To appear in *TMI-95, Sixth International Conf. on Theoretical and Methodological Issues in Machine Translation*, Leuven, Belgium: Jul. 95.

Grammarless Extraction of Phrasal Translation Examples from Parallel Texts

Dekai Wu[*]

Technical Report HKUST-CS95-23
May 1995

[*]*HKUST*
Department of Computer Science
University of Science & Technology
Clear Water Bay, Hong Kong
`dekai@cs.ust.hk`

Abstract

We describe a method for identifying subsentential phrasal translation examples in sentence-aligned parallel corpora, using only a probabilistic translation lexicon for the language pair. Our method differs from previous approaches in that (1) it is founded on a formal basis, making use of an *inversion transduction grammar* (ITG) formalism that we recently developed for bilingual language modeling, and (2) it requires no language-specific monolingual grammars for the source and target languages. Instead, we devise a generic, language-independent constituent-matching ITG with inherent expressiveness properties that correspond to a desirable level of matching flexibility. Bilingual parsing, in conjunction with a stochastic version of the ITG formalism, performs the phrasal translation extraction.

The Hong Kong
University of Science & Technology

Technical Report Series
Department of Computer Science

1 Introduction

Phrasal translation examples at the subsentential level are an essential resource for many MT and MAT architectures. This requirement is becoming increasingly direct for the example-based machine translation paradigm (Nagao 1984), whose translation flexibility is strongly restricted if the examples are only at the sentential level. It can now be assumed that a parallel bilingual corpus may be aligned to the sentence level with reasonable accuracy (Kay & Röscheisen 1988; Catizone *et al.* 1989; Gale & Church 1991; Brown *et al.* 1991; Chen 1993), even for languages as disparate as Chinese and English (Wu 1994). Algorithms for subsentential alignment have been developed as well at granularities of the character (Church 1993), word (Dagan *et al.* 1993; Fung & Church 1994; Fung & McKeown 1994), collocation (Smadja 1992), and specially-segmented (Kupiec 1993) levels. However, the identification of subsentential, nested, phrasal translations within the parallel texts remains a non-trivial problem, due to the added complexity of dealing with constituent structure. Manual phrasal matching is feasible only for small corpora, either for toy-prototype testing or for narrowly-restricted applications.

Automatic approaches to identification of subsentential translation units have largely followed what we might call a "parse-parse-match" procedure. Each half of the parallel corpus is first parsed individually using a monolingual grammar. Subsequently, the constituents of each sentence-pair are matched according to some heuristic procedure. A number of recent proposals can be cast in this framework (Sadler & Vendelmans 1990; Kaji *et al.* 1992; Matsumoto *et al.* 1993; Cranias *et al.* 1994; Grishman 1994).

The "parse-parse-match" procedure is susceptible to three weaknesses:

- *Appropriate, robust, monolingual grammars may not be available.* This condition is particularly relevant for many non-Western-European languages such as Chinese. A grammar for this purpose must be robust since it must still identify constituents for the subsequent matching process even for unanticipated or ill-formed input sentences.

- *The grammars may be incompatible across languages.* The best-matching constituent types between the two languages may not include the same core arguments. While grammatical differences can make this problem unavoidable, there is often a degree of arbitrariness in a grammar's chosen set of syntactic categories, particularly if the grammar is designed to be robust. The mismatch can be exacerbated when the monolingual grammars are designed independently, or under different theoretical considerations.

- *Selection between multiple possible arrangements may be arbitrary.* By an "arrangement" between any given pair of sentences from the parallel corpus, we mean a set of matchings between the constituents of the sentences. The problem is that in some cases, a constituent in one sentence may have several potential matches in the other, and the matching heuristic may be unable to discriminate between the options.

In this paper we describe a new approach that attacks the weaknesses of the "parse-parse-match" procedure by using (1) only a translation lexicon with no language-specific grammar, (2) a bilingual rather than monolingual formalism, and (3) a probabilistic formulation for resolving the choice between candidate arrangements. Unlike the earlier approaches, our method operates in a single stage, simultaneously choosing the constituents of each sentence and the matchings between them.

In the sections that follow, we begin by briefly defining the inversion transduction grammar formalism. We then raise several desiderata on the expressiveness of any formalism for constituent matching, and discuss how the characteristics of the inversion transduction formalism are particularly suited to address these criteria. A specific grammar is then developed, and related issues are discussed. Note that all the examples in this paper are for English and Chinese, but the methods proposed are not dependent on this choice of languages.

2 Inversion Transduction Grammars

In Wu (1995) we define an *inversion transduction grammar* (ITG) formalism for bilingual language modeling, i.e., modeling of two languages (referred to as L_1 and L_2) simultaneously. The description here is necessarily brief; for further details the reader is referred to Wu (1995).

An ITG is a context-free grammar that generates output on two separate streams, together with a matching that associates the corresponding tokens and constituents of each stream. The formalism also differs from standard context-free grammars in that the concatenation operation, which is implicit in any production rule's right-hand side, is replaced with two kinds of concatenation with either *straight* or *inverted* orientation. Thus, the following are two distinct productions in an ITG:

$$\begin{array}{lcl} \mathrm{C} & \rightarrow & [\mathrm{A\ B}] \\ \mathrm{C} & \rightarrow & \langle \mathrm{A\ B} \rangle \end{array}$$

Consider each nonterminal symbol to stand for a pair of matched strings, so that for example (A_1, A_2) denotes the string-pair generated by A. The operator $[]$ performs the "usual" pairwise concatenation so that $[AB]$ yields the string-pair (C_1, C_2) where $C_1 = A_1B_1$ and $C_2 = A_2B_2$. But the operator $\langle\rangle$ concatenates constituents on output stream 1 while reversing them on stream 2, so that $C_1 = A_1B_1$ but $C_2 = B_2A_2$. The inverted concatenation operator permits the extra flexibility needed to accommodate many kinds of word-order variation between source and target languages. Since inversion is permitted at any level of rule expansion, a derivation may intermix productions of either orientation within the parse tree. More on the ordering flexibility will be said later.

There are also lexical productions of the form:

$$\text{A} \quad \rightarrow \quad x/y$$

where x and y are symbols of languages L_1 and L_2, respectively. Either or both x and y may take the special value ϵ denoting an empty string, allowing a symbol of either language to have no counterpart in the other language by being matched to an empty string. We call x/ϵ an L_1-singleton and ϵ/y an L_2-singleton.

Parsing, in the context of ITGs, means to take as input a sentence-*pair* rather than a sentence, and to output a parse tree that imposes a shared hierarchical structuring on both sentences. For example, Figure 1 shows a parse tree for an English-Chinese sentence translation. The English is read in the usual depth-first left-to-right order, but for the Chinese, a horizontal line means the right subtree is traversed before the left, so that the following sentence pair is generated:

(1) a. [[[The Authority]$_{\text{NP}}$ [will [[be accountable]$_{\text{VV}}$ [to [the [[Financial Secretary]$_{\text{NN}}$]$_{\text{NNN}}$]$_{\text{NP}}$]$_{\text{PP}}$]$_{\text{VP}}$]$_{\text{VP}}$]$_{\text{SP}}$ ./。]$_{\text{S}}$

b. [[[管理局]$_{\text{NP}}$ [將會 [[向 [[[財政 司]$_{\text{NN}}$]$_{\text{NNN}}$]$_{\text{NP}}$]$_{\text{PP}}$ [負責]$_{\text{VV}}$]$_{\text{VP}}$]$_{\text{VP}}$]$_{\text{SP}}$ ./。]$_{\text{S}}$

Alternatively, we can show the common structure of the two sentences more compactly using bracket notation with the aid of the $\langle\rangle$ operator:

(2) [[[The/ϵ Authority/管理局]$_{\text{NP}}$ [will/將會 $\langle$[be/ϵ accountable/負責]$_{\text{VV}}$ [to/向 [the/ϵ [[Financial/財政 Secretary/司]$_{\text{NN}}$]$_{\text{NNN}}$]$_{\text{NP}}$]$_{\text{PP}}$ $\rangle_{\text{VP}}$]$_{\text{VP}}$]$_{\text{SP}}$ ./。]$_{\text{S}}$

where the horizontal line from Figure 1 corresponds to the $\langle\rangle$ level of bracketing.

We prove in Wu (1995) the following convenient theorem, which indicates that any ITG can be converted to a normal form, where all productions are either lexical productions or binary-fanout productions:

Theorem 1 *For any inversion transduction grammar G, there exists an equivalent inversion transduction grammar G' in which every production takes one of the following forms:*

$$\begin{array}{lllllllll} S & \rightarrow & \epsilon/\epsilon & A & \rightarrow & x/\epsilon & A & \rightarrow & [B\,C] \\ A & \rightarrow & x/y & A & \rightarrow & \epsilon/y & A & \rightarrow & \langle B\,C \rangle \end{array}$$

We assume that ITGs are in this normal form for the remainder of this paper.

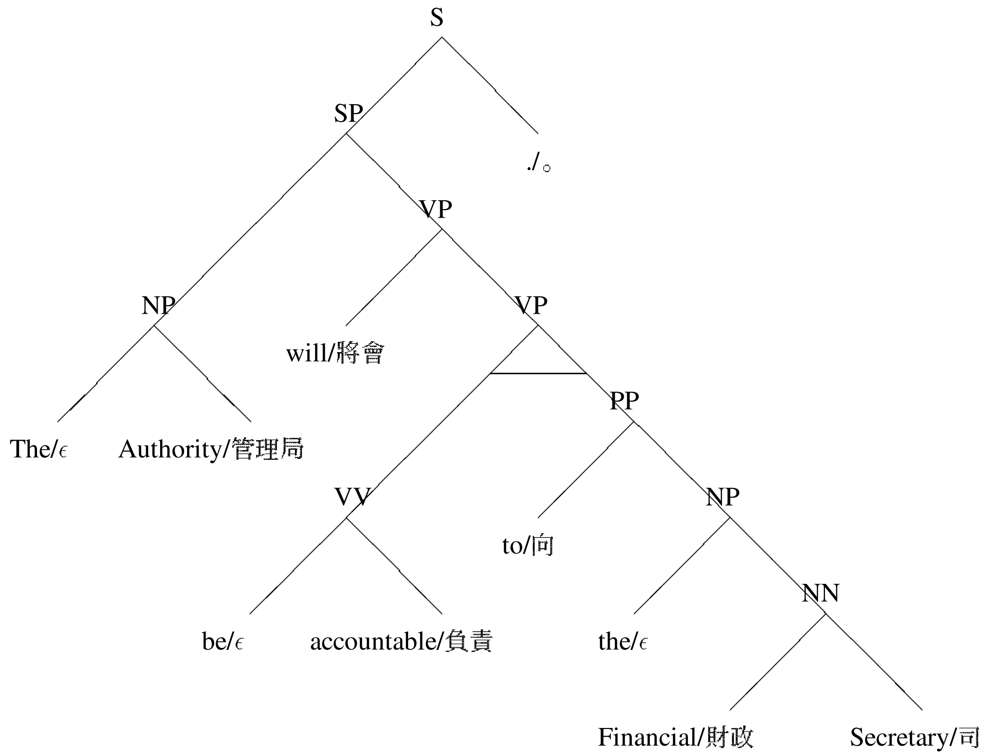


Figure 1: Inversion transducer parse tree.

3 Expressiveness Characteristics

Crossing Constraints We now turn to the first expressiveness desideratum for a matching formalism, namely *crossing constraints*, which prohibit arrangements where the matchings between subtrees cross each another, unless the subtrees' immediate parent constituents are also matched to each other. For example, given the constituent matchings depicted as solid lines in Figure 2, the dotted-line matchings corresponding to potential lexical translations would be ruled illegal. This constraint is important for computational reasons, to avoid exponential bilingual matching times. Note that a crossing constraint is implicit in most "parse-parse-match" approaches (Kaji *et al.* 1992; Cranias *et al.* 1994; Grishman 1994). Dependency-oriented versions of the constraint are found in Sadler & Vendelmans (1990); Matsumoto *et al.* (1993).

ITGs inherently implement a crossing constraint. The version of the crossing constraint as enforced by ITGs is actually even stronger. This is because even within a single constituent, the immediate subtrees are only permitted to cross in exact inverted order. As we shall argue below, this restriction reduces matching flexibility in a desirable fashion.

Fanout Constraints The second expressiveness desideratum for a matching formalism is to somehow limit the fanout of each constituent, which dictates the span over which matchings may cross. As the number of subtrees of a L_1-constituent grows, the number of possible matchings to subtrees of the corresponding L_2-constituent grows combinatorially, with corresponding time complexity growth on the matching process. Moreover, if constituents can immediately dominate too many tokens of the sentences, the crossing constraint loses effectiveness. We therefore seek to balance matching flexibility against constituent fanout, regardless of the formalism.

Recasting this issue in terms of the general class of context-free transduction grammars, the number of possible subtree matchings for a single constituent grows combinatorially with the number of symbols on a production's right-hand side. However, it turns out that the ITG restriction of allowing only matchings with straight or inverted orientation effectively cuts the combinatorial growth, while still maintaining flexibility where needed.

To see how ITGs maintain needed flexibility, consider Figure 3, which shows all 24 possible complete matchings between two constituents of length four each. Nearly all of these—22 out of 24—can be generated by an ITG as shown by the parse trees (whose nonterminal labels are

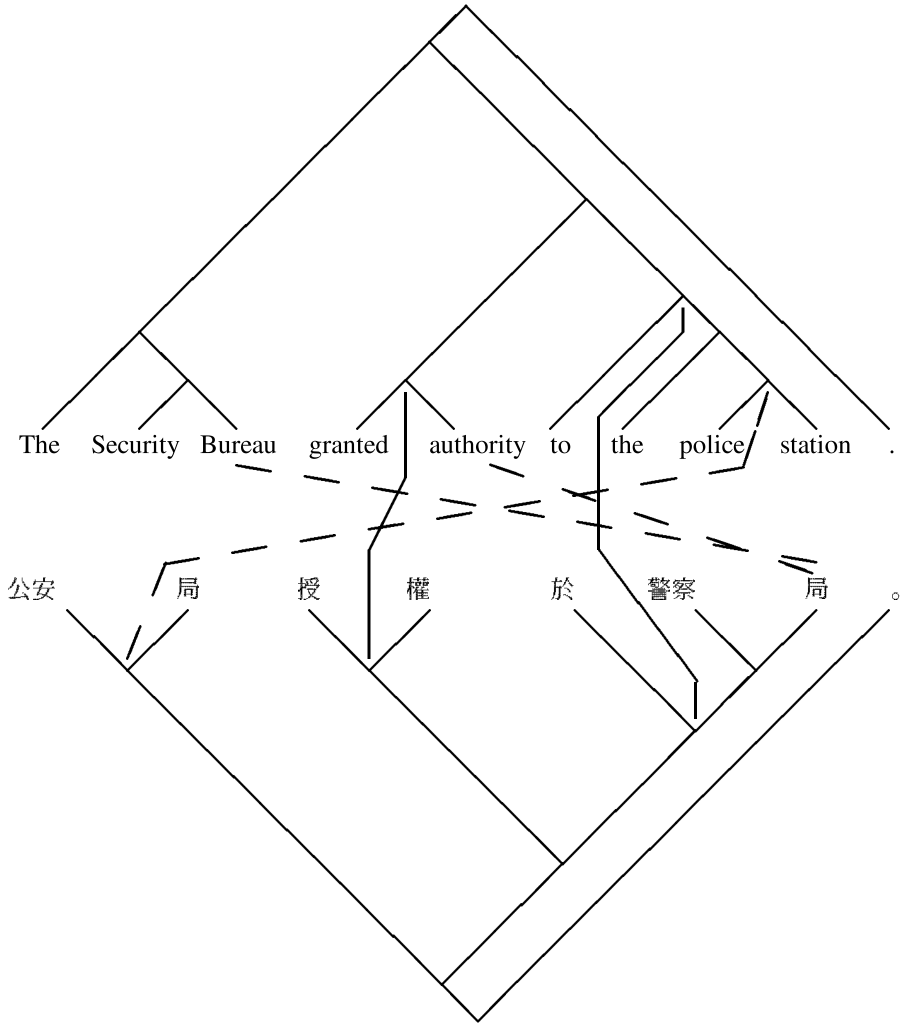


Figure 2: The crossing constraint (see text).

omitted).[2] The only two matchings that cannot be generated are very distorted transpositions that we might call “inside-out” matchings. We have been unable to find real examples of constituent arguments undergoing “inside-out” transposition between language pairs. The remaining 22 matchings, on the other hand, are representative of real word-order transpositions between language pairs.

[2] As discussed later, in many cases more than one parse tree can generate the same subconstituent matching. The trees shown are the canonical parses, as generated by the grammar of Figure 8.

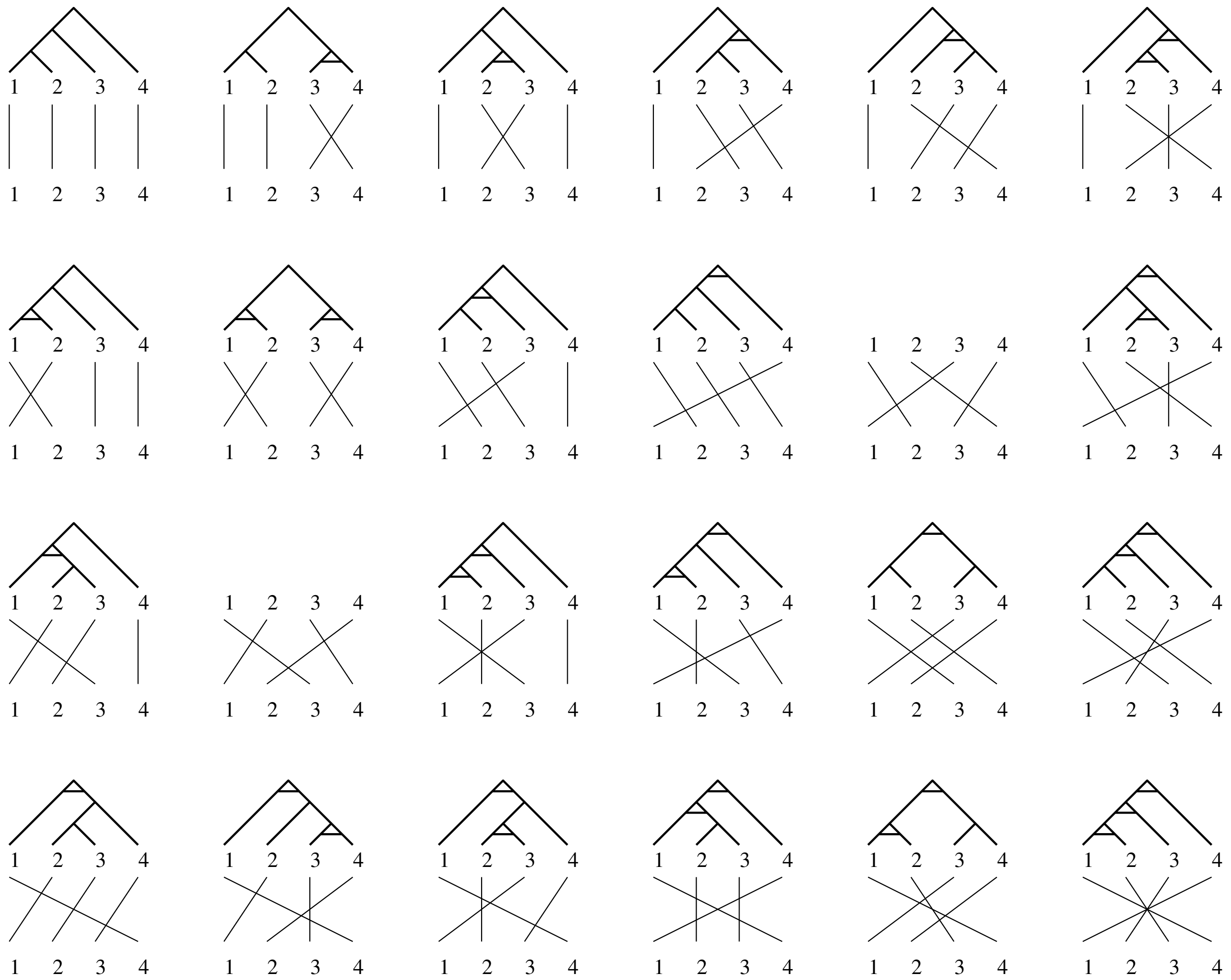


Figure 3: The 24 complete matchings of length four, with ITG parses for 22.

On the other hand, to see how ITGs cut combinatorial growth, consider the table in Figure 4, which compares growth in the number of legal complete matchings on a pair of subconstituent sequences. The third column shows the number of all possible complete matchings between two constituents with a fanout of f subconstituents each (therefore this is also the behavior for unconstrained context-free transduction grammars). Compare this against the second column, which shows the number of complete matchings that can be accepted by an ITG between a pair of length-f sequences of subconstituents. The fourth column shows the proportion of matchings that ITGs can accept. Flexibility is nearly total for sequences of up to $f \leq 4$ subconstituents, with a rapid drop thereafter corresponding to the elimination of undesirably tangled (i.e., non-

f	ITG	all matchings	ratio
0	1	1	1.000
1	1	1	1.000
2	2	2	1.000
3	6	6	1.000
4	22	24	0.917
5	90	120	0.750
6	394	720	0.547
7	1806	5040	0.358
8	8558	40320	0.212
9	41586	362880	0.115
10	206098	3628800	0.057
11	1037718	39916800	0.026
12	5293446	479001600	0.011
13	27297738	6227020800	0.004
14	142078746	87178291200	0.002
15	745387038	1307674368000	0.001
16	3937603038	20922789888000	0.000

Figure 4: Growth in number of legal *complete* subconstituent matchings for context-free transduction grammars with fanout f, versus ITGs on a pair of subconstituent sequences of length f each.

compositional) matchings.

Figure 5 shows the same numbers over all possible matchings, both complete and partial; in other words, for the more realistic case where some subconstituents are permitted to remain unmatched as singletons. The same desirable behavior is exhibited. The expressiveness of ITGs thus appears inherently suited to the degree of flexibility versus constraints needed for constituent matching.

f	ITG	all matchings	ratio
0	1	1	1.000
1	2	2	1.000
2	7	7	1.000
3	34	34	1.000
4	207	209	0.990
5	1466	1546	0.948
6	11471	13327	0.861
7	96034	130922	0.734
8	843527	1441729	0.585
9	7678546	17572114	0.437
10	71852559	234662231	0.306
11	687310394	3405357682	0.202
12	6693544171	53334454417	0.126
13	66167433658	896324308634	0.074
14	662393189919	16083557845279	0.041
15	6703261197506	306827170866106	0.022
16	68474445473303	6199668952527617	0.011

Figure 5: Growth in number of all legal subconstituent matchings (complete or partial) for context-free transduction grammars with fanout f, versus ITGs on a pair of subconstituent sequences of length f each.

4 Constituent Matching with a Generic ITG

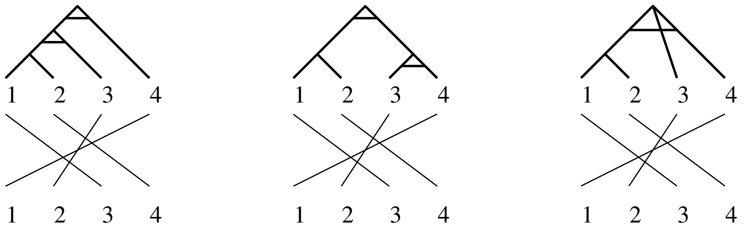
Because the expressiveness of ITGs constrains the space of possible matchings so appropriately, the possibility arises that the information supplied by a word-translation lexicon alone may be adequately discriminating to match constituents, without language-specific monolingual grammars for the source and target languages. That is, assuming that some algorithm for (bilingual) parsing is available, constituent matching with ITGs can be performed by devising a generic, language-independent grammar. As a first pass, a simple ITG of this kind is shown in Figure 6, employing only one nonterminal category. The first two productions are sufficient to generate all possible matchings of ITG expressiveness (this follows from the normal form theorem). The remaining productions are all lexical. Productions of the A $\rightarrow u_i/v_j$ form list all word translations found in the translation lexicon, and the others list all potential singletons without corresponding translations. Thus, a parser with this grammar can build a bilingual parse tree for any possible ITG matching on a pair of input sentences.

There are two broad problems with ambiguity in the simple approach. The first problem is illustrated by Figure 7; there is no justification for preferring either (a) or (b) over the other. In general the problem is that both the straight and inverted concatenation operations are associative. That is, $[A[AA]]$ and $[[AA]A]$ generate the same two output strings, which are also generated by $[AAA]$; and similarly with $\langle A\langle AA\rangle\rangle$ and $\langle\langle AA\rangle A\rangle$, which can also be generated by $\langle AAA\rangle$. Thus the parse shown in (c) is preferable to either (a) or (b) since it does not make an unjustifiable commitment either way. However, (c) is not permitted in normal form. We could add longer productions to the grammar, but this would unnecessarily complicate things and slow down parsing.

Instead, we employ a more complicated but better constrained grammar as shown in Figure 8, designed to produce only canonical tail-recursive parses. We differentiate type A and B constituents, representing subtrees whose roots have straight and inverted orientation, respectively. Under this grammar, a series of nested constituents with the same orientation will always have a left-heavy derivation. The guarantee that parsing will produce a tail-recursive tree facilitates easily identification of those nesting levels that are associative (and therefore arbitrary), so that those levels can be "flattened" by a post-processing stage after parsing into non-normal form trees like the one in Figure 7(c).

$$
\begin{array}{lcll}
\mathrm{A} & \rightarrow & [\mathrm{A\ A}] & \\
\mathrm{A} & \rightarrow & \langle \mathrm{A\ A} \rangle & \\
\mathrm{A} & \rightarrow & u_i/v_j & \text{for all } i, j\text{: lexical translations} \\
\mathrm{A} & \rightarrow & u_i/\epsilon & \text{for all } i\text{: } L_1\text{-only vocabulary} \\
\mathrm{A} & \rightarrow & \epsilon/v_j & \text{for all } j\text{: } L_2\text{-only vocabulary}
\end{array}
$$

Figure 6: A simple constituent-matching ITG.



(a) (b) (c)

Figure 7: Alternative ITG parse trees for the same matching.

A	$\rightarrow$	[A B]	
A	$\rightarrow$	[B B]	
A	$\rightarrow$	[C B]	
A	$\rightarrow$	[A C]	
A	$\rightarrow$	[B C]	
B	$\rightarrow$	$\langle$A A$\rangle$	
B	$\rightarrow$	$\langle$B A$\rangle$	
B	$\rightarrow$	$\langle$C A$\rangle$	
B	$\rightarrow$	$\langle$A C$\rangle$	
B	$\rightarrow$	$\langle$B C$\rangle$	
C	$\rightarrow$	u_i/v_j	for all i, j: lexical translations
C	$\rightarrow$	u_i/ϵ	for all i: L_1-only vocabulary
C	$\rightarrow$	ϵ/v_j	for all j: L_2-only vocabulary

Figure 8: A canonical-form constituent-matching ITG.

The second ambiguity problem is that (even under canonical form) a given sentence pair has multiple possible matchings (and therefore, parses). In the sentence pair of Figure 2, for example, both *Security Bureau* and *police station* are potential lexical matches to 公安局. To choose the best set of matchings, an optimization over some measure of overlap between the structural analysis of the two sentences is needed. Previous approaches to phrasal matching employ arbitrary heuristic functions on, say, the number of matched subconstituents.

The ITG framework suggests a probabilistic formalization of this optimization problem. A *stochastic inversion transduction grammar* is an ITG where a probability is associated with each production, subject to the constraint that

$$\sum_{1\le j,k\le N} (a_{i\to[jk]} + a_{i\to\langle jk\rangle}) + \sum_{\substack{1\le x\le W_1\\1\le y\le W_2}} b_i(x,y) = 1$$

where $a_{i\to[jk]} = P(i \to [jk]|i)$, $b_i(x,y) = P(i \to x/y|i)$, W_1 and W_2 are the vocabulary sizes of the two languages, and N is the number of nonterminal categories. Under the stochastic formulation, the parser's objective is to find the maximum-likelihood parse for a sentence pair. A general algorithm for this is given in Wu (1995).

We place probabilities on the grammar as shown in Figure 9. The b_{ij} distribution encodes the English-Chinese translation lexicon with degrees of probability on each potential word translation. A small ϵ-constant can be chosen for the probabilities $b_{i\epsilon}$ and $b_{\epsilon j}$, so that the optimal matching resorts to these productions only when it is otherwise impossible to match the singletons. The result is that the maximum-likelihood parser selects the parse tree that best meets the combined lexical translation preferences, as expressed by the b_{ij} probabilities.

5 Discussion

An experiment was carried out using a lexicon that was automatically learned from the HKUST English-Chinese Parallel Bilingual Corpus via statistical sentence alignment (Wu 1994) and statistical Chinese word and collocation extraction (Fung & Wu 1994; Wu & Fung 1994), followed by an EM word-translation learning procedure (Wu & Xia 1994). The latter stage gives us the b_{ij} probabilities directly. The translation lexicon contained approximately 6,500 English words and 5,500 Chinese words, and was *not* manually corrected for this experiment, having about 86% translation accuracy.

Approximately 2,000 sentence-pairs with both English and Chinese lengths of 30 words or less were then extracted from our corpus and (bilingually) parsed. Several additional criteria were used to filter out unsuitable sentence-pairs. If the lengths of the pair of sentences differed by more than a 2:1 ratio, the pair was rejected; such a difference usually arises as the result of an earlier error in automatic sentence alignment. Sentences containing more than one word absent from the translation lexicon were also rejected; the bracketing method is not intended to be robust

$$
\begin{array}{lcll}
\mathrm{A} & \stackrel{a}{\rightarrow} & [\mathrm{A\ B}] & \\
\mathrm{A} & \stackrel{a}{\rightarrow} & [\mathrm{B\ B}] & \\
\mathrm{A} & \stackrel{a}{\rightarrow} & [\mathrm{C\ B}] & \\
\mathrm{A} & \stackrel{a}{\rightarrow} & [\mathrm{A\ C}] & \\
\mathrm{A} & \stackrel{a}{\rightarrow} & [\mathrm{B\ C}] & \\
\mathrm{B} & \stackrel{a}{\rightarrow} & \langle \mathrm{A\ A} \rangle & \\
\mathrm{B} & \stackrel{a}{\rightarrow} & \langle \mathrm{B\ A} \rangle & \\
\mathrm{B} & \stackrel{a}{\rightarrow} & \langle \mathrm{C\ A} \rangle & \\
\mathrm{B} & \stackrel{a}{\rightarrow} & \langle \mathrm{A\ C} \rangle & \\
\mathrm{B} & \stackrel{a}{\rightarrow} & \langle \mathrm{B\ C} \rangle & \\
\mathrm{C} & \stackrel{b_{ij}}{\rightarrow} & u_i/v_j & \text{for all } i,j \text{ English-Chinese lexical translations} \\
\mathrm{C} & \stackrel{b_{i\epsilon}}{\rightarrow} & u_i/\epsilon & \text{for all } i \text{ English vocabulary} \\
\mathrm{C} & \stackrel{b_{\epsilon j}}{\rightarrow} & \epsilon/v_j & \text{for all } j \text{ Chinese vocabulary}
\end{array}
$$

Figure 9: A stochastic constituent-matching ITG.

against lexicon inadequacies. We also rejected sentence pairs with fewer than two matching words, since this gives the bracketing algorithm no discriminative leverage; such pairs accounted for less than 2% of the input data. Examples of the parsing output are shown in Figure 10.

The raw phrasal translations suggested by the parse output were then filtered to remove those pairs containing more than 50% singletons, since such pairs are likely to be poor translation examples. Examples that occurred more than once in the corpus were also filtered out, since repetitive sequences in our corpus tend to be non-grammatical markup. This yielded approximately 2,800 filtered phrasal translations, some examples of which are shown in Figure 11. A random sample of the phrasal translation pairs was then drawn, giving a precision estimate of 81.5%.

Although this already represents a useful level of accuracy, it does not in our opinion reflect the full potential of the formalism. Inspection revealed that performance was greatly hampered by our noisy translation lexicon which was automatically learned; it could be manually post-edited to reduce errors. Commercial online translation lexicons could also be employed if available. Higher precision could also be achieved without great effort by engineering a small number of

[I/我 hope/ϵ ϵ/<>望 employers/僱主 will/會 make full/ϵ ϵ/充分善 use/用 [of/ϵ those/那些] ⟨⟨[ϵ/的工 who/人] [have acquired/ϵ ϵ/學到 new/新 skills/技能]⟩ [through/透過 this/這個 programme/計劃]⟩ ./。]

[The/ϵ Authority/管理局 will/將會 ⟨[be/ϵ accountable/負責] [to the/ϵ ϵ/向 Financial/財政 Secretary/司]⟩ ./。]

[⟨[Even/ϵ more/更 important/重要] [,/ϵ however/但]⟩ [,/ϵ ϵ/的, is/是 to make the very best of our/ϵ ϵ/善用香港 own/本身 ϵ/的 talent/人才] ./。]

[I/我 have/已 <> at/ϵ length/詳細 ⟨ on/ϵ how/怎樣 we/我們 ϵ/講述⟩ [can/可以 boost/ϵ ϵ/促進 our/本港 ϵ/的 prosperity/繁榮] ./。]

Figure 10: Examples of sentence-pair parsing output. (<> = unrecognized input token.)

broad nonterminal categories. This would reduce errors for known idiosyncratic patterns, at the cost of manual rule building.

The automatically extracted phrasal translation examples are especially useful where the phrases in the two languages are not compositionally derivable solely from obvious word translations. An example is [have acquired/ϵ ϵ/學到 new/新 skills/技能] in Figure 10. The same principle applies to nested structures also, such as ⟨[ϵ/的工 who/人] [have acquired/ϵ ϵ/學到 new/新 skills/技能]⟩, on up to the sentence level.

Notice that under the ITG model, the word alignment problem becomes simply the special case of phrasal alignment at the parse tree leaves. Thus, the procedure intrinsically yields a set of word alignments, which we have found to be useful for other corpus analysis purposes.

In our experience, the main limitation of the formalism stems from the 1-to-1 matching assumption, which precludes matchings such as that between a single word like *when* as in

(3) The notes are always missing when useful.

and its translation into a two-part "bracketing" construction such as 當 ... 時 in

(4) 筆記總當有用時不見了。

With the ITG formalism it is necessary to build special productions to handle any such pattern.

1 % in real	1%的實質
Would you	你是否
an acceptable starting point for this new policy	是可接受為這項新政策的起點
are about 3 . 5 million	大概有350萬
born in Hong	在香港出生
for Hong	為香港
have the right to decide our	有權決定我
in what way the Government would increase their job opportunities ; and	政府如何增加他們的就業機會;及
last month	上個月
never to say " never "	不要說"永不"
reserves and surpluses	儲備和盈餘
starting point for this new policy	為這項新政策的起點
there will be many practical difficulties in terms of implementation	實行時會有很多實際困難
year ended 3 1 March 1 9 9 1	截至一九九一月三十一日

Figure 11: Examples of extracted phrasal translations.

6 Conclusion

We have described a method for extracting phrasal translation examples that replaces the conventional “parse-parse-match” approach with a single integrated procedure. The expressiveness constraints inherent in the ITG formalism are used to constrain the space of possible matchings, such that a word-translation lexicon alone supplies sufficient additional discriminating information to match constituents with surprisingly useful accuracy—without language-specific monolingual grammars for the source and target languages. Procedures of this kind are particularly effective for acquiring knowledge resources on languages less exhaustively studied than English.

References

BROWN, PETER F., JENNIFER C. LAI, & ROBERT L. MERCER. 1991. Aligning sentences in parallel corpora. In *Proceedings of the 29th Annual Conference of the Association for Computational Linguistics*, 169–176, Berkeley.

CATIZONE, ROBERTA, GRAHAM RUSSELL, & SUSAN WARWICK. 1989. Deriving translation data from bilingual texts. In *Proceedings of the First International Acquisition Workshop*, Detroit.

CHEN, STANLEY F. 1993. Aligning sentences in bilingual corpora using lexical information. In *Proceedings of the 31st Annual Conference of the Association for Computational Linguistics*, 9–16, Columbus, OH.

CHURCH, KENNETH W. 1993. Char-align: A program for aligning parallel texts at the character level. In *Proceedings of the 31st Annual Conference of the Association for Computational Linguistics*, 1–8, Columbus, OH.

CRANIAS, LAMBROS, HARRIS PAPAGEORGIOU, & STELIOS PEPERIDIS. 1994. A matching technique in example-based machine translation. In *Proceedings of the Fifteenth International Conference on Computational Linguistics*, 100–104, Kyoto.

DAGAN, IDO, KENNETH W. CHURCH, & WILLIAM A. GALE. 1993. Robust bilingual word alignment for machine aided translation. In *Proceedings of the Workshop on Very Large Corpora*, 1–8, Columbus, OH.

FUNG, PASCALE & KENNETH W. CHURCH. 1994. K-vec: A new approach for aligning parallel texts. In *Proceedings of the Fifteenth International Conference on Computational Linguistics*, 1096–1102, Kyoto.

FUNG, PASCALE & KATHLEEN MCKEOWN. 1994. Aligning noisy parallel corpora across language groups: Word pair feature matching by dynamic time warping. In *AMTA-94, Association for Machine Translation in the Americas*, 81–88, Columbia, Maryland.

FUNG, PASCALE & DEKAI WU. 1994. Statistical augmentation of a Chinese machine-readable dictionary. In *Proceedings of the Second Annual Workshop on Very Large Corpora*, 69–85, Kyoto.

GALE, WILLIAM A. & KENNETH W. CHURCH. 1991. A program for aligning sentences in bilingual corpora. In *Proceedings of the 29th Annual Conference of the Association for Computational Linguistics*, 177–184, Berkeley.

GRISHMAN, RALPH. 1994. Iterative alignment of syntactic structures for a bilingual corpus. In *Proceedings of the Second Annual Workshop on Very Large Corpora*, 57–68, Kyoto.

KAJI, HIROYUKI, YUUKO KIDA, & YASUTSUGU MORIMOTO. 1992. Learning translation templates from bilingual text. In *Proceedings of the Fourteenth International Conference on Computational Linguistics*, 672–678, Nantes.

KAY, MARTIN & M. RÖSCHEISEN. 1988. Text-translation alignment. Technical Report P90-00143, Xerox Palo Alto Research Center.

KUPIEC, JULIAN. 1993. An algorithm for finding noun phrase correspondences in bilingual corpora. In *Proceedings of the 31st Annual Conference of the Association for Computational Linguistics*, 17–22, Columbus, OH.

MATSUMOTO, YUJI, HIROYUKI ISHIMOTO, & TAKEHITO UTSURO. 1993. Structural matching of parallel texts. In *Proceedings of the 31st Annual Conference of the Association for Computational Linguistics*, 23–30, Columbus, OH.

NAGAO, MAKOTO. 1984. A framework of a mechanical translation between japanese and english by analogy principle. In *Artificial and human intelligence: Edited review papers presented at the International NATO Symposium on Artificial and Human Intelligence*, ed. by Alick Elithorn & Ranan Banerji, 173–180. Amsterdam: North-Holland.

SADLER, VICTOR & RONALD VENDELMANS. 1990. Pilot implementation of a bilingual knowledge bank. In *Proceedings of the Thirteenth International Conference on Computational Linguistics*, 449–451, Helsinki.

SMADJA, FRANK A. 1992. How to compile a bilingual collocational lexicon automatically. In *AAAI-92 Workshop on Statistically-Based NLP Techniques*, 65–71, San Jose, CA.

WU, DEKAI. 1994. Aligning a parallel English-Chinese corpus statistically with lexical criteria. In *Proceedings of the 32nd Annual Conference of the Association for Computational Linguistics*, 80–87, Las Cruces, New Mexico.

WU, DEKAI. 1995. Stochastic inversion transduction grammars, with application to segmentation, bracketing, and alignment of parallel corpora. In *Proceedings of IJCAI-95, Fourteenth International Joint Conference on Artificial Intelligence*, Montreal. To appear.

WU, DEKAI & PASCALE FUNG. 1994. Improving Chinese tokenization with linguistic filters on statistical lexical acquisition. In *Proceedings of the Fourth Conference on Applied Natural Language Processing*, 180–181, Stuttgart.

WU, DEKAI & XUANYIN XIA. 1994. Learning an English-Chinese lexicon from a parallel corpus. In *AMTA-94, Association for Machine Translation in the Americas*, 206–213, Columbia, Maryland.