Statistical Machine Translation with Rule Based Re-ordering of Source Sentences

Amit Sangodkar
Department of Computer Science and Engineering
IIT Bombay
Mumbai, India
amits@it.iitb.ac.in

Vasudevan N.
Department of Computer Science and Engineering
IIT Bombay
Mumbai, India
vasudevan@cse.iitb.ac.in

Om P. Damani
Department of Computer Science and Engineering
IIT Bombay
Mumbai, India
damani@cse.iitb.ac.in

Abstract

We propose a method of re-ordering the source language sentences as per the target language. This re-ordering is achieved using a Dependency parse of the sentence.. A statistical machine translation system is trained using such a re-ordered corpus. The accuracy of the translation is significantly improved for EILMT data as a result of re-ordering, but it reduced slightly for the IIIT data set. Further work is needed to understand the efficacy of the proposed approach

1 Introduction

A statistical machine translation system aligns the source words of a sentence with the target words in a parallel corpus and builds a phrase table. It uses this phrase table to translate new source language sentences into target language sentences.

The success of any machine translation system depends on how well the source language words are aligned with the target language words. In this paper, we attempt to re-arrange the source sentence as per the syntax of the target language prior to the training process, so that the alignment formed by the machine translation system is improved.

For example, consider the English sentence *Ram broke the window*. If the sentence is re-ordered as per Hindi syntax, it can be written as *Ram the window broke* where the Hindi sentence is *Ram ne khidki todi.* As can be seen, the re-ordered sentence has a better alignment with the Hindi sentence as compared to the original English sentence.

2 Translation Architecture

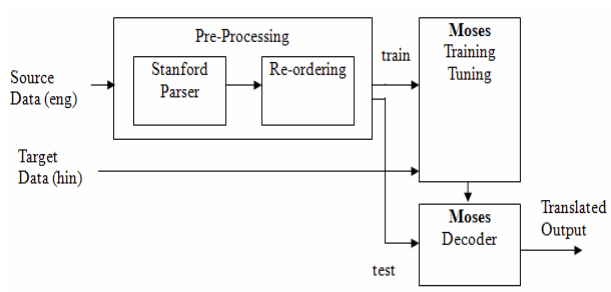

Figure 1: System Architecture

As shown in Figure 1, the input corpus is passed through a pre-processing stage before it is given to statistical MT system. The pre-processing phase consists of two parts. The first part is the Dependency parser which gives the dependency parse of the sentence [Marie2006]. The second part consists of re-ordering the original English sentence as per the target language syntax (Hindi in this case). This re-ordering is done using the typed dependencies among the words in the sentence.

The statistical MT system learns the language model and builds a phrase table based on the re-ordered sentence instead of the original sentence. We have used the Moses toolkit as the statistical

Proceedings of ICON-2008: 6th International Conference on Natural Language Processing
Macmillan Publishers, India. Also accessible from http://ltrc.iiit.ac.in/proceedings/ICON-2008

machine translation system and the Stanford Parser as the Dependency Parser.

The next sections briefly describes the Dependency Parser and illustrates the re-ordering process using the Stanford typed dependencies.

3 Dependency Parser

Dependency parser gives the dependencies among the words in a sentence. Consider the following example,

Sentence 1. *Many Bengali poets have sung songs in praise of this land.*

The dependency parse given by the Stanford Parser is:

Dependency Parse Tree
amod (poets-3, Many-1)
nn (poets-3, Bengali-2)
nsubj (sung-5, poets-3)
aux (sung-5, have-4)
dobj (sung-5, songs-6)
prep_in (sung-5, praise-8)
det (land-11, this-10)
prep_of (praise-8, land-11)

This sentence, as shown by the dependency parse, consists of relations such as *nsubj (subject), dobj (direct object), amod (adjectival modifier), nn (noun-noun compound), and prep (preposition)* and so on. The detailed description of the dependency relations can be found in [Manual2008]. The first word of the relation is a parent and the second word is the child. The dependency parse can also be represented in the form of a tree as shown below

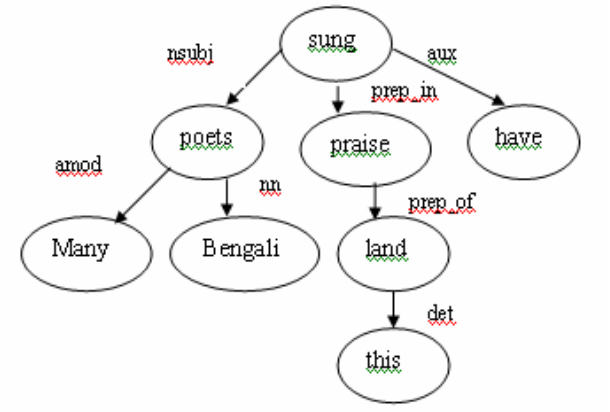

Figure 2

In Stanford Parser, there are 48 typed dependency relations arranged in a hierarchical manner with the most generic relation *dep* as the root. This is the *dependent* relation which is used in case the parser fails to identify any specific relation between two semantically related words in a sentence.

4 Re-ordering

In re-ordering, the original English sentence is rearranged as per the syntax of the target language. The re-ordering scheme is similar to that used in [Singh 2007] for ordering the relations of an Interlingua called UNL. The ordering decisions among dependency relations having a common parent are done based on two aspects: *parent-child positioning* and *relation priority.*

4.1 Parent-child Positioning

Some relations are such that the parent of these relations is ordered before the child and in some cases it is the other way round. Examples of the former type are *conj (conjunction), appos (apposition), advcl (adverbial clause) etc.* For instance, in the sentence "*John cried because he fell",* one of the dependency relation is advcl(cry, fell). In Hindi, *cry* is ordered before *fell* i.e. the parent before the child. In Sentence 1, for dependency relation nsubj(sung, poets), *poets* is ordered before *sung* i.e. the child before the parent.

4.2 Prioritizing the Relations

When a parent has multiple children in the dependency parse tree, the children nodes of the parent need to be ordered. This is done by assigning a priority to each relation. Higher the priority of a relation, the corresponding child node (relata) is ordered leftmost as compared to other relatas. In dependency parse of sentence (1), *nsubj* has higher priority than *dobj,* prep has a lower priority than *nsubj* but higher priority than *dobj,* so it's child word is ordered between that of *nsubj* and *dobj.*

Based, on these priorities between relation pairs, the final re-ordering is

Many Bengali poets this land of praise in songs sung have

which is similar to the syntax of the corresponding Hindi sentence

Kai kaviyon ne is mahaan bhoomi ki prashansa ke geet gaaye hai.

5 Experimental Setup

We have used the Moses toolkit as the statistical machine translation system. The source language is English and the target language is Hindi.

Initially the training corpus is cleaned. Sentences with length greater than 40, empty lines, and redundant spaces were removed. Then a 6-gram language model is learnt. After this the entire Moses tool is trained using train-factored-phrase model with alignment option as grow-diag-final-and and reordering option as msd-bidirectional-fe. This training is done using the original sentences.

After the training process, the tool is tuned by tuning scripts provided with Moses. Using the tuned system, translation of development data and testing data is done. Maximum phrase length for the decoding is 7. The score so obtained is taken as baseline accuracy of Moses in this experimental setup.

In the next step, the process is repeated with re-ordered English sentences instead of original English sentences.

6 Results

Results obtained are summarized in the Table 1.

Corpus	Metric	Baseline		Re-ordered	
		Dev	Test	Dev	Test
EILMT	BLEU	0.1488	0.1450	0.1751	0.1601
	NIST	4.7600	4.7287	4.8539	4.6923
IIIT Data Set	BLEU	0.0815	0.0842	0.0836	0.0853
	NIST	3.9036	4.2426	3.7335	4.0140

Table 1: Experiment Results

The BLEU score has improved from 0.1488 to 0.1755 using development data and the score on test data also shows improvement from 0.1450 to 0.1601 for EILMT data set. These results imply an improvement in the translation output of Moses system, using the re-ordered source language sentences, over the baseline system. However performance actually goes down slightly for the IIIT data set (NIST score). Further work is needed to find the reason for the low score.

There are also some limitations with this scheme. The main source of error is the parse errors of the Dependency Parser wherein the relation among words is captured incorrectly.

References

[Hieu2008] Hieu Hoang, Philipp Koehn, *Design of the Moses* Decoder *for Statistical Machine Translation*, ACL Workshop on Software engineering, testing, and quality assurance for NLP 2008.

[Marie2006] Marie-Catherine de Marneffe, Bill MacCartney and Christopher D. Manning, *Generating Typed Dependency Parses from Phrase Structure Parses*. In *Proceedings of LREC-06*. 2006.

[Manual2008] Stanford Dependencies Manual, Available at http://nlp.stanford.edu/software/dependencies_manual.pdf..

[Moses] Moses Tutorial, Available at http://www.statmt.org/moses/?n=Moses.Tutorial. .

[Singh2007] Smriti. Singh, Mrugunk. Dalal, Vishal Vachhani, Pushpak Bhattacharyya, Om P. Damani. Hindi Generation from Interlingua (UNL), Machine Translation Summit XI, 2007.