PanchBhoota: Hierarchical Phrase Based Machine Translation Systems for Five Indian Languages

Neha R Prabhugaonkar
DCST, Goa University
nehapgaonkar.1920@gmail.com

Apurva S Nagvenkar
DCST, Goa University
apurv.nagvenkar@gmail.com

Diptesh Kanojia
CSE, IIT Bombay
dipteshkanojia@gmail.com

Jyoti Pawar
DCST, Goa University
jyotidpawar@gmail.com

Pushpak Bhattacharyya
CSE, IIT Bombay
pb@cse.iitb.ac.in

Manish Shrivastava
CSE, IIT Bombay
mani.shrivastava@gmail.com

Abstract

We present our work on developing fifteen Hierarchical Phrase Based Statistical Machine Translation (HPBSMT) systems for five Indian language pairs namely Bengali-Hindi, English-Hindi, Marathi-Hindi, Tamil-Hindi, and Telugu-Hindi, in three domains each, HEALTH, TOURISM and GENERAL. We named them *PanchBhoota*, as these systems are elemental in nature. We used a very simple approach to train, tune, and test them using cdec toolkit. We hope that this work will motivate Indian Language Machine Translation researchers to look deeper into the field of HPBSMT which is known to perform better than Phrase Based Statistical Machine Translation.

1 Introduction

Human translators translate text by using their world knowledge, grammar rules and understanding of the context. They translate the text by attempting to decipher the source text on three levels: Semantic level: understanding words out of context, as in a dictionary. Syntactic level: understanding words in a sentence. Pragmatic level: understanding words in situations and context. Machines are not able to do the same in understanding the text.

Several techniques, their results and error analysis have helped build basic ideas for various Machine Translation (MT) systems. MT systems can be divided primarily into rule-based, statistical, and hybrid. The popular statistical models are word-based IBM models (Brown et al., 1998), phrase-based (Koehn et al., 2003) followed by example-based (Nagao, 1984), syntax based (Wu, 1997) etc. Further, we describe our work by explaining HPBSMT and why we preferred it over Phrase based SMT. We go ahead describing the work done to develop, train and tune the system and present our results with detailed error analyses.

2 Hierarchical Phrase Based Statistical MT

Phrase-Based models introduced phrases as a basic unit of translation, thus, making sentences a concatenation of two or more phrases. This approach is good at removal of translation error caused due to local reordering, translation of short idioms, insertions and deletions.

Phrase Based MT provides quite precise translations of phrases that commonly occur in training data. As the method relies on the training data, the performance of the system does not improve much, if the phrases are longer than three words. This is because the data is too sparse to learn longer phrases.

There are some Phrase Based MT models which incorporate no reordering, and some models which incorporate simple distortion. Models like Alignment Template System (ATS) and IBM phrase-based system make use of phrase reordering models that adds lexical sensitivity.

HPBSMT, on the other hand, uses sub-phrases to remove issues associated with phrase based MT. For e.g., "भारत का प्रधान मंत्री" {bhaarata kaa pradhaana mantrii} should translate to "Prime Minister of India". A possible grammar rule, in this case, is that the phrases on either side of the word *of* will be swapped when translating to Hindi. In case of phrase level translation, this rotation is fixed only for a particular phrase and there are different rules for other phrases requiring similar

rotation. This contributes to increasing redundant rules which are stored in a dictionary.

On the contrary, HPBSMT replaces these rules by a single rule i.e.

$$X \rightarrow \langle\ X_1 \text{ का } X_2\ ,\ X_2 \text{ of } X_1\ \rangle$$

Every rule is associated with a weight w that expresses how probable the rule is in comparison to other rules with same rule in the Hindi side. For ex:- "भारत का राष्ट्रीय पक्षी" {bhaarata kaa raashtriiya pakshii} {India of National bird} - National bird of India. This example will have a similar expression on the Hindi side but different on the English side i.e.

$$X \rightarrow\ \langle\ X_1 \text{ का } X_2, X_1\ '\text{s } X_2\ \rangle$$

This is an advantage of using sub-phrases. Basically, Hierarchical phrase based model not only reduces the size of a grammar, but also combines the strength of a rule-based and a phrase-based MT system. It can be observed from the working of grammar extraction or decoding because hierarchical model uses rules to express longer phrases. But, It keeps smaller phrases as they were. Synchronization is required between sub-phrases because these sub-phrases need to have a number attached to them since they are essentially all X.

This model does not require parser at the Hindi side because all the phrases are labeled as X. This is very important with respect to Indian languages, since none of the Indian languages have a good automated parser at the moment.

We used cdec (Dyer et al., 2010) for developing our systems.

2.1 cdec

cdec[1] is an open source frame-work for decoding, aligning with, and training a number of SMT models, including word-based models, phrase-based models, and models based on synchronous context-free grammars.

cdec uses a language model to assess the goodness (fluency, grammaticality, semantic coherence) of a sentence in a particular language. We used a language model which was built using a monolingual corpora of approx. 45 million lines. For language model training, we included the monolingual corpora by (Bojar et al., 2014), and added more Hindi monolingual corpora[2] after un-tagging it.

[1] http://www.cdec-decoder.org

3 Related Work

(Chiang, 2005)'s approach in SMT used hierarchical phrases. It combined fundamental ideas from both syntax based translation and phrase based translation.

The advantage of this approach is that hierarchical phrases have recursive structures instead of simple phrases. This higher level of abstraction of this approach, further improved the accuracy of SMT system. Approaches to syntax-based SMT have varied in their reliance on syntactic theories, or annotations made according to syntactic theories. HPB-SMT, in the context of Indian Languages, had earlier been explored by (Bibek et al., 2013) for automated grammar correction.

4 Training Data

We were provided with parallel corpora to train translation models and development sets to tune system parameters. Statistics of the training data are given in Table 1.

Language	Domain					
	Health		Tourism		General	
	Train	Dev	Train	Dev	Train	Dev
Bengali	24000	500	24000	500	48000	1000
English	24000	500	24000	500	48000	1000
Marathi	24000	500	24000	500	48000	1000
Tamil	24000	500	24000	500	48000	1000
Telugu	24000	500	24000	500	48000	1000

Table 1: Statistics of the Parallel data.

5 System Details

The block diagram of our MT system is shown below. After a careful look at the data provided, we realized that some tokenization and normalization of the data was required. After performing the tokenization of data, we started running the decoder on the data and analyzing the output for errors. The preprocessing of data and preparation before inputting it into the system is described below.

[2] http://www.cfilt.iitb.ac.in/wsd/annotated_corpus/

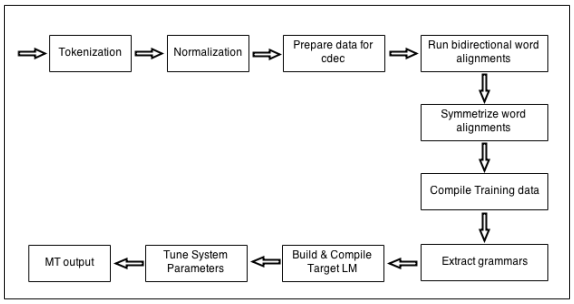

Figure 1: Block diagram of MT system

5.1 Preprocessing

5.1.1 Tokenization

The first step to develop the MT system was Tokenization of the datasets. Tokenization is the process of breaking a stream of text up into words, phrases, symbols, or other meaningful elements called tokens. We implemented a tokenizer inorder to identify tokens correctly. The datasets provided to us was from health and tourism domain which contained lot of acronyms, abbreviations related to diseases, organizations, dates, etc. Tokenization and lowercasing are techniques used for reducing data sparsity.

5.1.2 Corpus Normalization

Followed by the tokenization of the datasets, we performed Corpus Normalization. By text normalization, we mean converting informally inputted text into the canonical form, by eliminating noises or non-standard words in the text. The datasets provided to us for the contest contained many inconsistencies. We used simple regular expressions for the normalization of the text. Following are some of the inconsistencies which were observed and corrected:

- The data provided to us contained acronyms like O.R.S. and ORS for a similiar concept. Following is an example from Hindi.
  Sentence 1:
  बच्चे को निरंतर ओआरएस का घोल पिलाएँ ।
  Sentence 2:
  पहले ओ.आर.एस. का घोल पिलायें , इसके बाद बच्चे को किसी शिशु रोग विशेषज्ञ से शीघ्र दिखायें ।
  From the example mentioned above, we see that the same word ओआरएस is written in two different ways.

- Inconsistencies in the tagged data: The Part-of-Speech tags used to tag the datasets were not consistent. It contained tags apart from the BIS tagset provided to us. We used a script to untag the data which removed all the inconsistencies. The untag data was used as our training data.

- Multiple representations of Nukta based characters.

- Inconsistencies in typing: e.g. use of verticle line or pipe instead of purna virama as a sentence delimiter.
  Example 1:
  सोने के समय को निश्चित करें |
  Example 2:
  वे अपने विभाग की पूरी जिम्मेदारी लेते हैं ।
  Multiple representation for same character or word causes data sparcity. Some numbers, dates, acronyms, abbreviations and non-standard "words" were also normalized.

5.2 Prepare data for cdec

This step filters out sentence pairs that have over 100 words (in either language - source and target language) or have an very unusual length ratio, relative to the corpus average. This tends to remove sentence pairs that are either misaligned or will be hard to model.

cdec uses a simple text format to represent parallel corpora. In this format, each parallel sentence is a single line of text with the two parts separated by a triple pipe (|||). Here is an example parallel corpus consisting of three sentences:
exercise everyday . ||| रोज व्यायाम करें ।
eat food everyday . ||| रोज़ खाना खाएं ।
drink water everyday. ||| रोज़ पानी पियें ।

5.3 Word alignment

Word alignment is the process of identifying words or phrases that correspond in meaning in the source and target portions of a parallel sentence. Word alignment is often used to constrain the set of translation rules that are extracted from parallel sentences. Word alignments are generated using the fast align tool. First the alignment is done in forward mode followed by the reverse mode. The "for-

ward" and "reverse" alignments are later symmetrized.

cdec conventionally uses a variant of fast_align (Dyer et al., 2013). fast_align produces outputs in the widely-used *i-j* "Pharaoh format," where a pair *i-j* indicates that the *i*th word (zero-indexed) of the left language (by convention, the source language) is aligned to the *j*th word of the right sentence (by convention, the target language).

5.4 Compile Training Data

This step compiles the parallel training data into a data structure called a suffix array that enables very fast lookup of string matches. By representing the training data as a suffix array, it is possible to do a targeted extraction of rules for any input sentence, rather than extracting all rules licensed by the training data.

5.5 Extract Grammar

The model for extracting grammar is based on Synchronous Context Free Grammar (SCFG) also known as syntax-directed transduction grammar. A SCFG derivation begins with pair of linked start symbols unlike, in Context Free Grammar (CFG) which contains single start symbol. Further, at each level two components of single rule are applied which rewrites the two linked non terminals. The symbols i.e. non terminals are numbered to avoid ambiguities when there are same elements occurring twice on both the side. cdec contains a suffix array grammar extractor that can be used to efficiently extract SCFG grammars from very large corpora.

5.6 Language Model

KenLM is the only supported language model in cdec. We experimented with 5-gram and 4-gram unpruned language model with modified Kneser-Ney discount estimated with KenLM toolkit. Our system finally uses 5-gram unpruned language model as the target language model.

5.7 Tuning

cdec includes implementations of many discriminative parameter learning algorithms like MIRA, MERT, PRO etc. We used a variant of the Margin-Infused Relaxed Algorithm or MIRA (Chiang, 2012), a loss-aware large margin learning technique.

6 Result and Error Analysis

Table 3 gives the statistics of the number of translated sentences correctly matched with the reference data. Following are some of the examples:
Example 1:
Input: Iron is in abundance in eggs , fish .
Output: आयरन अंडे, मछली में प्रचुर मात्रा में होता है ।
Example 2:
Input: In Bangalore , monsoon stays from June to September .
Output: बंगलौर में मानसून जून से सितम्बर तक रहता है ।

The BLEU score and TER per language-pair and domain is given in Table 2.

As a post-processing feature, we had earlier used transliteration[3] systems to improve the accuracy. Since, the transliteration modules used parallel corpora, we had to withdraw this feature, and submit our current outputs, as they were. An example of such sentence is as follows:
Example:
Input: People of Hindustan , Pakistan , Bangladesh , Egypt do business in Manama Souk .
Output: hindustan pakistan , bangladesh , मिस्र के लोग मानामा सूक में व्यापार करते हैं ।
Reference: हिंदुस्तान , पाकिस्तान , बंग्लादेश , मिस्र के लोग मानामा सूक में व्यापार करते हैं ।

We can infer from the example above that a Transliteration system or Named Entity Recognition (NER) module would have improved the accuracy of our system.

6.1 Limitations of *cdec*

During our experiments, we came across some limitations while using the cdec system, it provides only limited support for extracting translation grammars from parallel data. It can only be used for batch operations and not for online operations, as of now. We are trying to figure out a way to do it.

6.2 Limitations of BLEU

BLEU is a simple metric for MT evaluation and omits many linguistic features while com-

[3] `http://www.cfilt.iitb.ac.in/Tools.html`

	Bn-Hi		*En-Hi*		*Mr-Hi*		*Ta-Hi*		*Tel-Hi*		*Avg. score*	
	BLEU	TER	BLEU	TER	BLEU	TER	BLEU	TER	BLEU	TER	BLEU	TER
Health	34.24	46.19	27.55	59.18	38.16	45.1	21.13	65.01	29.41	52.56	**30.098**	**53.608**
Tourism	34.05	46.98	23.45	65.13	38.05	43.81	17.39	68.55	26.38	53.68	**27.864**	**55.630**
General	37.28	44.55	26.93	60.63	40.44	43.02	21.23	64.99	29.38	51.45	**31.052**	**52.928**
Avg. Score	**35.190**	**45.907**	**25.977**	**61.647**	**38.883**	**43.977**	**19.917**	**66.183**	**28.390**	**52.563**		

Table 2: Average BLEU score and TER per lang-pair and Domain.

	Bn-Hi	*En-Hi*	*Mr-Hi*	*Ta-Hi*	*Te-Hi*
Health	17	20	24	5	12
Tourism	25	17	30	4	10
General	63	39	57	16	29

Table 3: Statistics of number of translated sentences correctly matched with the reference data.

paring the output with reference data. BLEU remains a benchmark for assessment of MT systems, however, it has been criticized by many researchers. Following are some of the limitations of BLEU score:

1. It captures only word-level similarity i.e. it computes n-gram based scores and ignores semantic similarity of words, for e.g., synonyms.
   Example 1:
   Input: They tell that water birth is a completely natural method .
   Output: वे बताते हैं कि वॉटर बर्थ पूरी तरह प्राकृतिक पद्धति है ।
   Reference: वे बताती हैं कि वॉटर बर्थ पूरी तरह प्राकृतिक पद्धति है ।
   Example 2:
   Input: There is good arrangement of stay here .
   Output: यहाँ ठहरने की अच्छी व्यवस्था है ।
   Reference: यहाँ ठहरने की बेहतर व्यवस्था है ।
   The meaning conveyed by both the outputs above is same as the reference provided.

2. BLEU cannot capture meaning similarity. We observed that the same meaning can be represented in very different lexical and grammatical forms. For example,

   Input: Muscles burn more calories as compared to fat .
   Output: मांसपेशियाँ फैट के मुकाबले ज्यादा कैलोरी बर्न होती है ।
   Reference: मांसपेशियाँ फैट के मुकाबले ज्यादा कैलोरी जलाती है ।

3. Diificult to interpret the meaning of BLEU.

7 Conclusion and Future Work

We have presented our work and experiences in developing fifteen HPBSMT systems for five language pairs namely Bengali-Hindi, English-Hindi, Marathi-Hindi, Tamil-Hindi, and Telugu-Hindi. We have described the corpora used and the details of training the systems in necessary detail. We also have evaluated the systems and given analyses of sample translations.

Indian languages are morphologically rich and close cousins to each other. Dravidian languages are agglutinative and similar to Marathi morphologically. The Marathi to Hindi translation by SMT is more or less at a high quality whose morphemes map to appropriate words/post positions in Hindi.

Further study into proper utilization of factors will be undertaken to improve quality. Interjection of a Rule Based MT system like Sampark mentioned by (Bhosale et al., 2011) and (Nair et al., 2013) can be done as a post processing feature to improve the quality of our system for Marathi-Hindi.

We would also like to integrate a transliteration system, along with a NER module to improve our accuracy.

We will also be looking into source level inflection handling and Word Sense Disambiguation (WSD), for a better lexical choice, in order to improve our accuracy.

Our experiences should be applicable in the development of high quality HPBSMT systems for these language pairs thereby effecive

sharing of knowledge written in any Indian language.

Acknowledgements

We would like to acknowledge Shri Arjun Atreya for their continous motivation and support. We sincerely thank the organizers of the contest for their hard work, year after year, and the reviewers for their careful evaluation of the system.

References

Ondrej Bojar, Vojtěch Diatka, Pavel Rychlý, Pavel Stranak, Vit Suchomel, Aleš Tamchyna, Daniel Zeman. 2014. *HindEnCorp - Hindi-English and Hindi-only Corpus for Machine Translation.* In *Proceedings of the Ninth International Conference on Language Resources and Evaluation* (LREC), May, 2014.

Ganesh Bhosale, Subodh Kembhavi, Archana Amberkar, Supriya Mhatre, Lata Popale and Pushpak Bhattacharyya. 2011. *Processing of Participle (Krudanta) in Marathi.* In *International Conference on Natural Language Processing*, (ICON 2011), Chennai, December, 2011.

Bibek Behera, Pushpak Bhattacharyya 2013. *Automated Grammar Correction Using Hierarchical Phrase-Based Statistical Machine Translation.* In *Proceedings of IJCNLP*, October, 2013, Nagoya, Japan.

P. Brown, J. Cocke, S. D. Pietra, V. D. Pietra, F. Jelinek, R. Mercer and P. Roossin. 1998. A statistical approach to language translation. In *Proceedings of the 12th of Computational Linguistics*, pages 71-76, Budapest, Hungary.

David Chiang. 2005. *A Hierarchical Phrase-Based Model for Statistical Machine Translation.* In *Proceedings of Association of Computational Linguistics*, June, 2005, University of Michigan, USA.

David Chiang. 2012. *Hope and fear for discriminative training of statistical translation models.* In *J. Machine Learning Research 13 (2012): 1159-1187*

C. Dyer, A. Lopez, J. Ganitkevitch, J. Weese, F. Ture, P. Blunsom, H. Setiawan, V. Eidelman, and P. Resnik. 2010. *cdec: A Decoder, Alignment, and Learning Framework for Finite-State and Context-Free Translation Models.* In *Proceedings of Association of Computational Linguistics*, July, 2010.

C. Dyer, V. Chahuneau, and N. A. Smith. 2013. *A Simple, Fast, and Effective Reparameterization of IBM Model.* In *Proceedings of Association of Computational Linguistics*, July, 2010.

P. Koehn, F. J. Och and D. Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association of Computational Linguistics on Human Language Technology*, pages 48-54, Morristown, NJ, USA.

Makoto Nagao 1984. *A framework of a Machine Translation between Japanese and English by analogy principle.* In *Artificial and Human Intelligence*, A. Elithorn and R. Banerji (eds.), North Holland, pages 173-180.

Sreelekha S. Nair, Raj Dabre and Pushpak Bhattacharyya. 2013. *Comparison of SMT and RBMT, the Requirement of Hybridization for Marathi Hindi MT.* In *International Conference on Natural Language Processing*, (ICON 2013), Noida, December, 2013.

K. Papineni, S. Roukos, T. Ward and W. J. Zhu. 2002. *BLEU: a method for automatic evaluation of machine translation. ACL-2002: 40th Annual meeting of the Association for Computational Linguistics*, pages 311–318.

Dekai Wu 1997. *Stochastic inversion transduction grammars and bilingual parsing of parallel corpora.*. In *Computational Linguistics*, 23(3):377-403, 1997.