for English to Hindi Machine Translation (Chaudhury et al., 2010; Sinha and Jain, 2003), while for Hindi to English, there are only two known systems viz. Anubharati and Hinglish (Rao, 2001; Dwivedi and Sukhadeve, 2010; Antony, 2013).

In this paper we present our efforts towards building a Hindi to English machine translation system. The choice is driven by the fact that there is a dearth of functional Hindi to English MT systems despite the vast need and utility of automatic translation tools for the same. Our approach uses statistical machine translation (SMT) where we use multiple strategies for translation and choose one translation as the final output. The selection is done using prediction strategies which analyze the candidate translations and predict the best suited translation.

We initially explored Phrase Based, Hierarchical and Factored models for SMT. For our experiments we considered only phrase based and hierarchical systems as factored model, taking POS as factor, gave unsatisfactory(6.3 BLEU score)(Papineni et al., 2002) results compared to the former two systems(>21 BLEU score). The proposed method shows an increase of 0.64 BLEU score and high agreement with human evaluation. However, each technique has its own pros and cons, as certain sentence can get translated better using a technique as compared to other.

The aforementioned prediction methodology dynamically selects the best translation based on the presence or absence of certain features in the translations. The insight is taken from the recent research focus on the quality estimation of machine translation (Bojar et al., 2013). We study and pick features that bear a high correlation with a better translation. The features aim to effectively determine better translations from the candidates.

We implemented the methodology by training a regression model on these features with the evaluation metric measure as the corresponding regression value. The regression model thus acquired is later utilized for guided selection of the translations from multiple models. The one having higher regression value i.e. correctness score, was taken to be the output of the system.

The rest of the paper is organised as follows. We first present the related work on quality estimation and combining models in MT (Section 2). In section 3, we describe our approach of training the baseline MT systems. In section 4, we propose our translation selection approach followed by feature selection. In section 5, we present the experiment setup along with results. In section 6, we evaluate our system against existing commercial Hindi-English MT systems along with human evaluation and conclude the paper with some future directions in section 7.

2. Related Work

The only two reported Hindi-English MT systems are *Anubharati* (Sinha, 2004) and *Hinglish* (Sinha and Thakur, 2005). The former uses a combination of example-based, corpus based and elementary grammatical analysis while latter is an extension of the former. Other general translation systems like Google and Bing Translate have support for Hindi-English translation.

The SMT research community has shown interest towards the quality prediction of the translations, including two shared tasks organized in last two years in WMT'12 (Callison-Burch et al., 2012) and WMT'13 (Bojar et al., 2013) (Workshop on Statistical Machine Translation). The

[1] http://en.wikipedia.org/wiki/List_of_languages_by_number_of_native_speakers

generic idea among the participants has been to model certain rich features to judge translation quality. Hardmeier et al.(2012) utilized nearly 99 features to train a regression model using 'tree kernels' for quality estimation. Avramidis(2012) also modeled the problem as regression, as well as classification problem and deduced that the later doesn't perform well on unseen data. Though the chosen features truly governs the success of such a system, but previous studies have shown an inclination towards choosing regression modeling over classification.

In other work by (Hildebrand and Vogel, 2008) 23 features from N-best list of several MT systems was considered to improve the translation quality. They have shown a consistent increase in BLEU score by combining all systems progressively. Other significant works using multiple systems have been reported in system combination task of WMT'11 (Callison-Burch et al., 2011)

3. Translation Models

3.1. Corpus and data division

We have used the ILCI corpora (Jha, 2010), which contains parallel sentences for 11 languages (including Hindi and English) from the health domain with Hindi as their source language. We have used 25000 parallel sentences (23951 after cleaning) for our experiments. The sentences are encoded in utf-8 format. We converted the Hindi sentences to wx[2] notation for easy tokenization. The corpus is split into training (75%, 18319 sentences), development(15%, 3235 sentences) and testing sets (10%, 2397 sentences). From the test set we randomly select 100 sentences for a separate human evaluation.

Division	No.of Sentences
Training	23951
Development	3235
Testing	2397
Test-Human Evaluation	100

Table 1: Data Division and Corpus Statistics

3.2. Training Translation Models

We trained two Hindi to English translation models, phrase-based (Koehn et al., 2003) and and hierarchical (Chiang, 2005), using Moses (Koehn et al., 2007). In phrased based modeling, both source and target sentences are divided into separate phrases while in hierarchical modeling the phrases can be recursive as well. Giza++ (Och and Ney, 2000) is used for phrase alignments and SRILM (Stolcke and others, 2002) with Kneser-ney smoothing (Kneser and Ney, 1995) is used for training the language model (LM) of order 5. Mert (Och, 2003; Bertoldi et al., 2009) is used for minimum error-rate training i.e. to tune the model using the development data-set. Top 1 result is considered as the default output. We obtained a BLEU score of **21.18** and **21.10** for phrase-based and hierarchical models respectively over the test set. For further experiments we have used these systems as our baseline system.

[2]http://sanskrit.inria.fr/DATA/wx.html

4. Translation Selection

The two translations obtained from the phrase based and hierarchical systems are then analyzed for their translation quality. The procedure involves calculating the feature vectors from the obtained translations and feeding them to a pre-trained regression model. The candidate translation giving higher regression value is selected as the final output.

Next we present the methodology for training the regression model, mentioned earlier. The training data for this task is same as the development set used for tuning the translation models. The target value, corresponding to a feature vector, is calculated using MT evaluation metric scores (BLEU, METEOR or NIST) over the system output with reference data. Each data entry corresponds to a sentence's translation and the value is the estimation of its quality.

4.1. Features

Among studied features, the following are employed for regression modeling.

- **Token count:** Number of tokens in source and target sentence and their ratio.
- **Language Modeling:** From the LM of source and target language, log-LM score and perplexity value (computed using SRILM).
- **Part of speech (POS) language modeling:** From Source and target POS LM, log-LM score and perplexity value (computed using SRILM).
- **Out of vocabulary(OOV) words:** Number of OOV words in translated output.

Apart from these syntactic and textual features, a linguistically motivated feature has also been included:

- **Entropy of Parse tree:** We have considered entropy of label, attachment and joint entropy of label+attachment. This score corresponds to the correctness of the parse tree, detailed down to each edge of the parse.

4.1.1. Parse tree confusion score

For calculating entropy of parse tree, we use an augmented version of MaltParser(Nivre et al., 2007), built in-house, to dynamically compute a confusion score for dependency arcs, in typed dependency parsing framework. This is based on the methods proposed by (Jain and Agrawal, 2013), where quantification of confusion is done by calculating entropy with class membership probabilities of the parser actions. The augmented version predicts confusion score according to different types of training behaviors. Maltparser provides three different ways of predicting the output and thus accordingly augmented version predicts confusion score namely, separately for arc-labels and arc

System	Algorithm	Feature Set	Hierarchical		Phrase		Evaluation		
			MSE	SCC	MSE	SCC	BLEU	NIST	Meteor
-	RBF	#6	0.0679	0.0144	0.0637	0.0143	21.44	6.68	56.39
$sys3$	Linear	#6	0.0664	0.0231	0.0628	0.0128	20.66	6.43	56.20
-	Polynomial	#6	0.0672	0.0142	0.0642	0.0059	20.90	6.47	56.08
-	RBF	#8	0.0620	0.1088	0.0585	0.1131	21.14	6.47	56.24
-	Linear	#8	0.0652	0.0698	0.0605	0.0760	20.97	6.45	56.29
-	Polynomial	#8	0.0687	0.0719	0.0651	0.0619	21.14	6.46	56.14
-	RBF	#11	0.0557	0.1814	0.0527	0.1771	21.67	6.55	56.54
$sys2$	Linear	#11	0.0603	0.1362	0.0558	0.1273	**21.82**	6.57	**56.73**
-	Polynomial	#11	0.0643	0.1004	0.0587	0.0959	21.37	6.52	56.39
$sys1$	RBF	#3	0.0649	0.0622	0.0608	0.0599	*22.21*	*6.71*	56.54

Table 2: Regression results. Mean Squared Error(MSE), Squared correlation coefficient(SCC)

formations and a joint model for predicting arc-labels and formations simultaneously.

4.2. Estimation Using Regression

Preprocessing The data is normalized by scaling the values between $[0, 1]$. However, with simple *min-max* scaling, the system was observed to perform clumsily due to presence of outlier values. To overcome this, we utilized interquartile-range[3] to first map the outliers to the min-max bounds.

The aforementioned regression model is built with support vector regression (SVR) using LIBSVM toolkit (Chang and Lin, 2011). Tuned parameters are attained with *gridregression*[4] script. The cost/margin trade-off, the epsilon in loss function and the kernel type are set to optimized values and all other parameters are left at default.

5. Experiments and Results

The results of some of our experiments using BLEU as target regression value are reported in Table 2. We experimented with radial basis function (RBF), polynomial and linear kernels using different feature sets. Using RBF kernel with feature set #6 give better results than the baseline phrase and hierarchical models. After adding LM feature i.e. feature set #8, a slight improvement in the BLEU scores is observed, however, contradictory results are found for the RBF kernel. Combining the complete feature set with linear kernel yield best results in terms of BLEU score of **21.82** indicating an increase of **0.64** from the baseline systems.

We also study the effect of each feature by creating an independent regression model for it. The LM feature gives the best BLEU score of **22.21**, indicating an increase in BLEU score upon the baseline systems. The reason for improvement is the high correlation between the observed (LM score) and predicted value (BLEU score). But this system does not go along well with human evaluation as discussed later.

[3]www.en.wikipedia.org/wiki/Interquartile_range

[4]www.csie.ntu.edu.tw/ cjlin/ libsvmtools/gridsvr /gridregression.py

6. Evaluation

6.1. Human Evaluation

Improvements in BLEU score does not ensure a better MT system (Zhang et al., 2004). To ascertain that, this multi-model system actually gives better translations than the baseline systems, we conducted a separate manual evaluation over 100 sentences selected randomly from the test set. Five human evaluators[5] are provided with source Hindi sentence and output of phrase and hierarchical systems. They are asked to select the better translation among the two candidate translations from our phrase based and hierarchical models. The better translation out of the two is decided by "Max-Wins" voting strategy.

Out of those 100 translations, 63 are marked better for the phrase based and rest for the hierarchical system. Selecting only the translations of phrase based system maximizes the agreement with human evaluators, however overall quality of translation of document is reduced as 37 sentences are selected from the hierarchical system according to human judgment.

Table 3 presents the automatic evaluation scores (BLEU, METEOR and NIST) and percentage agreement with human judgment, for three best performing systems. Here $sys1$ is the model generated by considering only LM as the feature using 'RBF kernel', $sys2$ considers all the aforementioned features using 'linear kernel' and $sys3$ is created using feature set #6 (refer table 4) using 'linear kernel'. $sys2$ gives higher BLEU score than baselines and the highest agreement with the human evaluation (61 out of 100 sentences).

Although using only LM feature ($sys1$) shows a slight improvement in automatic evaluation due to high correlation between LM score and BLEU score, this system does not show an accordance with human evaluation (47 out of 100 sentences). This correlation coefficient is high as BLEU score computes n-grams to evaluate a translation. The evaluation scores for $sys2$ are high and show high agreement with human judgment. Though the automatic evaluation

[5]Hindi as their mother tongue and proficient in English

System	BLEU	METEOR	NIST	Agreement(%)
Phrase	21.18	56.53	6.61	-
Hierarchical	21.10	56.00	6.52	-
$sys1$	22.21	56.54	6.71	47
$sys2$	21.82	56.73	6.57	**61**
$sys3$	20.66	56.20	6.43	59

Table 3: Evaluation scores and agreement with human evaluation of various translation systems.

Feature Set	Features
#3	log-lm score
#6	Entropy of label, attachment and joint entropy of label+attachment, token count of source and target language, ratio of token count of target language to source
#8	*feature set #6* + log-lm score and perplexity value
#11	*feature set #8* + POS log-lm score and perplexity and count of OOV words

Table 4: Description of Feature Sets

scores obtained for $sys3$ are low, yet this system also shows a high agreement with human judgment.

6.2. Comparison with Google and Bing Translate

We also compared our system with the Google and Bing translate. We tested the output of these two systems on our test sentences. The BLEU score of Google and Bing translations turn out to be 14.75 and 15.10 respectively. Translations from our system ($sys2$) are observed to be way better than these systems for the test set. However this could be due to the difference in domain of training corpus.

7. Conclusions and Future Work

In this paper we introduced an approach to estimate the quality of machine translation and dynamically select the better translation at run-time. Combining the text analysis and linguistic features, results in a system which shows improvement over the baseline system and shows high agreement with human judgment.

Sequentially running both the phrase and hierarchical system may result in increase in time of computation as parse tree and other feature computation add to decoding time. To overcome this issue we have employed distributed computing so as to compute all features in parallel for phrase and hierarchical systems.

In future, we plan to intergrate a few more linguistic and other statistical features, extracted at the decoding stage, which can be considered to improve the selection criteria. Prediction of the quality score using active learning is an interesting area to be looked into.

8. References

Antony, P. (2013). Machine translation approaches and survey for indian languages. 18(1):47–78.

Avramidis, E. (2012). Quality estimation for machine translation output using linguistic analysis and decoding features. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 84–90. Association for Computational Linguistics.

Bertoldi, N., Haddow, B., and Fouet, J.-B. (2009). Improved minimum error rate training in moses. *The Prague Bulletin of Mathematical Linguistics*, 91(1):7–16.

Bojar, O., Buck, C., Callison-Burch, C., Federmann, C., Haddow, B., Koehn, P., Monz, C., Post, M., Soricut, R., and Specia, L. (2013). Findings of the 2013 workshop on statistical machine translation. In *8th Workshop on Statistical Machine Translation*.

Callison-Burch, C., Koehn, P., Monz, C., and Zaidan, O. (2011). Findings of the 2011 workshop on statistical machine translation. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 22–64, Edinburgh, Scotland, July. Association for Computational Linguistics.

Callison-Burch, C., Koehn, P., Monz, C., Post, M., Soricut, R., and Specia, L. (2012). Findings of the 2012 workshop on statistical machine translation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 10–51, Montréal, Canada, June. Association for Computational Linguistics.

Chang, C.-C. and Lin, C.-J. (2011). Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27.

Chaudhury, S., Rao, A., and Sharma, D. M. (2010). Anusaaraka: An expert system based machine translation system. In *Natural Language Processing and Knowledge Engineering (NLP-KE), 2010 International Conference on*, pages 1–6. IEEE.

Chiang, D. (2005). A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 263–270. Association for Computational Linguistics.

Dwivedi, S. K. and Sukhadeve, P. P. (2010). Machine translation system in indian perspectives. *Journal of Computer Science*, 6(10):1111.

Hardmeier, C., Nivre, J., and Tiedemann, J. (2012). Tree kernels for machine translation quality estimation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 109–113. Association for Computational Linguistics.

Hildebrand, A. S. and Vogel, S. (2008). Combination of machine translation systems via hypothesis selection from combined n-best lists. In *MT at work: Proceedings of the Eighth Conference of the Association for Machine Translation in the Americas*, pages 254–261. Citeseer.

Jain, S. and Agrawal, B. (2013). A dynamic confusion score for dependency arc labels. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 1237–1242.

Jha, G. N. (2010). The tdil program and the indian language corpora initiative (ilci). In *Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC 2010). European Language Resources Association (ELRA)*.

Kneser, R. and Ney, H. (1995). Improved backing-off for m-gram language modeling. In *Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on*, volume 1, pages 181–184. IEEE.

Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 48–54. Association for Computational Linguistics.

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. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, pages 177–180. Association for Computational Linguistics.

Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryigit, G., Kübler, S., Marinov, S., and Marsi, E. (2007). Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135.

Och, F. J. and Ney, H. (2000). Improved statistical alignment models. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, pages 440–447. Association for Computational Linguistics.

Och, F. J. (2003). Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 160–167. Association for Computational Linguistics.

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

Rao, D. (2001). Machine translation in india: A brief survey. *National Centre for Software Technology, Mumbai*.

Sinha, R. and Jain, A. (2003). Anglahindi: an english to hindi machine-aided translation system. *MT Summit IX, New Orleans, USA*, pages 494–497.

Sinha, R. M. K. and Thakur, A. (2005). Machine translation of bi-lingual hindi-english (hinglish) text. *10th Machine Translation summit (MT Summit X), Phuket, Thailand*, pages 149–156.

Sinha, R. (2004). An engineering perspective of machine translation: Anglabharti-ii and anubharti-ii architectures. In *Proceedings of International Symposium on Machine Translation, NLP and Translation Support System (iSTRANS-2004)*, pages 10–17.

Stolcke, A. et al. (2002). Srilm-an extensible language modeling toolkit. In *INTERSPEECH*.

Zhang, Y., Vogel, S., and Waibel, A. (2004). Interpreting bleu/nist scores: How much improvement do we need to have a better system? In *LREC*.