International Journal of Computer Science and Informatics

Volume 4 | Issue 1 Article 5

July 2014

NLP CHALLENGES FOR MACHINE TRANSLATION FROM ENGLISH TO INDIAN LANGUAGES

MALLAMMA V REDDY
Department of Computer Science and Applications, Jnanabharathi Campus, Bangalore University, Bangalore, INDIA, mallamma_vreddy@yahoo.co.in

DR. M. HANUMANTHAPPA
hanu6572@hotmail.com

NLP CHALLENGES FOR MACHINE TRANSLATION FROM ENGLISH TO INDIAN LANGUAGES

MALLAMMA V REDDY[1], DR. M. HANUMANTHAPPA[2]

[1]Department of Computer Science and Applications, Jnanabharathi Campus, Bangalore University, Bangalore, INDIA
E-mail: [1]mallamma_vreddy@yahoo.co.in, [2]hanu6572@hotmail.com

Abstract- This Natural Langauge processing is carried particularly on English-Kannada/Telugu. Kannada is a language of India. The Kannada language has a classification of Dravidian, Southern, Tamil-Kannada, and Kannada. Regions Spoken: Kannada is also spoken in Karnataka, Andhra Pradesh, Tamil Nadu, and Maharashtra. Population: The total population of people who speak Kannada is 35,346,000, as of 1997. Alternate Name: Other names for Kannada are Kanarese, Canarese, Banglori, and Madrassi. Dialects: Some dialects of Kannada are Bijapur, Jeinu Kuruba, and Aine Kuruba. There are about 20 dialects and Badaga may be one. Kannada is the state language of Karnataka. About 9,000,000 people speak Kannada as a second language. The literacy rate for people who speak Kannada as a first language is about 60%, which is the same for those who speak Kannada as a second language (in India). Kannada was used in the Bible from 1831-2000. Statistical machine translation (SMT) is a machine translation paradigm where translations are generated on the basis of statistical models whose parameters are derived from the analysis of bilingual text corpora. The statistical approach contrasts with the rule-based approaches to machine translation as well as with example-based machine translation.

Keywords- *Multilingual Cross Langauge Information Retrieval (MCLIR), Morphology, Natural Language Processing (NLP), Statistical machine translation (SMT), Word-Sense Disambiguation (WSD)*

I. INTRODUCTION

\Natural Language Processing (NLP) is the process of computer analysis of input provided in a human language (natural language), and conversion of this input into a useful form of representation.

The field of NLP is primarily concerned with getting computers to perform useful and interesting tasks with human languages. The field of NLP is secondarily concerned with helping us come to a better understanding of human language. The input/output of a NLP system can be 1) written text 2) speech. We will mostly concern with written text (not speech), to process written text, we need: lexical, syntactic, semantic knowledge about the language, discourse information, real world knowledge. To process spoken language, we need everything required to process written text, plus the challenges of speech recognition and speech synthesis. There are two components of NLP. Natural Language Understanding: Mapping the given input in the natural language into a useful representation where Different level of analysis required: morphological analysis, syntactic analysis, semantic analysis, discourse analysis …Natural Language Generation: Producing output in the natural language from some internal representation where Different level of synthesis required: deep planning (what to say), syntactic generation.

- NL Understanding is much harder than NL Generation. But, still both of them are hard. The difficulty in NL understanding arises from the following facts: Natural language is extremely rich in form and structure, and very ambiguous.

– How to represent meaning,
– Which structures map to which meaning structures.

- One input can mean many different things. Ambiguity can be at different levels.

– Lexical (word level) ambiguity -- different meanings of words
– Syntactic ambiguity -- different ways to parse the sentence
– Interpreting partial information -- how to interpret pronouns
– Contextual information -- context of the sentence may affect the meaning of that sentence.

- Much input can mean the same thing.
- Interaction among components of the input is not clear.

A. NLP Terminology

The following language related information are useful in NLP

- Phonology – concerns how words are related to the sounds that realize them.
- Morphology – concerns how words are constructed from more basic meaning units

called morphemes. A morpheme is the primitive unit of meaning in a language.

- Syntax – concerns how can words be put together to form correct sentences and determines what structural role each word plays in the sentence and what phrases are subparts of other phrases.
- Semantics – concerns what words mean and how these meaning combine in sentences to form sentence meaning. The study of context-independent meaning.
- Pragmatics – concerns how sentences are used in different situations and how use affects the interpretation of the sentence.
- Discourse – concerns how the immediately preceding sentences affect the interpretation of the next sentence. For example, interpreting pronouns and interpreting the temporal aspects of the information.
- World Knowledge – includes general knowledge about the world. What each language user must know about the other's beliefs and goals.

B. Ambiguity

Example: I made her duck.

- How many different interpretations does this sentence have?
- What are the reasons for the ambiguity?
- The categories of knowledge of language can be thought of as ambiguity resolving components.
- How can each ambiguous piece be resolved?
- Does speech input make the sentence even more ambiguous?

  – Yes – deciding word boundaries

- Some interpretations of: I made her duck.

  1. I cooked duck for her.
  2. I cooked duck belonging to her.
  3. I created a toy duck which she owns.
  4. I caused her to quickly lower her head or body.
  5. I used magic and turned her into a duck.

- Duck – morphologically and syntactically ambiguous:

Example: noun or verb.

- Her – syntactically ambiguous: dative or possessive.
- Make – semantically ambiguous: cook or create.
- make – syntactically ambiguous:
  – Transitive – takes a direct object. => 2
  – Di-transitive – takes two objects. => 5
  – Takes a direct object and a verb. => 4

C. Ambiguities are resolved using the following methods

- Models and algorithm: are introduced to resolve ambiguities at different levels.
- Part-of-speech tagging: Deciding whether duck is verb or noun.
- Word-sense disambiguation: Deciding whether make is create or cook.
- Lexical disambiguation: Resolution of part-of-speech and word-sense ambiguities are two important kinds of lexical disambiguation.
- Syntactic ambiguity: her duck is an example of syntactic ambiguity, and can be addressed by probabilistic parsing.

D. Models to represent Linguistic Knowledge

We will use certain formalisms (models) to represent the required linguistic knowledge.

- State Machines: FSAs, FSTs, HMMs, ATNs, RTNs
- Formal Rule Systems: Context Free Grammars, Unification Grammars, Probabilistic CFGs.
- Logic-based Formalisms: first order predicate logic, some higher order logic.
- Models of Uncertainty: Bayesian probability theory.

II. A STATISTICAL MACHINE TRANSLATION APPROACH

The idea behind statistical machine translation comes from information theory. A document is translated according to the probability distribution p(k | e) that a string k in the target language (for example, Kannada) is the translation of a string e in the source language (for example, English).

The problem of modeling the probability distribution p (k | e) has been approached in a number of ways. One intuitive approach is to apply Bayes Theorem, that is

$$p(e|f) \propto p(f|e)p(e), \quad (1)$$

Where the translation model p(k | e) is the probability that the source string is the translation of the target string, and the language model p(e) is the probability of seeing that target language string. This decomposition is attractive as it splits the problem into two sub problems. Finding the best translation is done by picking up the one that gives the highest probability:

$$\tilde{e} = \arg\max_{e \in e^*} p(e|f) = \arg\max_{e \in e^*} p(f|e)p(e) \quad (2)$$

For a rigorous implementation of this one would have to perform an exhaustive search by going through all

strings e * in the native language. Performing the search efficiently is the work of a machine translation decoder that uses the foreign string, heuristics and other methods to limit the search space and at the same time keeping acceptable quality. This trade-off between quality and time usage can also be found in speech recognition.

As the translation systems are not able to store all native strings and their translations, a document is typically translated sentence by sentence, but even this is not enough. Language models are typically approximated by smoothed n-gram models, and similar approaches have been applied to translation models, but there is additional complexity due to different sentence lengths and word orders in the languages.

Times is specified, Times Roman or Times New Roman may be used. If neither is available on your word processor, please use the font closest in appearance to Times. Avoid using bit-mapped fonts if possible. True-Type 1 or Open Type fonts are preferred. Please embed symbol fonts, as well, for math, etc.

A. Word-Based Translation

In word-based translation, the fundamental unit of translation is a word in some natural language. Typically, the number of words in translated sentence are different, because of compound words, morphology and idioms. The ratio of the lengths of sequences of translated words is called fertility, which tells how many foreign words each native word produces. Necessarily it is assumed by information theory that each covers the same concept. In practice this is not really true. For example, the English word corner can be translated in Spanish by either rincón or esquina, depending on whether it is to mean its internal or external angle.

Simple word-based translation can't translate between languages with different fertility. Word-based translation systems can relatively simply be made to cope with high fertility, but they could map a single word to multiple words, but not the other way about. For example, if we were translating from French to English, each word in English could produce any number of French words— sometimes none at all. But there's no way to group two English words producing a single French word.

An example of a word-based translation system is the freely available GIZA++ package, which includes the training program for IBM models and HMM model and Model 6[1].

The word-based translation is not widely used today; phrase-based systems are more common. Most phrase-based system are still using GIZA++ to align the corpus. The alignments are used to extract phrases or deduce syntax rules [2]. And matching words in bi-text is still a problem actively discussed in the community. Because of the predominance of GIZA++, there are now several distributed implementations of it online [3].

B. Phrase-Based Translation

In phrase-based translation, the aim is to reduce the restrictions of word-based translation by translating whole sequences of words, where the lengths may differ. The sequences of words are called blocks or phrases, but typically are not linguistic phrases but phrases found using statistical methods from corpora. It has been shown that restricting the phrases to linguistic phrases (syntactically motivated groups of words, see syntactic categories) decreases the quality of translation [4].

C. Syntax-Based Translation

Syntax-based translation is based on the idea of translating syntactic units, rather than single words or strings of words (as in phrase-based MT), i.e. (partial) parse trees of sentences/utterances. The idea of syntax-based translation is quite old in MT, though its statistical counterpart did not take off until the advent of strong stochastic parsers in the 1990s. Examples of this approach include DOP-based MT and, more recently, synchronous context-free grammar.

III. CHALLENGES WITH STATS MACHINE TRANSLATION

A. Sentence alignment

In parallel corpora single sentences in one language can be found translated into several sentences in the other and vice versa. Sentence aligning can be performed through the Gale-Church alignment algorithm.

B. Compound words

In linguistics, a compound is a lexeme (less precisely, a word) that consists of more than one stem. Compounding or composition is the process of word formation that creates compound lexemes (the other word-formation process being derivation). The meaning of the compound may be very different from the meanings of its components in isolation.

C. Idioms

Depending on the corpora used, idioms may not translate "idiomatically". For example, using Canadian Hansard as the bilingual corpus, "hear" may almost invariably be translated to "Bravo!" since in Parliament "Hear, Hear!" becomes "Bravo!"[5]

D. Morphology

In linguistics, morphology is the identification, analysis and description of the structure of a given

language's morphemes and other linguistic units, such as root words, affixes, parts of speech, intonation/stress, or implied context (words in a lexicon are the subject matter of lexicology).

E. Different word orders

Word order in languages differs. Some classification can be done by naming the typical order of subject (S), verb (V) and object (O) in a sentence and one can talk, for instance, of SVO or VSO languages. There are also additional differences in word orders, for instance, where modifiers for nouns are located, or where the same words are used as a question or a statement.

In speech recognition, the speech signal and the corresponding textual representation can be mapped to each other in blocks in order. This is not always the case with the same text in two languages. For SMT, the machine translator can only manage small sequences of words, and word order has to be thought of by the program designer. Attempts at solutions have included re-ordering models, where a distribution of location changes for each item of translation is guessed from aligned bi-text. Different location changes can be ranked with the help of the language model and the best can be selected.

F. Syntax

Syntax deals with the analysis of NLP [12] input on sentence level, the generation of NLP output on sentence level, structural descriptions on sentence level, mostly in form of PS-(phrase structure) trees and/or unification-based formalisms. Structural rules on sentence level (this can vaguely be compared to how "grammar"of a language is traditionally taught)

Acronyms used in structural descriptions of natural language ("vocabulary") = the auxiliary dictionary for the node descriptions:

S =sentence/clause	N = (a single) noun
NP =noun phrase	V =verb
VP =verb phrase	AUX =auxiliary verb
AJ/ADJ=adjective	ADJP =adjective phrase
ADV =adverb	ADVP =adverb phrase
DET =determiner	CONJ =conjunction
COMP =complementizer	PRO =pro-constituent
PUNC =punctuation	

G. Out of vocabulary (OOV) words

SMT systems store different word forms as separate symbols without any relation to each other and word forms or phrases that were not in the training data cannot be translated. This might be because of the lack of training data, changes in the human domain where the system is used, or differences in morphology.

IV. EXPERIMENTAL SETUP

We use a Query Translation [1] [8] based approach in our system since it is efficient to Translate/Transliterate the query vis-a-vis documents. It also offers the flexibility of adding cross-lingual capability to an existing monolingual IR engine by just adding the query translation module. Our MCLIR System uses the following.

- Machine Readable Dictionary: We have used 'BUBShabdasagar-2011' MRD as a translation lexicon resource for our research. The dictionary was available in the ISCII character encoding form and in the plain text format. The entries were converted into UTF-8/Western Windows encoding. The English→Kannada bi-lingual dictionary has around 52,000 English entries and 40,000 Kannada entries. The English→Telugu bi-lingual has relatively less coverage and has around 6110 entries.

- Stop-Word: The English stop word list of 807 English words was used for removing stop word from the query at the time of query formulation. E.g. about, above, across, after, etc.

- Stemmer: The Porter stemmer is used for conflating the morphological variants to a stem word. The Suffix stripping and suffix joining algorithm were used for display the output. E.g. working -> work + ing work=Root word and ing=suffix.

- Transliterator: For overcoming the problem of out-of vocabulary we have used similar scheme to ITRANS [5] transliteration scheme as shown in "Fig. 1".

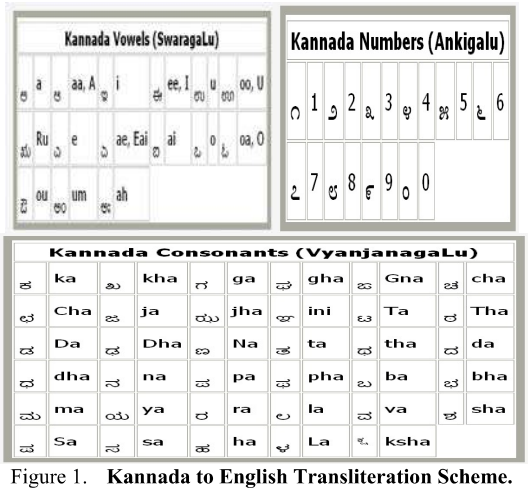


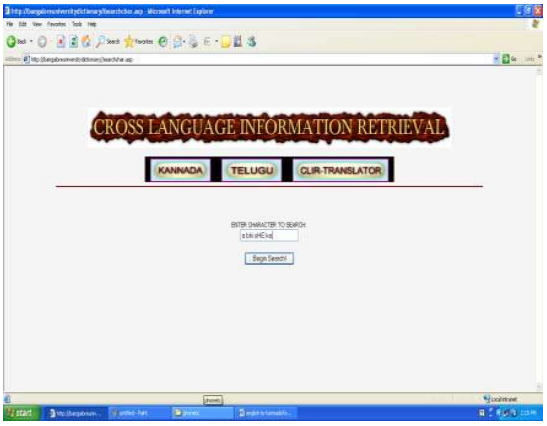
Figure 1. **Kannada to English Transliteration Scheme.**

- Part of Speech Tagger (POST): Traditional grammar classifies the words into different categories. These are called parts of speech. The Stanford part of speech tagger [6] is used for obtaining the part of speech of query term in context of the sentence.

Example: Adjective JJ happy, bad

V. RESULT

Cross Language Information Retrieval Tool is built by using the ASP.NET as front end and for a Database the Kannada is encrypted by using the Encoding system [7], [8] the sample results as shown in below "Fig. 2", and "Fig. 3",



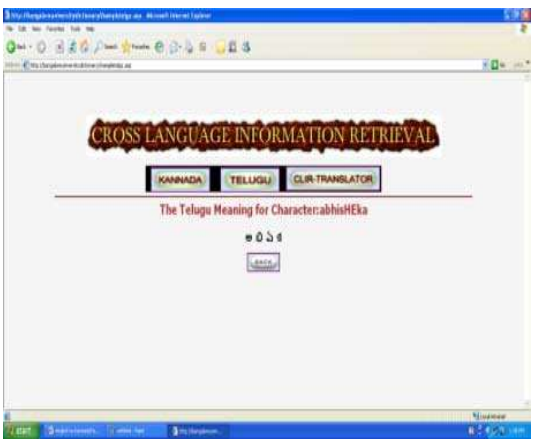
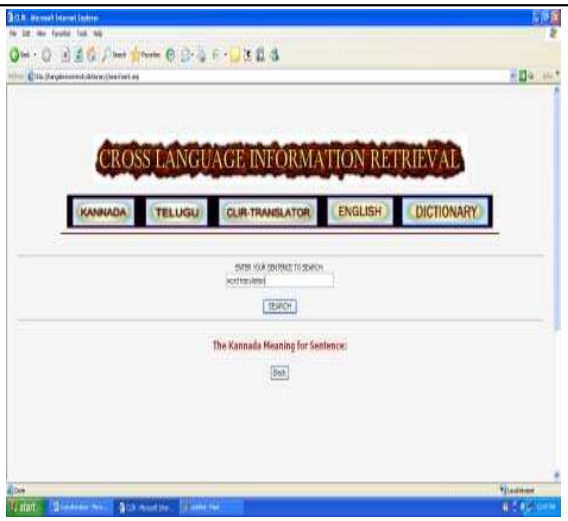


Figure 2. **Sample result for English-Kannada/Telugu Transliteration.**



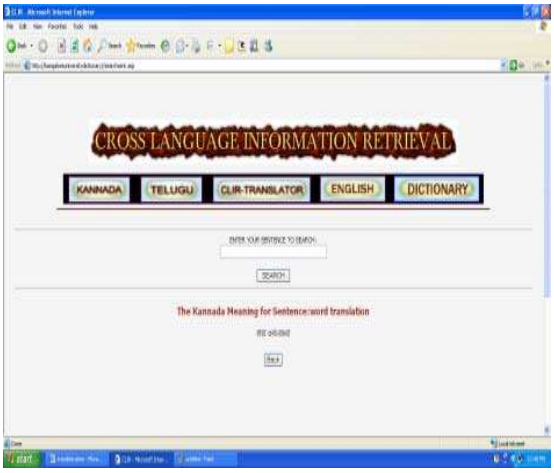


Figure 3. **Sample result for English-Kannada/Telugu Translation.**

VI. CONCLUSION

We presented our English-Kannada and English-Telugu CLIR system developed for the Ad-Hoc bilingual Task. Our approach is based on query Translation using bilingual dictionaries. A statistical

modeling approach to the machine transliteration/translation problem has been presented in this paper. The parameters of the model are automatically learned from a bilingual proper name list. Moreover, the model is applicable to the extraction of proper names and transliterations. The proposed method can be easily extended to other language pairs that have different sound systems without the assistance of pronunciation dictionaries.

ACKNOWLEDGMENT

I owe my sincere feelings of gratitude to Dr. M. Hanumanthappa, for his valuable guidance and suggestions which helped me a lot to write this paper. This is the major research project entitled Cross-Language Information Retrieval sanctioned to Dr. M. Hanumanthappa, PI-UGC-MH, Department of computer science and applications by the University grant commission. We thank to the UGC for financial assistance. This paper is in continuation of the project carried out at the Bangalore University, Bangalore, India.

REFERENCES

[1] F. Och and H. Ney. (2003). A Systematic Comparison of Various Statistical Alignment Models. Computational Linguistics, 29(1):19-51

[2] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, E. Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. ACL 2007, Demonstration Session, Prague, Czech Republic

[3] Q. Gao, S. Vogel, "Parallel Implementations of Word Alignment Tool", Software Engineering, Testing, and Quality Assurance for Natural Language Processing, pp. 49-57, June, 2008

[4] Philipp Koehn, Franz Josef Och, Daniel Marcu: Statistical Phrase-Based Translation (2003)

[5] W. J. Hutchins and H. Somers. (1992). An Introduction to Machine Translation, 18.3:322. ISBN 0-12-36280-X

[6] Language Resource Development available at http://tdil-dc.in

[7] Mallamma.V.Reddy,.Hanumanthappa.M, "Kannada and Telugu Native Languages to English Cross Language Information Retrieval" published in (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 2 (5) , Sep-Oct2011, page-1876-1880. IISN: 0975-9646.

[8] ITRANS. Indian Language Transliteration Package, Available at: www.aczone.com/itrans

[9] STANFORD TAGGER. At http://nlp.stanford.edu/software/tagger.shtml

[10] GIZA++ accessed from "http://www.fjoch.com/GIZA++.html" on 2011.

[11] Article GIZA++ accessed from http://wiki.apertium.org/wiki/Using_GIZA%2B%2B on 2011.

[12] http://www.fbi.h-da.de/organisation/personen/harriehausenmuehlbauer-bettina.html