RESEARCH ARTICLE OPEN ACCESS

Hybrid-Statistical Machine Translation From English to Hindi

Srishti Dhamija [1], Kriti Aggarwal [2], Shashi Pal Singh [3], Ajai Kumar [4]
Banasthali Vidyapith [1] & [2]
Banasthali
AAI, Centre for development of Advanced Computing [3] & [4]
Pune - India

ABSTRACT
The fundamental aim of this paper is to take a fragment written in English and translate it in Hindi language by the use of statistical and rule based approach that represents an accurate translation of the original sentence. An n-gram based language model, i.e. a type of probabilistic model, is combined with the syntax based translation model that includes the parsing using CYK algorithm and word alignment by IBM models. In this method, tree frame is basically used as statistical model which is then combined with some linguistically motivated reordering rules to improve the lexical analysis system accuracy. Results are presented according to translation accuracy and efficiency.

Keywords:- Language Model, Syntax-Based Translation Model, Rule-Based Approach, Lexical Analysis, Reordering.

I. INTRODUCTION

Soon after the first electronic computers became available, warren weaver(1949) proposed [5] that computers would one day be able to take a document written in one human language as input and translate it efficiently into the other language automatically, the task which is now referred to as *Machine Translation*. Broadly characterised, *Statistical machine translation (SMT)* is based on automatic text translation by the use of statistical models and examples of translations, by matching fragments of contents to the documents already translated by people and the stitching them together. All knowledge of translation is gathered in a large collection of human translated document, called parallel corpus. This is a natural collection from: news articles, many government proceedings, journals, websites, marketing material etc. Though other machine translation systems which are developed according to their paradigms are also in use, mainly rule based or example based systems. SMT has overcome the academic research about MT systems and achieved significant interest over last two decades.

Statistical MT systems are *statistical* [5] because they choose statistics or these learning techniques as the way of translating a document, gathered from parallel corpora, among many other ways. Now, the language is full of nuance and ambiguity, so any fragment (either short or long) will be translated in many ways. This task of translatable fragments is fundamental to statistical machine translation and is of primary focus. A phrase based translation system finds the *phrase pairs* in parallel corpora which are stored with their frequency statistics. The evaluation of MT system is an active research area in itself. Other than human judgement field relies on automatic measures of output quality i.e. BLEU (Bilingual Evaluation Understudy) metrics. It is a precision based evaluation measure that collects statistics on per sentence basis but these statistics are aggregated over a test corpus to provide more robust evaluation.

Statistical methods are advantageous over non-statistical techniques as they produce better translations. The vague or ill-defined relationships between words, phrases and grammatical structures are captured by probability distribution and statistical techniques. A further benefit of these systems is that they need not rely on features of the languages involved which enable the machine translation systems to be built for multiple language pairs with minimal modifications to the technique. No-doubt knowledge of the languages involved is often needed for improved quality of translation. Additional language specific information including morphological features, reordering and grammatical models are incorporated by statistical models.

II. OVERVEIW OF SMT MODELS

The goal of machine translation is to translate from an English input sentence i.e. f to an output Hindi

sentence i.e. e, that has the equivalent meaning as f. we do this by building statistical model to show the translation process, and find

$$E=\text{argmax } p(e|f)$$

Brown et al. (1993) [4] introduced source channel model, where e i.e. the output language sentence is viewed as being generated by the source with probability p(e) defined by the language model and then passed to translation channel to produce f, the input language sentence ,according to the translation probability p(f | e). The task of translation system is to determine e from observed sentence f and the best translation is by computing:

$e = \text{argmax } P(f|e)\, P(e)$

Using Bayes theorem, this problem can be decomposed as:

$$P(f \mid e)P(e) = \frac{\text{argmax } P(f|e)P(e)}{P(f)}$$

Since the source text f is constant across any alternative translation, it can be disregarded as

$$P(f \mid e)P(e)=\text{argmax } P(f|e)P(e)$$

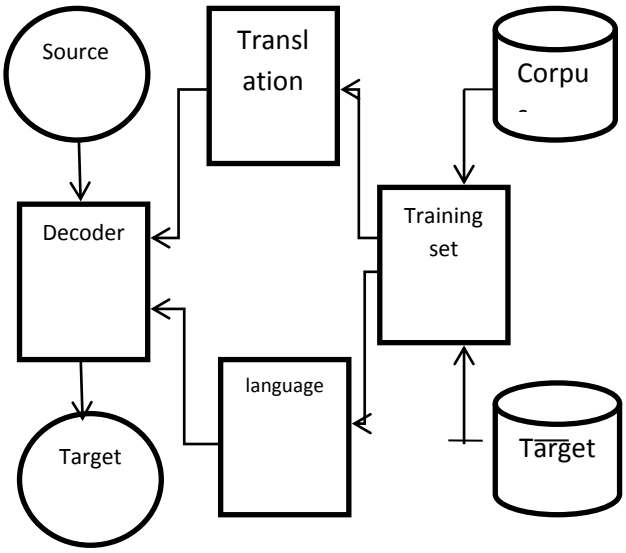


Fig.1 Statistical Machine Translation

So this generative model [4] which resulted from decomposition of p(e|f) produce two fundamental components of basic SMT: the *language model* p(e) and *the translation model* p(f|e). The language model searches for the best translation regardless of the input text whereas the translation model conditions the search for best translation on the input text.so these two components provide adequacy and fluency to the translated text. The third component is *decoder*, a module that performs the search for best translation e, given the space of all possible translations depending on probability estimates P(e) and P(e |f).

A. Language Model

The LM [4] tries to estimate the likelihood of a given sentence translation in the target language. The more common it is, the more likely it will be that it is a good translation mainly in the terms of fluency. This is done by counting the relative number of occurrences of the sentence in a monolingual corpus. P(e) for a sentence with m words is defined as the joint probability of a sequence of all words in that sentence;

$$P(e)= P(w_1,w_2,\ldots..,w_m)$$

This is then decomposed into series of conditional probabilities by applying the chain rule:

$$P(e)=P(w_1)P(w_2|w_1)P(w_3|w_2w_1)P(w_4|w_3w_2w_1)\ldots.P(w_m \mid w_1\ldots. w_{m-1}).$$

So the probability of a word w, given a number of previous words, is calculated using *Maximum Likelihood Estimation* (MLE) i.e. the count of occurrences of the complete sequence divided by the count of conditional sequence.

$$P(w_3|w_1w_2)= \frac{\text{count } P(w_1w_2w_3)}{\text{count } P(w1w2)}$$

1) N-Grams:

In a large corpus, chances of searching occurrences of a given new sentence to translate is very small. In case if not even a single occurrence of sequence of words is seen in the corpus, P(e) will tend to 0 and so will P(e|f). Solution to this is finding the occurrence of parts of such sentences, more specifically the n-grams [4] or a sequence of up to n words. Larger the n greater will be the information about the context of sequence. Whereas smaller the n, it will be more reliable because more cases will be seen in the training data and better will be the statistical estimates. Generally size of n varies according to the size of corpus i.e. the greater the corpus longer will be the n grams that can be counted. N gram models are based on Markov assumption that probability of a word can be calculated depending on its entire history by computing the probability of a word given the last few words as in bigrams as shown:

$$P(e)= P(w_1)\, P(w_2|w_1)\, P(w_3|w_2)\ldots..P(w_m|w_{m-1})$$

B. Translation Model

Second stage [4] of SMT system is translation modelling which includes the step of word alignment over the sentence aligned bilingual corpus. Most systems still use generative models for this purpose

such as one implemented in freely available tool GIZA++. It is an implementation of IBM alignment models which treat word alignment as a hidden process and maximize the probability of (e ,f) pairs using Expectation Maximization algorithm.
For better alignment of Indian languages, information about the cognates is needed as Indian languages have borrowed a large number of words from English. This list was prepared by CPMS (computational phonetic model of scripts) and is added to bilingual corpora for initialization of EM algorithm.

2) CYK Algorithm

Cocke-Younger-Kasami algorithm[8] is a parsing algorithm for Chomsky Normal Form(CNF). It uses bottom-up parsing and dynamic programming. It has high efficiency in certain situations and the worst case running time of CYK is Theta (n^3.|G|), where n is length of the parsed string and | G | is the size of CNF grammar.

Generating a parse tree

This algorithm is only a recognizer which will only determine if the sentence is in the language. It can further be extended into a parser that also construct a parse tree, by storing parse tree nodes as the elements of array. To build the tree structure, the node is linked to array elements that were used to produce it. If all parse trees of the sentence are to be kept, it is mandatory to store in the array element a list of all the ways the node can be obtained. This is done with back-pointers.

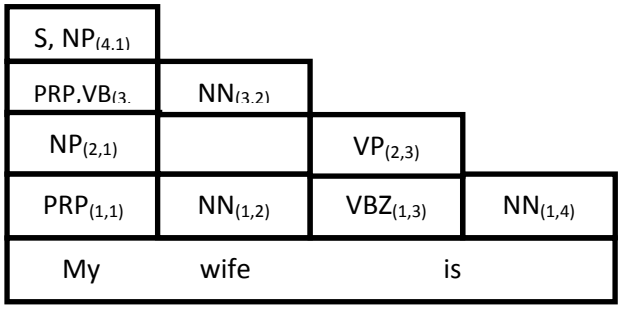


Fig. 2 Example of CYK algorithm

Production –

- S → NP VP
- NP → PRP$ NN
- VP → VBZ NN

String w: My wife is Chinese.

- PRP→ My
- NN→ wife
- VBZ→ is
- NN→ Chinese.

i is the no. of rows and j is the no. of columns in the table. CYK algorithm correctly computes $a_{i,j}$ for all i and j; thus w is in L(G) if and only if S is in $a_{1,n}$.

ALGORITHM

Step 1: get a POS tag for a SL sentence of length n via Stanford parser, where n is no. of words.

Step 2: in a matrix of size n*n, assign these POS tags to the first row of matrix i.e. a[1][n] where n=1,2,3…n.

Step3: for $(n-1)^{th}$ row,

For first two consecutive tags check in production rules-

If rule exist,

assign LHS non terminal of production to (n-1) row's first column and then jump to third column of (n-1) row and check the same for the next two column values of first row, thus now assigning value to the third column of $(n-1)^{th}$ row.

Else

Check next two consecutive tags and assign value to second column of $(n-1)^{th}$ row, further jump by 1.

Iterate till no. of row's is equal to no. of words.

3) IBM Alignment Models 1 through 3

Och and Ney [1] describe the statistical alignment as trying to compute the probabilistic links between the SL string e, and target language string h, and the alignment a between positions in e and h.

$m_1^j = m_1 \ldots\ldots\ldots\ldots m_j$

Hindi and English sentences contain tokens H and E. Tokens in sentences are aligned according to one another. Set of possible alignments is denoted by A and all translations from H to E is a_e that holds the index of corresponding token E in English sentence.

$A = \{(h,1) : h=1\ldots.H ; e= 1,\ldots.,E\}$
$H \rightarrow e= a_e$
$E = a_e$
Using the above notation the basic alignment model can be given as:

$$Pr(e_1^E \mid m_1^H) = \sum Pr(e_1^E, a_1^E \mid m_1^H)$$

This is ***IBM model.***

Model 2
It overcomes the limitations of model 1 i.e. adds the way of differentiating between the alignments that

align words on the other end of sentences from maximum likely ones. Probability that h^{th} target word is connected to e^{th} source word is calculated by distortion probability.

Model 3

This makes one to many translations probable i.e. fertility based alignment is introduced. Reverse distortion probabilities are assigned uniformly.

***Problems in Word Alignment*[1] :-**

Given a sentence aligned parallel corpus there can be many alignments of a single word in a sentence but we aim to have best out of all as shown:

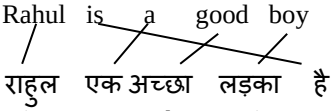
Ex: Rahul is a good boy

There are several problems associated with this approach, based on IBM model, which can be dealt with:

- **Translation model**
- **Distortion model**
- **Fertility model**

The first problem is to find the most likely translation of the given source language (SL) text, irrespective of positions. This is taken care of by the [7] **translation model**.

Ex: Rahul is a good boy

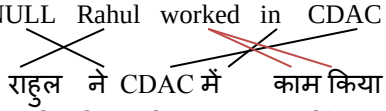


(*one-to-one alignment*)

Second problem is to align positions in SL sentence with positions in the TL sentence, which is addressed by **distortion model** [7].

Word orders of both languages is to be taken care of in this model.

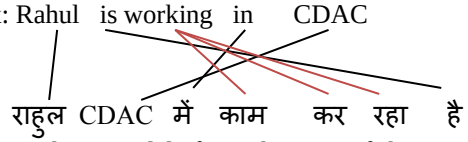
Ex: NULL Rahul worked in CDAC



(*word order and spurious words*)

Third problem is to find out the number of TL words generated from one SL word. Sometimes SL word may generate no TL word or a TL word may be generated by no SL word (**NULL insertion**). The **fertility model** [7] accounts for this.

Ex: Rahul is working in CDAC



These three models form the core of the IBM model based generative SMT. Since English is SVO language and Hindi is SOV, which creates the task of distortion model harder. Apart from TAM (tense, aspect and modality) verbs also creates errors in fertility model because TAM information is distributed over several words which in turn reduces the alignment accuracy. But using the cognate list can help us improving this.

4) EM Algorithm

EM Algorithm [1] is used to find a maximum likelihood parameters of the statistical model. It proceeds from observation that the following is a way to solve these two sets of unknowns and find the probability of the translated output. Then change this with their alternative meanings as per the requirement and further estimate the second set probability. Then use these both to find a better estimate, thus alternating between two until the result converges to fixed points.

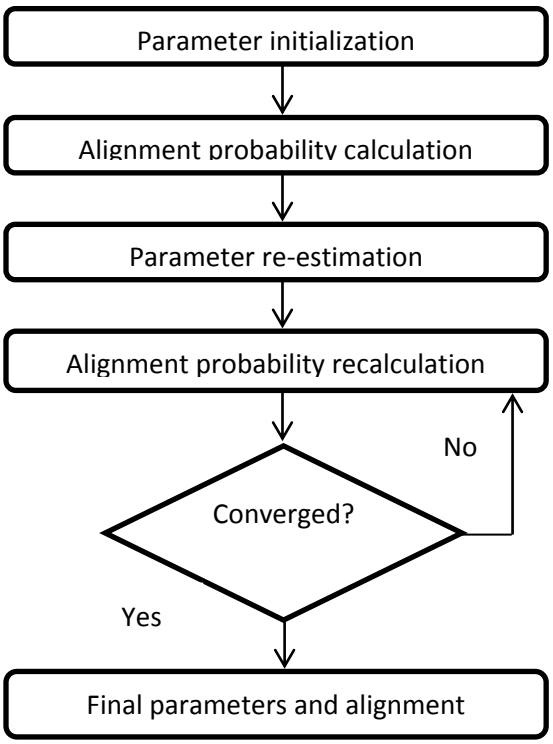


Fig.3 EM algorithm

5) Synonym Handling -

Further this paper gives a way to handle problems of synonyms present in large bilingual corpora.

Ex: ram is a good boy

Hindi Translation-

राम एक अच्छा लड़का है

सामान

Synonyms of the word 'good', as given in corpus, are अच्छा and सामान. Based on the categories and user requirement appropriate meaning of the word is selected by user at run time.

III. RULE BASED RE-ORDERING

Technique of including rule based and morphological analysis give better accuracy of SMT [6]. In this paper we present our work by making some linguistic rules based on tenses, modality etc like we can combine the phrase based models with some reordering rules as per English language.

TABLE 1 LINGUISTIC RULES

TENSE	CONDITION FOR POS TAG	RULES
Simple Present Condition-VB/VBZ	I + do I He + does He She + does She You ,we ,they + do You ,we ,they	Concate' ता'with VB/VBZ/NN& do ='हूँ' 'ता हूँ' Concate'ता ' + does='है' 'ता है' Concate 'ती' with does='है' 'ती है' Concate 'ते' + do= 'हैं' 'ते हैं'
Simple Present continuous Condition-is/am/are + VBG	I He She You, we, they	VBG + 'रहा' Am = 'हूँ' VBG + 'रहा' Is= 'हूँ' VBG + 'रही' Is = 'हूँ' VBG + 'रहे' Are='हैं'

TENSE	CONDITION FOR POS TAG	RULES
Simple past Condition-VBD	ed	'किया गया'
Past continuous Condition-was/were/did + VBG	I He She We/ you/ they/ it + did We/ you/ they/ it	VBG + 'रहा' & was= 'था' VBG + 'रहा' & Was = 'था' VBG + 'रही' & Was = 'थी' VBG + 'रहे' & Were = 'थे' VBG + 'रहे' & Were = 'थे'
Past continuous Condition-was/were/did + VBG	I He She We/ you/ they/ it + did We/ you/ they/ it	VBG + 'रहा' & was= 'था' VBG + 'रहा' & Was = 'था' VBG + 'रही' & Was = 'थी' VBG + 'रहे' & Were = 'थे' VBG + 'रहे' & Were = 'थे'
Simple future Condition-MD && ! be	I He/It She/ This We/ They/ You	Will = 'उंगा' Will = 'एगा' Will = 'एगी' Will = 'एंगे'

Future continuous Condition-MD+ be	I + VBG	VBG + 'ऊँगा' & Will = 'रहा'
	He + VBG	VBG + 'गा' & Will = 'रहा'
	She + VBG	VBG + 'गी' & Will = 'रही'
	You/ we/ they + VBG	VBG + 'गे' & Will='रहे'

REFERENCES

[1] Khin thandar Nwet and Ni Lar Thein- University of Computer Studies, Yangon, Mayanmar : Word Alignment based on Hybrid Approach for Mayanmar-English Machine Translation, | Issue : 2011

[2] Cristina Espana I Bonet LSI Department - Universitat Politecnica de Catalunya : Statistical Machine Translation- a practical tutorial | Issue: March 2010

[3] Shweta Dubey and Tarun Dhar Diwan- Assistant professor Dr. CV Raman University Bilaspur, India: Supporting large English-Hindi parallel corpus using word alignment| Issue : July 2012

[4] Lucia specia – University of Wolverhampton, Stafford street : Fundamental and New approaches to Statistical Machine Translation.

[5] James Brunning- Cambridge University Engineering Dept. and Jesus College : Alignment Models and Algorithms for Statistical Machine Translation| Issue : August 2010.

[6] Rahul.C.Dinunath.K, Remya Ravindran, K.P.Soman- Department of Computational Engineering & Networking, Amrita Vishwa Vidyapeetham, Coimbatore : Rule Based Reordering and Morphological Processing For English-Malyalam Statistical Machine Translation.

[7] G.Chinnapa and Anil Kumar Singh- Language Technologies Research Centre International Institute of Information Technology, Hyderabad : A Java Implementation of an Extended Word Alignment Algorithm Based on the IBM Models|Issue: 2006