Local Language Computing (Language Translation Systems)

H.V.Abeywickrama, T.P.K. Dahanaykage, D.M.A. Dissanayakage, K.M.T.V. Ganegedara and R.A.I.J. Ranawaka
Department of Computer Science & Engineering, University of Moratuwa.

Abstract — *Language undoubtedly is one of the most effective communication methods invented ever. But the difference between the languages used in different parts of the world, has negatively affected the rapid development of Information and Communication Technology (ICT) to a considerable extent, resulting in a number of issues including digital divide. This document brings to light the problems faced in Local Language Computing (LLC) such as the complexity of translation from one language, (basically English to another language) and information about the competed and ongoing research in this area.*

This review presents the findings of many researchers under a number of headings. First the document describes the main features of the Local Language Translators. Then it goes on to describe the main facts and concerns of implementing such a translator. The last part of the document consists of the importance of research when it comes to the topic of LLC.

This document basically contains technical and other important details on the findings of some selected researches in the area of Local Language Computing.

Index Terms — Local Languages, Translators, Computation Complexity,

I. Introduction

A. Background

Information and Communication Technology is regarded as a powerful tool for gaining competitive advantage in any industry. Computer literacy rate of most non English using countries is low. Since most of ICT projects are done in English, a majority of people in such countries are not substantially benefited due to lack of fluency in English. This has become the sole reason which makes it difficult to introduce ICT to certain social groups. Subsequently this scenario leads to the digital divide. Local language computing is a research area intended to increase the above discussed ICT penetration. [1]

This review focuses on methodologies and techniques practiced in developing LLC systems, also gives some examples of ongoing projects. Technical details will be described thoroughly in the review. Also review will focus more on local language translation systems and their implementations.

B. Overview

Local language computing is defined as "The process of developing, tailoring and/or enhancing the capability of hardware and software to input process and output information in the language, norms and metaphors used by the community." [Hussain and Mohan (2007)]

Local Language Computing can be mainly categorized into 3 main areas, namely

- Document language conventions to be modeled
- Enable basic input output from a local language
- Develop systems such as language translation systems [1]

II. Local Language Translators

A. Overview

1) Character encoding

For a LLC system to function, a character set must be present. Those characters are represented by a code. To achieve portability across systems this character to code translation should be done using a standard encoding scheme. Also the encoding scheme should support collation and searching and also it should be capable of representing each and every character efficiently.[2]

2) Font

Font is the way a character is presented on the screen. There are numerous ways to write a single character. This implies that numerous fonts can be introduced for the same character. Earlier one to one mapping was used to map characters to font. But it caused problems in some local languages. Newer technologies are able to provide many to one mapping which is also supported by modern operating systems. [2]

3) Text input

This refers to entering fonts using a hardware device; for an example a key board. When we are approaching a generic local language computing system, those devices should also comply with the standards. Depending on the local language, existing key boards may have to be changed or new devices need to be introduced. But all of this must happen according to particular standards. [2]

4) Application Support

Though we can fulfill all of the above requirements, those efforts will be in vain if the application does not support local languages. So application should have local language support with relevant menus, error messages and etc. [2]

5) Utilities

Utilities such as spelling checkers and translators that are needed for improved functionality. [2]

B. *Implementation*

Implementing a local language translation system can be mainly broken into three parts.

a. Corpus
b. POSTAG(Part Of Speech Tagger)
c. Statistical Machine translation system [3]

1) Corpus

Corpus is a huge collection of words from a language. Creating a corpus for a local language will take some effort, as the creators should find appropriate sources. For a local language translation system a parallel corpus is needed, containing local language word and international correspondent to that. For that international language corpus should be translated into local language and local language corpus should be translated into international language and then combine them together. [3]

2) POSTAG

This is a statistical approach to mark words with tags such as adjectives and adverbs. It uses a set of morphological rules to markup the words. Ex: If we take the sentence "The dog ate", we can tag it as the-determiner, dog as the noun and ate as the verb. There are a few different approaches to implement this, which are mainly categorized into statistical methods and non-statistical methods. In non-statistical method there can be a small set of parses which can be legitimately ambiguous. And they sometimes interpret a completely irrelevant meaning for the words. In non-statistical method it is assumed that deciding legitimacy is not a responsibility of the parser and that it should be the responsibility of the disambiguation unit that works in parallel to parser.

But statistical parsing assumes that the only distinction to be made is between correct parse and all the rest.

POSTAG is the first natural processing task to receive a thoroughgoing statistical treatment.

To each word in a sentence a tagger assigns the part of speech that it assumes in that sentence. So a word can have multiple tags. Though most English words have a single part of speech tag there are some words which have multiple tags. Tagger should then have the ability to choose the correct tag.[3][4]

These taggers can be mainly categorized in to following two varieties.[3]

a) Using Conditional Random Fields

This method will use probabilistic means to markup the words. It will define conditional probability distribution of label sequence for given input sequences.

b) Using Maximum Entropy Markov Model

This is a statistical modeling. It assumes that unknown values to be learned are connected in a Markov chain rather than being conditionally independent.

c) Algorithm in brief

A simple algorithm can be used to select the correct tags and assign them. In plain words the algorithm is used to assign each word in the sentence with its most common part of speech tagger. In mathematical terms it can be stated as,

t – All possible tags
w_i – i^{th} word of a sentence

So the most common tag is the one that maximize the probability $P(t|w_i)$. So we have to find

arg maxt $P(t|w_i)$ where arg maxt says "find the t that maximizes the following quantity.

We can extend this to a whole text by looking for most common sequence of tags.

It can be done by introducing another probability; that is $P(t_i|t_i\text{-}1)$, which calculates the probability of a tag with tags before that.[4]

3) Improvements

These tagging can be made more accurate by using machine learning system techniques. Some of such techniques are stated below. [5]

(1) Simple voting

Multiple taggers are used to select a tag for a word. There is a special voting scheme where each tagger can choose its preference. The tag with highest votes will be selected as the correct tag. We can further enhance this by defining the size of vote for each tagger. Either we can use a single vote for each tagger or we can define different number of votes depending on the tagger. Accuracy of the tagger will be taken into consideration when allocating votes in that manner. We can also take into consideration the fact that how often they fail to recognize the correct tag. [5]

(2) Stack probabilistic voting

This is a pair-wise method. One tagger presents tag1 , another tagger tag2. Then they use probabilistic methods to find the best tag. All taggers will continue in this manner. It can be noted that this is a slight different version of earlier discussed voting system. Further enhancements can be made by choosing groups rather than pairs.[5]

(3) Memory based combination

In this method it will store all earlier performed tasks in the memory. When the time comes to take a new decision it will analyze the earlier results and decide what to be done in the current problem. Each earlier task is stored as fixed length vectors and each task is called a case.[5]

(4) Decision tree combination

A decision tree is constructed by recursively partitioning the training set. After the decision tree is constructed pruning is done to increase the effectiveness. Then the constructed decision tree is used to make decisions in selecting appropriate tags.[5]

4) *Statistical Machine Translation System (SMT)*

This is a machine translation paradigm. It uses a statistical model whose parameters are derived by analysis of bilingual text corpus. Translation is done by analyzing probabilistic distribution of a string in a target language.[6]

Machine translations can be analyzed in a deeper sense as follows.

One of the most challenging problems faced by Machine Translation (MT) is to order the translated words in a way such that, they fit the target language.

Although a SMT is able to generate correct word to word translation, it cannot solve the problem that one syntactic and semantic unit in the source language might appear in a different position in the target language. But a monotone SMT system can efficiently handle different word order if this word order disparity is found within the limits of a multiword translation unit and the system can implicitly memorize each pair of source and target phrases in the training stage.[6]

The solutions suggested

- The long-distance reordering problem in a deterministic way, by converting the source portion of the parallel corpus into an intermediate representation, in which source words are reordered to match the target language closely.
- Short-range re-orderings in a non-deterministic way using POS information and an input graph model.

The proposed method splits translation into two independent stages.

S→S′→T

S – Sentence of the source language

S'- The translation to the word order of the targeted language

T- Target sentence

SBR deals with the S → S′ part alone. [6]

For further clarification, an example of a translation system is presented below.

The discussed system is known as BEES(Bilingual Expert for English to Sinhala) machine translation. The concept of "Varanegeema" (conjugation) in Sinhala language has been considered as the philosophical basis of this approach to the development of BEES. The "Varanegeema" in Sinhala language is able to handle large number of language primitives associated with nouns and verbs.[7]

"BEES" comprise seven modules namely English Morphological analyzer, English Parser, English to Sinhala base word translator, Sinhala Morphological Generator, Sinhala Parser, Transliteration module and Intermediate Editor. In addition to the main modules, system comprises four dictionaries, namely, English dictionary, Sinhala dictionary, English-Sinhala Bilingual dictionary and the Concept dictionary. [7]

BEES primarily shares the features with the Rule-based, Context-based and human-assisted approaches to machine translation. The 'BEES' has been implemented using Java and Swi-Prolog to run on both Linux and Windows environments. [7]

The English to Sinhala machine translation system has been evaluated through three steps.

All the language processing primitives such as morphological analyzers, parsers, translator and the transliteration module have been tested through the white box testing approach

Word Error Rate and the Sentence Error Rate were calculated using these evaluation results.

The intelligibility and the accuracy tests have been conducted with the human support.

a) Syntax Based Recording (SBR)

This is one of the most effective solutions for the problem of word ordering in Statistical Machine Translation (SMT). This is based on a pre-translation reordering framework targeted at short and long distance word distortion dependencies. This approach has shown good results to a considerable extent, even for the translation tasks which has great need of reordering such as translations from Chinese to English and Arabic to English. In the process, the translation and reordering models have been trained on a sparse bilingual data.[6]

III. Importance in researches and other facts

A. Research Capacity Building

For better LLC systems to emerge and all these to improve, good researches have to take place. Here, Research Capacity Building (RCB) frameworks come to play.

Research Capacity Building (RCB) frameworks define levels and set of practices that help build capacity.

- Research is the key for local language knowledge.
- Without knowledge, it is hard to set up any policy.
- Government owned research institutions are the most important factor.
- International collaborations are necessary catalysts.
- Open Source Projects are the best place to learn.
- A lot more new opportunities are coming: voice commands, real-time speech translation, voice search engine, etc.[8]

B. Principals of RCB

- Skill development – Need a skill development process through training and supervision
- Training on close to practice research – Direct researches' ability to produce close-to-practice knowledge
- Dissemination and impact – Peer reviewed publications and presentations at academic conferences
- Sustainability and continuity – Maintain acquired skills and structures undertake research
- Infrastructure development – "Infrastructure as a set of structures and processes that are set up to enable the

smooth and effective running of research projects" [Rhee and Riggins (2007)] [8]

A problem faced by most LLC systems is the lack of standards. This leads to incompatibility issues and portability issues. Unicode is a good standard to follow in character encoding in local language computing. [9]

IV. Conclusion

Local Language computing is a key area of information and technology. Its importance is growing day by day. For local language computing to improve more research needs to be conducted. POSTAG plays an important role in local language computing and there are numerous researches to enhance the quality of taggers.

Local Language Computing is a rapidly developing area in today's Information and Technology world. Considerable amount of research has taken place in this regard and solutions that are successful have been developed through the results of the research work. Statistical Machine Translation System (eg: Bilingual Expert for English to Sinhala- BEES) is the most common approaches described in this document. Developing a translation system alone would not suffice. It has to be compatible with the existing systems and must be based on accepted standards. Both these factors are important to the successful development of a translator. Although a lot of research has taken place and solutions have been proposed, no translator which is accurate has ever been made yet. Thus research work has to be promoted further in order to make LLC a complete success.

References

[1] "A Focus group study on local language computing in SMES'." [Online]. Available: http://dl.lib.uom.lk/theses/handle/123/402. [Accessed: 30-Dec-2011].

[2] Gihan Dias, Aruni Goonetilleke.(June, 2004)."Development of Standards for Sinhala Computing".[Online} Available:http://www.siyabas.lk/docs/sinhala%20standards.pdf[Dec.30,2011]

[3] Mirna Adriani, Hammam Riza.(2006,July)."Local Language Computing." Development of Indonesian Language Resources and Translation System. [Online] Available:http://panl10n.net/english/outputs/Indonesia/FinalReportID.pdf [Dec.30,2011]

[4] Eugene Charniac. "Statistical Techniques for natural language parsing". {Online] http://bllip.cs.brown.edu/papers/charniak97statistical.pdf Available: [Jan 15th 2012]

[5] Hans Van Halteren, Jakub Zavrel, Walter Daelemans. "Improving Accuracy in Word Class Tagging through the Combination of Machine Learning Systems". [Online] Available:http://acl.ldc.upenn.edu/J/J01/J01-2002.pdf [Jan 15th 2012]

[6] "ScienceDirect - Computer Speech & Language : Syntax-based reordering for statistical machine translation." [Online]. Available:http://www.sciencedirect.com/science/article/pii/S0885230811000040. [Accessed: 30-Dec-2011].

[7] "A Computational grammar of Sinhala for English-Sinhala machine translation." •Online]. Available: http://dl.lib.uom.lk/theses/handle/123/890. [Accessed: 30-Dec-2011].

[8] Sana Shams,Dr. Sarmad Hussain "Strategies for Research Capacity Building in Local Language Computing: PAN Localization Project Case Study"[Online] Available:http://www.hltd.org/pdf/HLTD201130.pdf[Dec.30,2011]

[9] Amar Gurang "Local Language Computing Standards" {Online]Availbale: http://www.panl10n.net/Presentations/Laos/RegionalConference/LocalLanguageStandards/local_language_standards_nepal.pdf[Accessed:18 jan 2012]