A Punjabi To Hindi Machine Translation System

Gurpreet Singh Josan
Lecturer, Yadvindra College of Engineering, Talwandi Sabo Bathinda.
josangurpreet@rediffmail.com

Gurpreet Singh Lehal
Professor, Dept. of Comp. Sci., Punjabi University Patiala.
gslehal@gmail.com

Abstract

Punjabi and Hindi are two closely related languages as both originated from the same origin and having lot of syntactic and semantic similarities. These similarities make direct translation methodology an obvious choice for Punjabi-Hindi language pair. The purposed system for Punjabi to Hindi translation has been implemented with various research techniques based on Direct MT architecture and language corpus. The output is evaluated by already prescribed methods in order to get the suitability of the system for the Punjabi Hindi language pair.

1. Introduction

The Direct MT system is based upon exploitation of syntactic similarities between more or less related natural languages. Although its deficiencies soon became apparent, it remains popular in certain situations due to its usefulness, robustness and relative simplicity. One of such situation is machine translation of closely related languages. The general opinion is that it is easier to create an MT system for a pair of related languages (Hajic et.al. 2000). In the last decade, some of the systems utilizing this approach for translating between similar languages have confirmed this concept. In this paper, our attempt to use the same concept for language pair of Punjabi-Hindi is described.

Punjabi and Hindi both are classified as Indo-Iranian languages. Although they are in the same family, but still they have lot of differences in order to make them not mutually intelligible. Punjabi and Hindi are not mutually intelligible in written form. As far as spoken form is concerned, Punjabi and Hindi are mutually intelligible to certain degree. This relation is further asymmetric with the speakers of Punjabi more able to understand Hindi but reverse is not true.

1.1 Punjabi Language

Punjabi is the official language of the Indian state of Punjab and also one of the official languages of Delhi. It is used in government, education, commerce, art, mass media and in every day communication. A good deal of Sikh religious literature is written in Punjabi language. According to SIL Ethnologue, Punjabi is the language of about 57 million people and ranked 20th among the total languages of the world. It is written in Gurmukhy, Shahmukhy and roman scripts.

1.2 Hindi Language

Hindi on the other hand has been one of the two official languages of all of India. Hindi is a language of about 577 million peoples all over the world and is ranked as 5th most widely spoken language by SIL Ethnologue.

2. The Need

India being a large and multilingual society, and in the interest of the regional languages, the government of India has allowed to use regional languages as the official language of respective region and adopt bilingual form (Hindi/English) as the official language of Union Government. Most of the state governments work in their respective regional languages whereas the union government's official documents and reports are in bilingual form (Hindi/English). In order to have a proper communication there is a need to translate these reports and documents in the respective regional languages and vice versa. Some other applications of Punjabi to Hindi MT system are Text Translation, Website Translation, Message Translation (Email), Cross Language Information Retrieval and Web Service.

Coling 2008: Companion volume – Posters and Demonstrations, pages 157–160
Manchester, August 2008

Existing system: Keeping in view the importance of MT system among Indian languages, an MT system called "Anusaarka" has been developed at IIIT Hyderabad covering all the major Indian languages. It is a language accessor and produces an image of source language in target language. Output will have to be post-edited by a person, to make it grammatically correct, stylistically proper, etc. Moreover, some amount of training will be needed on the part of the reader to read and understand the output. Our system is more practical in nature than Anusaarka and it produce more grammatical and stylistic output. No training is needed on the part of reader.

3. System description

To start with, a direct translation system is created on windows platform, in which words from source language are chosen, their equivalents in target language are found out from the lexicon and are replaced to get target language. The source text is passed through various pre processing phase and out put is also passed through a post processing phase.

3.1 Lexical Resources

In this research work, we have developed and used various resources as follow:

Root word Lexicon: It is a bilingual dictionary that contains Punjabi language word, its lexical category like whether it is noun, verb or adjective etc and corresponding Hindi word. It also contains the gender information in case of nouns and type information (i.e. transitive or intransitive) in case of verb. This dictionary contains about 33000 entries covering almost all the root words of Punjabi language.

inflectional form lexicon: It contains all the inflectional forms, root word and corresponding Hindi word. Ambiguous words has the entry "amb" in the Hindi word field. It contains about 90,000 entries.

Ambiguous word lexicon: It contains about 1000 entries covering all the ambiguous words with their most frequent meaning.

Bigram Table: Used for resolving ambiguity, this table contains Punjabi bigrams along with Hindi meaning. Bigrams are created from a corpus of 7 million words.

Trigram Table: Same as Bigram, but contain Punjabi trigrams used for resolving ambiguity. Created from 7 million words corpus.

3.2 System Architecture

The system architecture, as shown in figure 3.1, has the following stages through which the source text is passed.

Text normalization

There are number of ASCII based fonts to represent Punjabi text and each font has variations in assigning ASCII code to Punjabi Alphabets. This cause a problem while scanning a text. Therefore, the first step is to normalize the source text by converting it into Unicode format. It gives us three fold advantages; first it will reduce the text scanning complexity. Secondly it also helps in internationalizing the system as if the output is in Unicode format then it can be used in various applications in various ways. Thirdly, it eases the transliteration task.

Tokenization

The system is designed to do sentence level translation in order to have a track about the context of a word. Once the whole text is scanned, next step is to break up the data into sentences. Individual words or tokens are extracted out from the sentence and processed to find out its equivalent in the target language. Tokens are separated by using break characters like space, comma, question mark etc.

Translation Engine

The translation engine is responsible for translation of each token obtained from the previous step. It uses various lexical resources for finding the match of a given token in target language. It involves different modules like Named Entity Recognition, Repetitive construct handler, Word Mapping, Ambiguity Resolution, and Transliteration.

The token obtained in the previous stage is passed through following stages:

1. The token is checked for proper names of persons as they need to be transliterated.
2. If token is not a proper name then it is checked for repetitive units like ਘਰੋਂ ਘਰ{gharōṁ ghar}(home to home) by comparing the word and its root with next or previous words and their roots. A limited morph analysis is required for this step. The repetitive construct handling involves two stages. First, detection of repetitive construct and second, handling of such construct. **Detection:** For detection of repetitive construct, we check the next and previous word. If the next and previous words are same or the roots of next and previous words are same as that of current word, then we

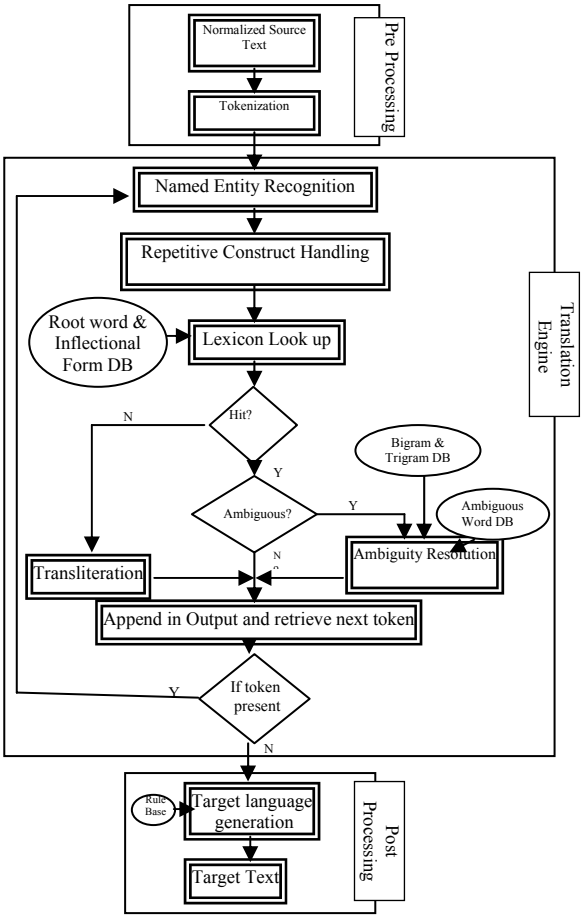

Figure 3.1 System Architecture

mark it as a repetitive construct. The root of the words will be obtained from the database discussed earlier.

Handling: If repetitive construct is found then the next step is to get the lexical information of the token. This information is again obtained from the database. The lexical information of token is used to trigger the handling process. If the token is noun then the token is replaced by its root and then passed to the next step. E.g. in case of ਘਰੋਂ ਘਰ{gharōṁ ghar}(home to home) we check the token ਘਰੋਂ and its root i.e. ਘਰ with the next token which in this case is again ਘਰ. Thus, system marks it as repetitive constructs. Then the lexical category of ਘਰ is checked from the database which comes out to be noun. So we replace the ਘਰੋਂ with its root ਘਰ and pass the replaced token to the next step.

3. Then the token is looked into the database for a match. The Database contains various types of tables. First of all token is checked in the root database and inflectional form database. It gives two types of output if match occurs. Either the corresponding Hindi word is produced or "amb" is appeared which shows that word is ambiguous.
4. For the ambiguous words, we call "resolver" module that resolve the ambiguity with the help of n-gram language modeling. The system uses trigram table in the first place, which contains the two words in the vicinity of an ambiguous word and corresponding meaning for that particular context. If it fails to resolve the ambiguity then bigram table is searched. Bigram table is similar to trigram table except it contains only one word in the vicinity of ambiguous words. If both trigram and bigram fails to resolve then module will use most frequently used meaning.
5. If token is not matched in inflectional form database, then word may be a foreign word i.e. word of other language like English. Such words and all those tokens, for which no entry is found in database, are transliterated. Transliteration is performed in three stages as follow:
   a. Direct Mapping
   b. Rule Based Improvement
   c. Soundex technique Based improvement.
6. The system uses Direct mapping approach at first stage and then applies some rules to make the spellings of output similar to target language. In the third stage soundex technique is used to deal with the special cases like occurrence of half characters and other symbols not present in Punjabi.
7. All these steps are repeated for all the sentences in the source text.

Target Language Generation

After converting all source text to target text, there are some discrepancies as discussed previously and need to be solved. For removing these discrepancies a rule base is used. This database gives the rules to make the text grammatically correct.

4. Implementation

The system is implemented in ASP.net at front end and MSAccess at back end. A class is created whose object will accept a string in Punjabi language and returns its corresponding Hindi string. Based on this class, various online applications are created. A web site is created with interface that enables a user to write his input sentence in Punjabi and system will produce the output in Hindi. Another application enables the user to translate a webpage in Punjabi to Hindi on the fly. The user has to

mention the URL of webpage to be translated. In another application, an online interface for cross language information retrieval system has been created whereby a user can enter his key word in Punjabi. These keywords are translated in Hindi and result is posted to Google search engine. The user is presented with the results returned by Google from Hindi web pages. Another interface enables the users to write E-mail in Punjabi. This message is translated to Hindi and send to the target email address. The receiver get the mail in Hindi. For the developers who want to use this Punjabi To Hindi MT module, a web service is also created.

5. Results

5.1 Subjective test analysis

The overall rating grade for Intelligibility of the translated text came out to be 2.76 on a 3 point scale. About 94% sentences are intelligible.
The overall rating grade for fidelity of the translated text came out to be 2.72 on 3 point scale. Similarly, the accuracy percentage for the system is found out to be 90.67%. The accuracy score is comparable with other similar systems (Hajic J. et.al. 2000; Hric J. et.al. 2000; Homola et.al. 2005) as shown in table 5.1.

MT SYSTEM	Accuracy
RUSLAN	40% correct 40% with minor errors. 20% with major error.
CESILKO (Czech-to-Slovak)	90%
Czech-to-Polish	71.4%
Czech-to-Lithuanian	87.6%
Our System	90.67%

Table 5.1 Comparative analysis of %age accuracy

5.2 Error Analysis

Word Error rate, which is the percentage of erroneous words from all words, is found out to be 2.34%. It is comparably lower than that of the general systems like Salt, Incyta, Internostrum, where it ranges from 3.0 to 4.9 (Tomas J. et.al., 2003). The Sentence Error rate is found out to be 24.26%.

6. Conclusion

The accuracy of the translation achieved by our system justifies the hypothesis that the simple word-for-word translation along with statistical and rule based approach provides a high accuracy and simple solution for language pair of Punjabi and Hindi especially when the objective is just to have a rough idea on the subject matter.

References

Altintas K., Cicekli I., "A Machine Translation System Between a Pair of Closely Related Languages", In Proceedings of ISCIS 2002, October 2002, Orlando, Florida.

Anusaarka-overcoming the language barrier in India, http://www.iiit.net/ltrc/Publications/anuvad.html

Bemova A., Oliva K. Panevova J.," Some Problems of Machine translation between closely related languages", In Proceedings of the 12th conference on Computational linguistics - Volume 1, Budapest, Hungry, 1988 pp 46 - 48

HAJIC J, HRIC J, KUBON V., "CESILKO– an MT system for closely related languages", In *ACL2000, Tutorial Abstracts and Demonstration Notes*, pp. 7-8. ACL, Washington.

Hajic J., "Ruslan-An MT System between closely related languages", In *Proceedings of the 3rd Conference of The European Chapter of the Association for Computational Linguistics*, Copenhagen, Denmark, 1987, pp.113-117.

HRIC J, HAJIC J, KUBON V., "Machine Translation of Very Close Languages", proceedings of the 6th Applied Natural Language Processing Conference, April 29--May 4, 2000, Seattle, Washington, USA. pp 7-12.

Homola P., Kubon V., "A Machine Translationn System into a Minority Language", In Proceddings of the Workshop on *Modern Approaches in Translation Technologies 2005* - Borovets, Bulgaria, pp 31-35.

Marote R. C, Guillen E., Alenda A.G., Savall M.I.G., Bellver A.I., Buendia S.M., Rozas S.O., Pina H.P., Anton P.M.P., Forcada M.L., "The Spanish-Catalan machine translation system interNOSTRM", In proceedings of MT Summit VIII, 18-22 Sept. 2001, Santiago de Compostela, Galicia, Spain.

Scannell K.P.,"Machine Translation for Closely Related language Pair", Proceedings of the Workshop on Strategies for developing machine translation for minority languages at LREC 2006, Genoa, Italy, May 2006, pp103-107.

Slype V., 1979. "Critical Methods for Evaluating the Quality of Machine Translation," Prepared for the European Commission Directorate General Scientific and Technical Information and Information Management. Report BR-19142. Bureau Marcel van Dijk.