Ordering Phrases with Function Words

Hendra Setiawan and **Min-Yen Kan**
School of Computing
National University of Singapore
Singapore 117543
{hendrase,kanmy}@comp.nus.edu.sg

Haizhou Li
Institute for Infocomm Research
21 Heng Mui Keng Terrace
Singapore 119613
hli@i2r.a-star.edu.sg

Abstract

This paper presents a Function Word centered, Syntax-based (FWS) solution to address phrase ordering in the context of statistical machine translation (SMT). Motivated by the observation that function words often encode grammatical relationship among phrases within a sentence, we propose a probabilistic synchronous grammar to model the ordering of function words and their left and right arguments. We improve phrase ordering performance by lexicalizing the resulting rules in a small number of cases corresponding to function words. The experiments show that the FWS approach consistently outperforms the baseline system in ordering function words' arguments and improving translation quality in both perfect and noisy word alignment scenarios.

1 Introduction

The focus of this paper is on function words, a class of words with little intrinsic meaning but is vital in expressing grammatical relationships among words within a sentence. Such encoded grammatical information, often implicit, makes function words pivotal in modeling structural divergences, as projecting them in different languages often result in long-range structural changes to the realized sentences.

Just as a foreign language learner often makes mistakes in using function words, we observe that current machine translation (MT) systems often perform poorly in ordering function words' arguments; lexically correct translations often end up reordered incorrectly. Thus, we are interested in modeling the structural divergence encoded by such function words. A key finding of our work is that modeling the ordering of the dependent arguments of function words results in better translation quality.

Most current systems use statistical knowledge obtained from corpora in favor of rich natural language knowledge. Instead of using syntactic knowledge to determine function words, we approximate this by equating the most frequent words as function words. By explicitly modeling phrase ordering around these frequent words, we aim to capture the most important and prevalent ordering productions.

2 Related Work

A good translation should be both faithful with adequate lexical choice to the source language and fluent in its word ordering to the target language. In pursuit of better translation, phrase-based models (Och and Ney, 2004) have significantly improved the quality over classical word-based models (Brown et al., 1993). These multiword phrasal units contribute to fluency by inherently capturing intra-phrase reordering. However, despite this progress, inter-phrase reordering (especially long distance ones) still poses a great challenge to statistical machine translation (SMT).

The basic phrase reordering model is a simple unlexicalized, context-insensitive distortion penalty model (Koehn et al., 2003). This model assumes little or no structural divergence between language pairs, preferring the original, translated order by penalizing reordering. This simple model works well when properly coupled with a well-trained language

Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pages 712–719,
Prague, Czech Republic, June 2007.

model, but is otherwise impoverished without any lexical evidence to characterize the reordering.

To address this, lexicalized context-sensitive models incorporate contextual evidence. The local prediction model (Tillmann and Zhang, 2005) models structural divergence as the relative position between the translation of two neighboring phrases. Other further generalizations of orientation include the global prediction model (Nagata et al., 2006) and distortion model (Al-Onaizan and Papineni, 2006).

However, these models are often fully lexicalized and sensitive to individual phrases. As a result, they are not robust to unseen phrases. A careful approximation is vital to avoid data sparseness. Proposals to alleviate this problem include utilizing bilingual phrase cluster or words at the phrase boundary (Nagata et al., 2006) as the phrase identity.

The benefit of introducing lexical evidence without being fully lexicalized has been demonstrated by a recent state-of-the-art *formally* syntax-based model[1], Hiero (Chiang, 2005). Hiero performs phrase ordering by using linked non-terminal symbols in its synchronous CFG production rules coupled with lexical evidence. However, since it is difficult to specify a well-defined rule, Hiero has to rely on weak heuristics (i.e., length-based thresholds) to extract rules. As a result, Hiero produces grammars of enormous size. Watanabe et al. (2006) further reduces the grammar's size by enforcing all rules to comply with Greibach Normal Form.

Taking the lexicalization an intuitive a step forward, we propose a novel, finer-grained solution which models the content and context information encoded by function words - approximated by high frequency words. Inspired by the success of syntax-based approaches, we propose a synchronous grammar that accommodates gapping production rules, while focusing on the statistical modeling in relation to function words. We refer to our approach as the Function Word-centered Syntax-based approach (FWS). Our FWS approach is different from Hiero in two key aspects. First, we use only a small set of high frequency lexical items to lexicalize non-terminals in the grammar. This results in a much smaller set of rules compared to Hiero,

[1] Chiang (2005) used the term "formal" to indicate the use of synchronous grammar but without linguistic commitment

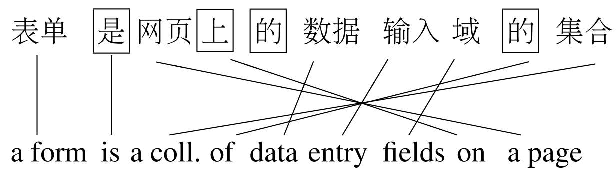

Figure 1: A Chinese-English sentence pair.

greatly reducing the computational overhead that arises when moving from phrase-based to syntax-based approach. Furthermore, by modeling only high frequency words, we are able to obtain reliable statistics even in small datasets. Second, as opposed to Hiero, where phrase ordering is done implicitly alongside phrase translation and lexical weighting, we directly model the reordering process using orientation statistics.

The FWS approach is also akin to (Xiong et al., 2006) in using a synchronous grammar as a reordering constraint. Instead of using Inversion Transduction Grammar (ITG) (Wu, 1997) directly, we will discuss an ITG extension to accommodate gapping.

3 Phrase Ordering around Function Words

We use the following Chinese (c) to English (e) translation in Fig.1 as an illustration to conduct an inquiry to the problem. Note that the sentence translation requires some translations of English words to be ordered far from their original position in Chinese. Recovering the correct English ordering requires the inversion of the Chinese postpositional phrase, followed by the inversion of the first smaller noun phrase, and finally the inversion of the second larger noun phrase. Nevertheless, the correct ordering can be recovered if the position and the semantic roles of the arguments of the boxed function words were known. Such a function word centered approach also hinges on knowing the correct phrase boundaries for the function words' arguments and which reorderings are given precedence, in case of conflicts.

We propose modeling these sources of knowledge using a statistical formalism. It includes 1) a model to capture bilingual orientations of the left and right arguments of these function words; 2) a model to approximate correct reordering sequence; and 3) a model for finding constituent boundaries of the left and right arguments. Assuming that the most frequent words in a language are function words, we can apply orientation statistics associated with these words to reorder their adjacent left and right neighbors. We follow the notation in (Nagata et al., 2006) and define the following bilingual orientation values given two neighboring source (Chinese) phrases: Monotone-Adjacent (MA); Reverse-Adjacent (RA); Monotone-Gap (MG); and Reverse-Gap (RG). The first clause (monotone, reverse) indicates whether the target language translation order follows the source order; the second (adjacent, gap) indicates whether the source phrases are adjacent or separated by an intervening phrase on the target side.

Table 1 shows the orientation statistics for several function words. Note that we separate the statistics for left and right arguments to account for differences in argument structures: some function words take a single argument (e.g., prepositions), while others take two or more (e.g., copulas). To handle other reordering decisions not explicitly encoded (i.e., lexicalized) in our FWS model, we introduce a universal token $\mathcal{U}$, to be used as a backoff statistic when function words are absent.

For example, orientation statistics for 是 (to be) overwhelmingly suggests that the English translation of its surrounding phrases is identical to its Chinese ordering. This reflects the fact that the arguments of copulas in both languages are realized in the same order. The orientation statistics for postposition 上 (on) suggests inversion which captures the divergence between Chinese postposition to the English preposition. Similarly, the dominant orientation for particle 的 (of) suggests the noun-phrase shift from modified-modifier to modifier-modified, which is common when translating Chinese noun phrases to English.

Taking all parts of the model, which we detail later, together with the knowledge in Table 1, we demonstrate the steps taken to translate the example in Fig. 2. We highlight the function words with boxed characters and encapsulate content words as indexed symbols. As shown, orientation statistics from function words alone are adequate to recover the English ordering - in practice, content words also influence the reordering through a language model. One can think of the FWS approach as a foreign language learner with limited knowledge about Chinese grammar but fairly knowledgable about the role of Chinese function words.

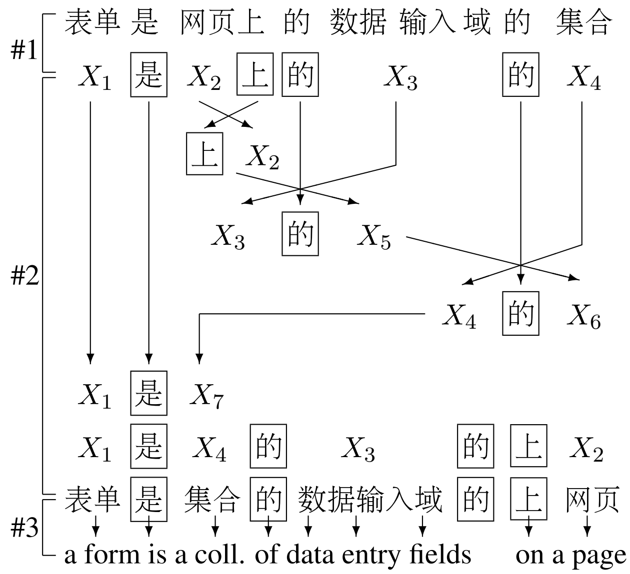

Figure 2: In Step 1, function words (boxed characters) and content words (indexed symbols) are identified. Step 2 reorders phrases according to knowledge embedded in function words. A new indexed symbol is introduced to indicate previously reordered phrases for conciseness. Step 3 finally maps Chinese phrases to their English translation.

4 The FWS Model

We first discuss the extension of standard ITG to accommodate gapping and then detail the statistical components of the model later.

4.1 Single Gap ITG (SG-ITG)

The FWS model employs a synchronous grammar to describe the admissible orderings.

The utility of ITG as a reordering constraint for most language pairs, is well-known both empirically (Zens and Ney, 2003) and analytically (Wu, 1997), however ITG's *straight* (monotone) and *inverted* (reverse) rules exhibit strong cohesiveness, which is inadequate to express orientations that require gaps. We propose SG-ITG that follows Wellington et al. (2006)'s suggestion to model at most one gap.

We show the rules for SG-ITG below. Rules 1-3 are identical to those defined in standard ITG, in which monotone and reverse orderings are represented by square and angle brackets, respectively.

Rank	Word	unigram	MA_L	RA_L	MG_L	RG_L	MA_R	RA_R	MG_R	RG_R
1	的	0.0580	0.45	**0.52**	0.01	0.02	0.44	**0.52**	0.01	0.03
2	，	0.0507	**0.85**	0.12	0.02	0.01	**0.84**	0.12	0.02	0.02
3	。	0.0550	**0.99**	0.01	0.00	0.00	**0.92**	0.08	0.00	0.00
4	”	0.0155	**0.87**	0.10	0.02	0.00	**0.82**	0.12	0.05	0.02
5	“	0.0153	**0.84**	0.11	0.01	0.04	**0.88**	0.11	0.01	0.01
6	和	0.0138	**0.95**	0.02	0.01	0.01	**0.97**	0.02	0.01	0.00
7	任务	0.0123	**0.73**	0.12	0.10	0.04	**0.51**	0.14	0.14	0.20
8	可以	0.0114	**0.78**	0.12	0.03	0.07	**0.86**	0.05	0.08	0.01
9	或	0.0099	**0.95**	0.02	0.02	0.01	**0.96**	0.01	0.02	0.01
10	将	0.0091	**0.87**	0.10	0.01	0.02	**0.88**	0.10	0.01	0.00
21	是	0.0056	**0.85**	0.11	0.02	0.02	**0.85**	0.04	0.09	0.02
37	上	0.0035	0.33	**0.65**	0.02	0.01	0.31	**0.63**	0.03	0.03
-	$\mathcal{U}$	0.0002	**0.76**	0.14	0.06	0.05	**0.74**	0.13	0.07	0.06

Table 1: Orientation statistics and unigram probability of selected frequent Chinese words in the HIT corpus. Subscripts L/R refers to lexical unit's orientation with respect to its left/right neighbor. $\mathcal{U}$ is the universal token used in back-off for $N = 128$. Dominant orientations of each word are in **bold**.

(1) $X \rightarrow c/e$
(2) $X \rightarrow [XX]$ (3) $X \rightarrow \langle XX \rangle$
(4) $X^{\diamond} \rightarrow [X \diamond X]$ (5) $X^{\diamond} \rightarrow \langle X \diamond X \rangle$
(6) $X \rightarrow [X * X]$ (7) $X \rightarrow \langle X * X \rangle$

SG-ITG introduces two new sets of rules: gapping (Rules 4-5) and dovetailing (Rules 6-7) that deal specifically with gaps. On the RHS of the gapping rules, a diamond symbol ($\diamond$) indicates a gap, while on the LHS, it emits a superscripted symbol $X^{\diamond}$ to indicate a gapped phrase (plain Xs without superscripts are thus contiguous phrases). Gaps in $X^{\diamond}$ are eventually filled by actual phrases via dovetailing (marked with an $*$ on the RHS).

Fig.3 illustrates gapping and dovetailing rules using an example where two Chinese adjectival phrases are translated into a single English subordinate clause. SG-ITG can generate the correct ordering by employing gapping followed by dovetailing, as shown in the following simplified trace:

$X_1^{\diamond} \rightarrow \langle$ 1997 $\boxed{\text{的}}$ 第一版, **V.1** $\diamond$ **1997** $\rangle$
$X_2^{\diamond} \rightarrow \langle$ 1998 $\boxed{\text{的}}$ 第二版, **V.2** $\diamond$ **1998** $\rangle$
$X_3 \rightarrow [X_1 * X_2]$
$\rightarrow$ [1997 $\boxed{\text{的}}$ 第一版 $\boxed{\text{和}}$ 1998 $\boxed{\text{的}}$ 第二版,
V.1 $\diamond$ **1997** $*$ **V.2** $\diamond$ **1998**]
$\rightarrow$ 1997 的第一版和1998 的第二版,
V.1 and **V.2** that were released in **1997** and **1998**

where $X_1^{\diamond}$ and $X_2^{\diamond}$ each generate the translation of their respective Chinese noun phrase using gapping and X_3 generates the English subclause by dovetailing the two gapped phrases together.

Thus far, the grammar is unlexicalized, and does not incorporate any lexical evidence. Now we modify the grammar to introduce lexicalized function words to SG-ITG. In practice, we introduce a new set of lexicalized non-terminal symbols Y_i, $i \in \{1...N\}$, to represent the top N most-frequent words in the vocabulary; the existing unlexicalized X is now reserved for content words. This difference does not inherently affect the structure of the grammar, but rather lexicalizes the statistical model.

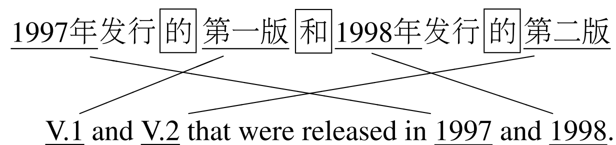

Figure 3: An example of an alignment that can be generated only by allowing gaps.

In this way, although different Y_is follow the same production rules, they are associated with different statistics. This is reflected in Rules 8-9. Rule 8 emits the function word; Rule 9 reorders the arguments around the function word, resembling our orientation model (see Section 4.2) where a function word influences the orientation of its left and right arguments. For clarity, we omit notation that denotes which rules have been applied (monotone, reverse; gapping, dovetailing).

(8) $Y_i \rightarrow c/e$ (9) $X \rightarrow XY_iX$

In practice, we replace Rule 9 with its equivalent 2-normal form set of rules (Rules 10-13). Finally, we introduce rules to handle back-off (Rules 14-16) and upgrade (Rule 17). These allow SG-ITG to revert function words to normal words and vice versa.

(10)	$R \rightarrow Y_i X$	(11)	$L \rightarrow X Y_i$
(12)	$X \rightarrow LX$	(13)	$X \rightarrow XR$
(14)	$Y_i \rightarrow X$	(15)	$R \rightarrow X$
(16)	$L \rightarrow X$	(17)	$X \rightarrow Y_{\mathcal{U}}$

Back-off rules are needed when the grammar has to reorder two adjacent function words, where one set of orientation statistics must take precedence over the other. The example in Fig.1 illustrates such a case where the orientation of 上 (on) and 的 (of) compete for influence. In this case, the grammar chooses to use 的 (of) and reverts the function word 上 (on) to the unlexicalized form.

The upgrade rule is used for cases where there are two adjacent phrases, both of which are not function words. Upgrading allows either phrase to act as a function word, making use of the universal word's orientation statistics to reorder its neighbor.

4.2 Statistical model

We now formulate the FWS model as a statistical framework. We replace the deterministic rules in our SG-ITG grammar with probabilistic ones, elevating it to a stochastic grammar. In particular, we develop the three sub models (see Section 3) which influence the choice of production rules for ordering decision. These models operate on the 2-norm rules, where the RHS contains one function word and its argument (except in the case of the phrase boundary model). We provide the intuition for these models next, but their actual form will be discussed in the next section on training.

1) Orientation Model *ori(o|H,Y_i)***:** This model captures the preference of a function word Y_i to a particular orientation $o \in \{MA, RA, MG, RG\}$ in reordering its $H \in \{left, right\}$ argument X. The parameter H determines which set of Y_i's statistics to use (left or right); the model consults Y_i's left orientation statistic for Rules 11 and 13 where X precedes Y_i, otherwise Y_i's right orientation statistic is used for Rules 10 and 12.

2) Preference Model *pref(Y_i)***:** This model arbitrates reordering in the cases where two function words are adjacent and the backoff rules have to decide which function word takes precedence, reverting the other to the unlexicalized X form. This model prefers the function word with higher unigram probability to take the precedence.

3) Phrase Boundary Model *pb(X)***:** This model is a penalty-based model, favoring the resulting alignment that conforms to the source constituent boundary. It penalizes Rule 1 if the terminal rule X emits a Chinese phrase that violates the boundary ($pb = e^{-1}$), otherwise it is inactive ($pb = 1$).

These three sub models act as features alongside seven other standard SMT features in a log-linear model, resulting in the following set of features $\{f_1, \ldots, f_{10}\}$: f_1) orientation $ori(o|H, Y_i)$; f_2) preference $pref(Y_i)$; f_3) phrase boundary $pb(X)$; f_4) language model $lm(e)$; $f_5 - f_6$) phrase translation score $\phi(e|c)$ and its inverse $\phi(c|e)$; $f_7 - f_8$) lexical weight $lex(e|c)$ and its inverse $lex(c|e)$; f_9) word penalty wp; and f_{10}) phrase penalty pp.

The translation is then obtained from the most probable derivation of the stochastic SG-ITG. The formula for a single derivation is shown in Eq. (18), where $X_1, X_2, ..., X_L$ is a sequence of rules with $w(X_l)$ being the weight of each particular rule X_l. $w(X_l)$ is estimated through a log-linear model, as in Eq. (19), with all the abovementioned features where λ_j reflects the contribution of each feature f_j.

$$(18) \quad P(X_1, ..., X_L) = \prod_{l=1}^{L} w(X_l)$$

$$(19) \quad w(X_l) = \prod_{j=1}^{10} f_j(X_l)^{\lambda_j}$$

5 Training

We train the orientation and preference models from statistics of a training corpus. To this end, we first derive the event counts and then compute the relative frequency of each event. The remaining phrase boundary model can be modeled by the output of a standard text chunker, as in practice it is simply a constituent boundary detection mechanism together with a penalty scheme.

The events of interest to the orientation model are (Y_i, o) tuples, where $o \in \{MA, RA, MG, RG\}$ is an orientation value of a particular function word Y_i. Note that these tuples are not directly observable from training data. Hence, we need an algorithm to derive (Y_i, o) tuples from a parallel corpus. Since both left and right statistics share identical training steps, thus we omit references to them.

The algorithm to derive (Y_i, o) involves several steps. First, we estimate the bi-directional alignment by running GIZA++ and applying the "grow-diag-final" heuristic. Then, the algorithm enumerates all Y_i and determines its orientation o with respect to its argument X to derive (Y_i, o). To determine o, the algorithm inspects the monotonicity (monotone or reverse) and adjacency (adjacent or gap) between Y_i's and X's alignments.

Monotonicity can be determined by looking at the Y_i's alignment with respect to the most fine-grained level of X (i.e., word level alignment). However, such a heuristic may inaccurately suggest gap orientation. Figure 1 illustrates this problem when deriving the orientation for the second 的 (of). Looking only at the word alignment of its left argument 域 (fields) incorrectly suggests a gapped orientation, where the alignment of 数据输入 (data entry) intervened. It is desirable to look at the alignment of 数据输入域 (data entry fields) at the phrase level, which suggests the correct adjacent orientation instead.

To address this issue, the algorithm uses gapping conservatively by utilizing the consistency constraint (Och and Ney, 2004) to suggest phrase level alignment of X. The algorithm exhaustively grows consistent blocks containing the most fine-grained level of X not including Y_i. Subsequently, it merges each hypothetical argument with the Y_i's alignment. The algorithm decides that Y_i has a gapped orientation only if all merged blocks violate the consistency constraint, concluding an adjacent orientation otherwise.

With the event counts $C(Y_i, o)$ of tuple (Y_i, o), we estimate the orientation model for Y_i and $\mathcal{U}$ using Eqs. (20) and (21). We also estimate the preference model with word unigram counts $C(Y_i)$ using Eqs. (22) and (23), where V indicates the vocabulary size.

$$
\begin{aligned}
&(20)\ ori(o|Y_i) &=& \ C(Y_i, o)/C(Y_i, \cdot), i \leqslant N \\
&(21)\ ori(o|\mathcal{U}) &=& \ \sum_{i>N} C(Y_i, o)/\sum_{i>N} C(Y_i, \cdot) \\
&(22)\ pref(Y_i) &=& \ C(Y_i)/C(\cdot), i \leqslant N \\
&(23)\ pref(\mathcal{U}) &=& \ 1/(V-N)\sum_{i>N} C(Y_i)/C(\cdot)
\end{aligned}
$$

Samples of these statistics are found in Table 1 and have been used in the running examples. For instance, the statistic $ori(RA_L|\text{的}) = 0.52$, which is the dominant one, suggests that the grammar inversely order 的(of)'s left argument; while in our illustration of backoff rules in Fig.1, the grammar chooses 的(of) to take precedence since $pref(\text{的}) > pref(\text{上})$.

6 Decoding

We employ a bottom-up CKY parser with a beam to find the derivation of a Chinese sentence which maximizes Eq. (18). The English translation is then obtained by post-processing the best parse.

We set the beam size to 30 in our experiment and further constrain reordering to occur within a window of 10 words. Our decoder also prunes entries that violate the following constraints: 1) each entry contains at most one gap; 2) any gapped entries must be dovetailed at the next level higher; 3) an entry spanning the whole sentence must not contain gaps.

The score of each newly-created entry is derived from the scores of its parts accordingly. When scoring entries, we treat gapped entries as contiguous phrases by ignoring the gap symbol and rely on the orientation model to penalize such entries. This allows a fair score comparison between gapped and contiguous entries.

7 Experiments

We would like to study how the FWS model affects 1) the ordering of phrases around function words; 2) the overall translation quality. We achieve this by evaluating the FWS model against a baseline system using two metrics, namely, orientation accuracy and BLEU respectively.

We define the orientation accuracy of a (function) word as the accuracy of assigning correct orientation values to both its left and right arguments. We report the aggregate for the top 1024 most frequent words; these words cover 90% of the test set.

We devise a series of experiments and run it in two scenarios - manual and automatic alignment - to assess the effects of using perfect or real-world input. We utilize the HIT bilingual computer manual corpus, which has been manually aligned, to perform Chinese-to-English translation (see Table 2). Manual alignment is essential as we need to measure orientation accuracy with respect to a gold standard.

		Chinese	English
train	words	145,731	135,032
(7K sentences)	vocabulary	5,267	8,064
dev	words	13,986	14,638
(1K sentences)	untranslatable	486 (3.47%)	
test	words	27,732	28,490
(2K sentences)	untranslatable	935 (3.37%)	

Table 2: Statistics for the HIT corpus.

A language model is trained using the SRILM-Toolkit, and a text chunker (Chen et al., 2006) is applied to the Chinese sentences in the test and dev sets to extract the constituent boundaries necessary for the phrase boundary model. We run minimum error rate training on dev set using Chiang's toolkit to find a set of parameters that optimizes BLEU score.

7.1 Perfect Lexical Choice

Here, the task is simplified to recovering the correct order of the English sentence from the scrambled Chinese order. We trained the orientation model using manual alignment as input. The aforementioned decoder is used with phrase translation, lexical mapping and penalty features turned off.

Table 4 compares orientation accuracy and BLEU between our FWS model and the baseline. The baseline (lm+d) employs a language model and distortion penalty features, emulating the standard Pharaoh model. We study the behavior of the FWS model with different numbers of lexicalized items N. We start with the language model alone (N=0) and incrementally add the orientation (+ori), preference (+ori+pref) and phrase boundary models (+ori+pref+pb).

As shown, the language model alone is relatively weak, assigning the correct orientation in only 62.28% of the cases. A closer inspection reveals that the lm component aggressively promotes reverse reorderings. Including a distortion penalty model (the baseline) improves the accuracy to 72.55%. This trend is also apparent for the BLEU score.

When we incorporate the FSW model, including just the most frequent word (Y_1=的), we see improvement. This model promotes non-monotone reordering conservatively around Y_1 (where the dominant statistic suggests reverse ordering). Increasing the value of N leads to greater improvement. The most effective improvement is obtained by increasing N to 128. Additional (marginal) improvement is obtained at the expense of modeling an additional 900+ lexical items. We see these results as validating our claim that modeling the top few most frequent words captures most important and prevalent ordering productions.

pharaoh (dl=5)	22.44 ± 0.94
+ori	23.80 ± 0.98
+ori+pref	23.85 ± 1.00
+ori+pref+pb	23.86 ± 1.08

Table 3: BLEU score with the 95% confidence intervals based on (Zhang and Vogel, 2004). All improvement over the baseline (row 1) are statistically significant under paired bootstrap resampling.

Lastly, we study the effect of the pref and pb features. The inclusion of both sub models has little affect on orientation accuracy, but it improves BLEU consistently (although not significantly). This suggests that both models correct the mistakes made by the ori model while preserving the gain. They are not as effective as the addition of the basic orientation model as they only play a role when two lexicalized entries are adjacent.

7.2 Full SMT experiments

Here, all knowledge is automatically trained on the train set, and as a result, the input word alignment is noisy. As a baseline, we use the state-of-the-art phrase-based Pharaoh decoder. For a fair comparison, we run minimum error rate training for different distortion limits from 0 to 10 and report the best parameter (dl=5) as the baseline.

We use the phrase translation table from the baseline and perform an identical set of experiments as the perfect lexical choice scenario, except that we only report the result for N=128, due to space constraint. Table 3 reports the resulting BLEU scores.

As shown, the FWS model improves BLEU score significantly over the baseline. We observe the same trend as the one in perfect lexical choice scenario where top 128 most frequent words provides the majority of improvement. However, the pb features yields no noticeable improvement unlike in prefect lexical choice scenario; this is similar to the findings in (Koehn et al., 2003).

		N=0	N=1	N=4	N=16	N=64	N=128	N=256	N=1024
Orientation Acc. (%)	lm+d	72.55							
	+ori	62.28	76.52	76.58	77.38	77.54	**78.17**	77.76	*78.38*
	+ori+pref		76.66	76.82	77.57	77.74	**78.13**	77.94	*78.54*
	+ori+pref+pb		76.70	76.85	77.58	77.70	**78.20**	77.94	*78.56*
BLEU	lm+d	75.13							
	+ori	66.54	77.54	77.57	78.22	78.48	**78.76**	78.58	*79.20*
	+ori+pref		77.60	77.70	78.29	78.65	**78.77**	78.70	*79.30*
	+ori+pref+pb		77.69	77.80	78.34	78.65	**78.93**	78.79	*79.30*

Table 4: Results using perfect aligned input. Here, (lm+d) is the baseline; (+ori), (+ori+pref) and (+ori+pref+pb) are different FWS configurations. The results of the model (where N is varied) that features the largest gain are **bold**, whereas the highest score is *italicized*.

8 Conclusion

In this paper, we present a statistical model to capture the grammatical information encoded in function words. Formally, we develop the Function Word Syntax-based (FWS) model, a probabilistic synchronous grammar, to encode the orientation statistics of arguments to function words. Our experimental results shows that the FWS model significantly improves the state-of-the-art phrase-based model.

We have touched only the surface benefits of modeling function words. In particular, our proposal is limited to modeling function words in the source language. We believe that conditioning on both source and target pair would result in more fine-grained, accurate orientation statistics.

From our error analysis, we observe that 1) reordering may span several levels and the preference model does not handle this phenomena well; 2) correctly reordered phrases with incorrect boundaries severely affects BLEU score and the phrase boundary model is inadequate to correct the boundaries especially for cases of long phrase. In future, we hope to address these issues while maintaining the benefits offered by modeling function words.

References

Benjamin Wellington, Sonjia Waxmonsky, and I. Dan Melamed. 2006. Empirical Lower Bounds on the Complexity of Translational Equivalence. In *ACL/COLING 2006*, pp. 977–984.

Christoph Tillman and Tong Zhang. 2005. A Localized Prediction Model for Statistical Machine Translation. In *ACL 2005*, pp. 557–564.

David Chiang. 2005. A Hierarchical Phrase-Based Model for Statistical Machine Translation. In *ACL 2005*, pp. 263–270.

Dekai Wu. 1997. Stochastic Inversion Transduction Grammars and Bilingual Parsing of Parallel Corpora. *Computational Linguistics*, 23(3):377–403.

Deyi Xiong, Qun Liu, and Shouxun Lin. 2006. Maximum Entropy Based Phrase Reordering Model for Statistical Machine Translation. In *ACL/COLING 2006*, pp. 521–528.

Franz J. Och and Hermann Ney. 2004. The Alignment Template Approach to Statistical Machine Translation. *Computational Linguistics*, 30(4):417–449.

Masaaki Nagata, Kuniko Saito, Kazuhide Yamamoto, and Kazuteru Ohashi. 2006. A Clustered Global Phrase Reordering Model for Statistical Machine Translation. In *ACL/COLING 2006*, pp. 713–720.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, Robert L. Mercer 1993. The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics*, 19(2):263–311.

Philipp Koehn, Franz J. Och, and Daniel Marcu. 2003. Statistical Phrase-Based Translation. In *HLT-NAACL 2003*, pp. 127–133.

Richard Zens and Hermann Ney. 2003. A Comparative Study on Reordering Constraints in Statistical Machine Translation. In *ACL 2003*.

Taro Watanabe, Hajime Tsukada, and Hideki Isozaki. 2006. Left-to-Right Target Generation for Hierarchical Phrase-Based Translation. In *ACL/COLING 2006*, pp. 777–784.

Wenliang Chen, Yujie Zhang and Hitoshi Isahara 2006. An Empirical Study of Chinese Chunking In *ACL 2006 Poster Sessions*, pp. 97–104.

Yaser Al-Onaizan and Kishore Papineni. 2006. Distortion Models for Statistical Machine Translation. In *ACL/COLING 2006*, pp. 529–536.

Ying Zhang and Stephan Vogel. 2004. Measuring Confidence Intervals for the Machine Translation Evaluation Metrics. In *TMI 2004*.