A Data Mining Approach to Learn Reorder Rules for SMT

Avinesh PVS
IIIT Hyderabad
Language Technologies Research Centre
avinesh@research.iiit.ac.in

Abstract

In this paper, we describe a syntax based source side reordering method for phrase-based statistical machine translation (SMT) systems. The source side training corpus is first parsed, then reordering rules are automatically learnt from source-side phrases and word alignments. Later the source side training and test corpus are reordered and given to the SMT system. Reordering is a common problem observed in language pairs of distant language origins. This paper describes an automated approach for learning reorder rules from a word-aligned parallel corpus using association rule mining. Reordered and generalized rules are the most significant in our approach. Our experiments were conducted on an English-Hindi EILMT corpus.

1 Introduction

In recent years SMT systems (Brown et al., 1990), (Yamada and Knight, 2001), (Chiang, 2005), (Charniak et al., 2003) have been in focus. It is easy to develop a MT system for a new pair of languages using an existing SMT system and a parallel corpora. It isn't a surprise to see SMT being attractive in terms of less human labour as compared to traditional rule-based systems. However to achieve good scores SMT requires large amounts of sentence aligned parallel text. Such resources are available only for few languages, whereas for many languages the online resources are low. So we propose an approach for a pair of resource rich and resource poor languages.

Some of the previous approaches include (Collins et al., 2005), (Xia and McCord, 2004). Former describes an approach for reordering the source sentence in German-English MT system. Their approach involves six transformations on the parsed source sentence. Later propose an approach which automatically extracts rewrite patterns by parsing the source and target sides of the training corpus for French-English pair. These rewritten patterns are applied to the source sentence so that the source and target word orders are similar. (Costa-jussà and Fonollosa, 2006) consider Part-Of-Speech (POS) based source reordering as a translation task. These approaches modify the source language word order before decoding in order to produce a word order similar to the target language. Later the reordered sentence is given as an input to the standard phrase-based decoder to be translated without the reordering condition.

We propose an approach along the same lines those described above. Here we follow a data mining approach to learn the reordering/rewrite rules applied on an English-Hindi MT system. The rest of the paper is organized as follows. In Section 2 we briefly describe our approach. In Section 3 we present a rule learning framework using Association Rule Mining (Agrawal et al., 1993). Section 4 consists of experimental setup and sample rules learnt. We present some discussion in Section 5 and finally detail proposed future work in Section 6.

Proceedings of the NAACL HLT 2010 Student Research Workshop, pages 52–57,
Los Angeles, California, June 2010. ©2010 Association for Computational Linguistics

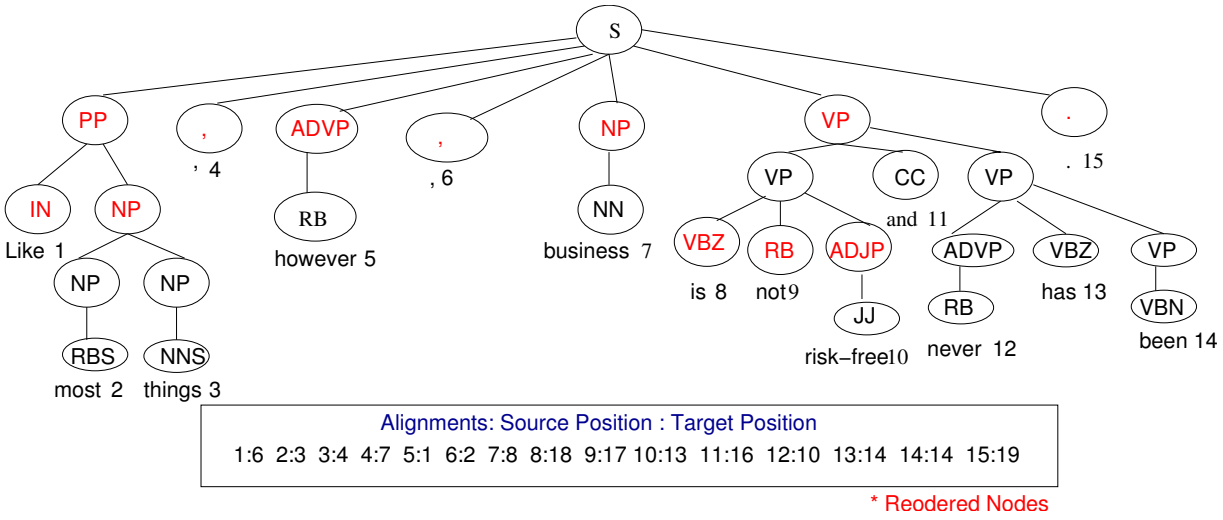

Figure 1: English-Hindi Example

2 Approach

Our approach is inspired by Association rule mining, a popular concept in data mining for discovering interesting relations between items in large transaction records. For example, the rule {milk, bread} ⇒ {butter} found in the customer database would indicate if a customer buys milk and bread together, he or she is also likely to buy butter. Similar notions can be projected to the learning of reorder rules. For example, {NNP, VB, NNP} ⇒ {1,3,2} would indicate if NNP,VB and NNP occur together in source text, then its ordering on the target side would be {1,3,2}. The original problem of association rule mining doesn't consider the order of items in the rule, whereas in our problem order is important as well.

In this approach we start with extracting the most frequent patterns from the English language model. The English language model consists of both POS and chunk tag n-gram model built using SRILM toolkit [1]. Then to learn the reordering rules for these patterns we used a word-aligned English-Hindi parallel corpus, where the alignments are generated using GIZA++ (Och and Ney, 2003). These alignments are used to learn the rewrite rules by calculating the target positions of the source nodes. Fig 1 shows an English phrase structure tree (PS) [2] and its alignments corresponding to the target sentence.

2.1 Calculation of target position:

Target position of a node is equal to the target position of the **head** among the children (Aho and Ullman, 1972). For example the head node of a NP is the right most NN, NNP, NNS (or) NNX. Rules developed by Collins are used to calculate the head node (Collins, 2003).

Psn(T,Node)=Psn(T,Head(Node))
In Fig 1, Position of VP in target side is 18.
Psn(T,VP)=Psn(T,Head(VP))=Psn(T,VBZ)=18

3 Association rule mining

We modified the original definition by Rakesh Agarwal to suit our needs (Agrawal et al., 1993; Srikant and Agrawal, 1995) . The problem here is defined as: Let E=P:{e_1,e_2,e_3,...e_n } be a sequence of N children of a node P. Let A={a_1,a_2,a_3,...a_n } be the alignment set of the corresponding set E.

Let D=P:{ S_1,S_2,S_3,...S_m } be set consisting of all possible ordered sequence of children of the node P, Ex: S_1=S:{NP,VP,NP}, where S is the parent node and NP, VP and NP are its children. Each set in D has a unique ID, which represents the occurrence of the source order of the children. A rule is defined as an implication of the form X⇒Y where X⊆E and

[1] http://www-speech.sri.com/projects/srilm/
[2] Stanford Parser: http://nlp.stanford.edu/software/lex-parser.shtml

Y⊆Target Positions(E,A). The sets of items X and Y are called LHS and RHS of the rule. To illustrate the concepts, we use a simple example from the English-Hindi parallel corpus.

Consider the set of items I={Set of POS tags} $\bigcup$ {Set of Chunk tags}. For Example, I={NN,VBZ,NNS,NP,VP} and an example rule could be {NN,VBZ,NNS} $\Rightarrow$ {1,3,2}, which means that when NN, VBZ and NNS occur in a continuous pattern they are reordered to 1,3 and 2 positions respectively on the target side. The above example is a naive example. If we consider the training corpus with the alignments we could use constraints on various measures of significance. We use the best-known constraints, namely minimum threshold support and confidence. The support supp(X) of an itemset X is defined as the proportion of sentences which contain the itemset. The confidence of a rule is defined as

conf(X$\Rightarrow$Y)=supp(X$\bigcup$Y)/supp(X).

Association rules require language specific minimum support and minimum confidence at the same time. To achieve this, association rule learning is done in two steps. Firstly, minimum support is applied to find all frequent itemsets in the source language model. In the second step, these frequent itemsets and the minimum confidence constraints are used to generate rules from the word-aligned parallel corpus.

3.1 Frequent Pattern mining

For the first task of collecting the most frequent itemsets we used Fpgrowth algorithm [3] (Borgelt, 2005) implemented by Christian Borgelt. We used a POS and a chunk tag English language model. In a given parse tree the pattern model based on the order of pre-terminals is called POS language model and the pattern model based on the Non-terminals is called the Chunk language model. The below algorithm is run on every Non-terminal and pre-terminal node of a parse tree. In the modified version of mining frequent itemsets we also include generalization of the frequent sets, similar to the work done by (Chiang, 2005).

[3] http://www.borgelt.net/fpgrowth.html

Steps for extracting frequent LHSs: Consider X_1,X_2,X_3,X_4,...X_x are all possible children of a node S. The transaction here is the sequence of children of the node S. The sample example is shown in Fig 2.

1. Collect all occurrences of the children of a node and their frequencies from the transactions and name the set L_1.

2. Calculate L_2=L_1 $*$ L_1 which is the frequency set of two elements.

3. Similarly calculate L_n, till n = maximum possible children of parent S.

4. Once the maximum possible set is calculated, K-best frequent sets are collected and then elements which occur above a threshold(Θ) are combined to form a single element.
   Ex, most common patterns occurring as a children of NP are {JJ,NN,NN},{JJ,NN} etc.

5. The threshold was calculated based on various experiments, and then set to Θ=20% less than the frequency of least frequent itemset between the elements of the two L's.

   For example,
   L_3={JJ,NN}$_*${NN}={JJ,NN,NNP}.
   If freq{JJ,NN}=10, and freq{NNP}=20 and {JJ,NN,NNP}=9, Θ=10-(20% of 10)=8.
   So {JJ,NN} $\Rightarrow$ X_1.

   This way the generalized rules are learnt for all the tables (L_n, L_{n-1}..L_3). Using these generalized rules, the initial transactions are modified.

6. Recalculate L_1,L_2,..L_n based on the rules learnt above. Continue the process until no new rules are extracted at the end of the iteration.

3.2 Generate rules

The second problem is to generate association rules for these large itemsets with the constraints of minimal confidence. Suppose one of the large itemsets of a parent node S is L_k, L_k = P:{e_1,e_2,,e_k }, association rules with these itemsets are generated in the following way: Firstly a set P:{ e_1,e_2,..e_k } is

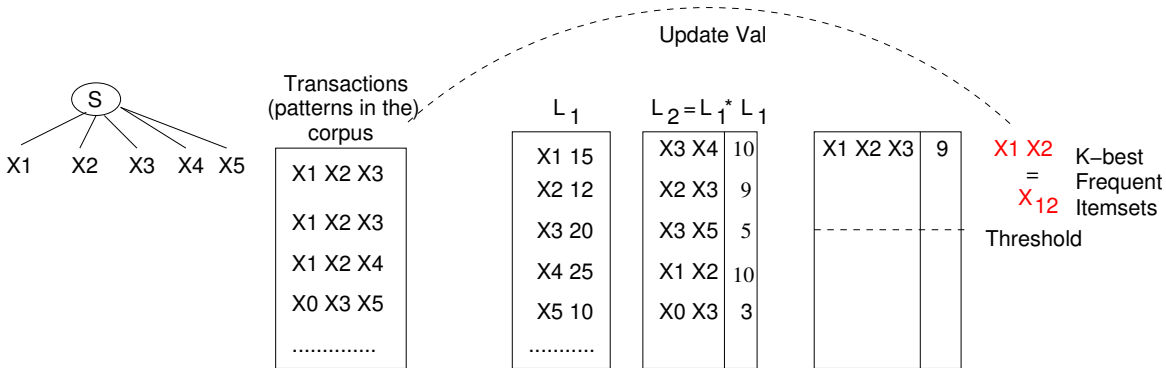

Figure 2: N-stage Generalization

matched with the source sequences of parent P and then their corresponding alignment information is used to generate the target sequence. The numbers on the rhs represent the position of the elements in the target sentence. Then by checking the constraint confidence this rule can be determined as interesting or not. Constraint confidence used here is the probability of occurrence of the non-monotone rule.

If c_1,c_2,c_3,c_4...c_x are the children of a Node X. LHS is the original order of the children. RHS is the sorted order of the children on the basis of Psn(T,Psn(S,c_i)), where $1 \leq i \leq x$.

From Fig 1, let us consider the top node and find the rule based on the head based method.

Suppose that given from the above frequency rule
L_k = S:{'PP' ',' 'ADVP' ',' 'NP' 'VP'}
Children(S) = 'PP' ',' 'ADVP' ',' 'NP' 'VP' '.'
The target positions are calculated as shown in the Table 1. RHS is calculated based on the target positions.

Table 1: Target Positions of Children(S)

Psn(T,'PP')	= Psn(T,1)	=6
Psn(T,',')	= Psn(T,4)	=7
Psn(T,'ADVP')	= Psn(T,5)	=1
Psn(T,',')	= Psn(T,6)	=2
Psn(T,'NP')	=Psn(T,7)	=8
Psn(T,'VP')	=Psn(T,8)	=18
Psn(T,'.')	= Psn(T,15)	=19

LHS = PP , ADVP , NP VP .
RHS = 3 4 1 2 5 6 7

3.2.1 Use of Generalization:

The above rule generated is the most commonly occurring phenomenon in English to Hindi machine translation. It is observed that adverbial phrase generally occurs at the beginning of the sentence on the Hindi side. The rule generated above will be captured less frequently because the exact pattern in LHS is rarely matched. Using the above generalization in frequent itemset mining we can merge all the most frequent occurring patterns into a common pattern.
The above example pattern is modified to the below using the generalization technique.

Rule: X1 ADVP , X2 $\Rightarrow$ 2 3 1 4

3.2.2 Rules and their Application

These generated rules are taken to calculate the probability of the non-monotone rules with respect to monotone rules. If the probability of the non-monotone rule was $\geq$0.5 then the rule was appended to the final list. The final list included all the generalized and non-generalized rules of different parent nodes.

The final list of rules is applied on both training and test corpus based on the longest possible sequence match. If the rule matches, then the source structures are reordered as per the rule. Specific rules are given more priority over the generalized rules.

4 Experiments

Table 2, Table 3 show some of the high frequency and generalized rules. The total number of rules learnt were 727 for a 11k training corpus. Number of generalizations learnt were 54.

Table 2: Most Frequent Rules

Rule	LHS	RHS
1	IN NP	2 1
2	NP VP NP	1 3 2
3	NP PP	2 1
4	VBG PP	2 1
5	VBZ ADVP NP	2 3 1

Table 3: Generalized Rules

Rule	LHS	RHS
1	X_1 ADVP , X_2	2 3 1 4
2	X_3 VBZ‖VBG X_4	1 3 2
3	ADVP X_5 .	2 1 3
4	MD RB X_6	3 1 2
5	VB X_7 NP-TMP	2 3 1

Once the training and test sentences are reordered using the above rules, they are fed to the Moses system. It is clear that without reordering the performace of the system is worst. Training and test data consisted of 11,300 and 500 sentences respectively.

Table 4: Evaluation on Moses

Config	Blue Score	NIST
Moses Without Reorder	0.2123	5.5315
Moses + Our Reorder	0.2329	5.6605
Moses With Reorder	0.2475	5.7069

5 Discussion

Our method showed a drop in terms of blue score as compared to Moses reordering; this is probably due to the reordering based on lexicalized rules in Moses. The above generalization works effectively in case of the Stanford parser as it stitches the nodes at top level. English-Hindi tourism corpus distributed as a part of ICON 2008 shared task. Our learning based on phrase structure doesn't handle the movement of children across nodes. Whereas, dependency structure based rule learning would help in handling more constructs in terms of word-level reordering patterns. Some of the least frequent patterns are actually interesting patterns in terms of reordering. Learning these kinds of patterns would be a challenging task.

6 Future Work

Work has to be done in terms of prioritization of the rules, for example first priority should be given to more specific rules (the one with constraints) then to the general rules. More constraints with respect to morphological features would also help in improving the diversity of the rules. We will also look into the linguistic clause based reordering features which would help in reordering of distant pair of languages. Manual evaluation of the output will throw some light on the effectiveness of this system. To further evaluate the approach we would also try the approach on someother distant language pairs.

References

Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. 1993. Mining association rules between sets of items in large databases. In *SIGMOD '93: Proceedings of the 1993 ACM SIGMOD international conference on Management of data*, pages 207–216, New York, NY, USA. ACM.

Alfred V. Aho and Jeffrey D. Ullman. 1972. *The Theory of Parsing, Translation and Compiling*, volume 1. Prentice-Hall, Englewood Cliffs, NJ.

Christian Borgelt. 2005. An implementation of the fp-growth algorithm. In *OSDM '05: Proceedings of the 1st international workshop on open source data mining*, pages 1–5, New York, NY, USA. ACM.

Peter F. Brown, John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Fredrick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin. 1990. A statistical approach to machine translation. *COMPUTATIONAL LINGUISTICS*, 16(2):79–85.

Eugene Charniak, Kevin Knight, and Kenji Yamada. 2003. Syntax-based language models for statistical machine translation. In *MT Summit IX. Intl. Assoc. for Machine Translation*.

David Chiang. 2005. A hierarchical phrase-based model for statistical machine translation. In *In ACL*, pages 263–270.

Michael Collins, Philipp Koehn, and Ivona Kučerová. 2005. Clause restructuring for statistical machine translation. In *ACL '05: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 531–540, Morristown, NJ, USA. Association for Computational Linguistics.

Michael Collins. 2003. Head-driven statistical models for natural language parsing. Technical report.

Marta R. Costa-jussà and José A. R. Fonollosa. 2006. Statistical machine reordering. In *EMNLP '06: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 70–76, Morristown, NJ, USA. Association for Computational Linguistics.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Ramakrishnan Srikant and Rakesh Agrawal. 1995. Mining generalized association rules. In *Research Report RJ 9963, IBM Almaden Research*.

Fei Xia and Michael McCord. 2004. Improving a statistical mt system with automatically learned rewrite patterns. In *COLING '04: Proceedings of the 20th international conference on Computational Linguistics*, page 508, Morristown, NJ, USA. Association for Computational Linguistics.

Kenji Yamada and Kevin Knight. 2001. A syntax-based statistical translation model. In *ACL '01: Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, pages 523–530, Morristown, NJ, USA. Association for Computational Linguistics.