Acquisition of Phrase-level Bilingual Correspondence using Dependency Structure

Kaoru Yamamoto and Yuji Matsumoto
Graduate School of Information Science,
Nara Institute of Science and Technology,
8916-5, Takayama-cho, Ikoma-shi, Nara, Japan

Abstract
This paper describes a method to find phrase-level translation patterns from parallel corpora by applying dependency structure analysis. We use statistical dependency parsers to determine dependency relations between base phrases in a sentence. Our method is tested with a business expression corpus containing 10000 English-Japanese sentence pairs and achieved approximately 90% accuracy in extracting bilingual correspondences. The result shows that the use of dependency relation helps to acquire interesting translation patterns.

1 Introduction
Since the advent of statistical methods in Machine Translation, the bilingual sentence alignment (Brown et al., 1991) or word alignment (Dagan et al., 1992) have been explored and achieved numerous success over the last decade. In contrast, fewer results are reported in phrase-level correspondence. As word sequences are not translated literally a word for a word, acquiring phrase-level correspondence still remains an important problem to be exploited.

This paper proposes a method to extract phrase-level correspondence from sentence-aligned parallel corpora using statistically probable dependency relations, i.e. head-modifier relations in a sentence.

The distinct characteristics of our approach is two-fold. First, our approach uses dependency relations rather than alignment, cognate and/or position heuristics previously applied (Melamed, 1995). Our approach is based on the assumption that the word ordering and positions may not necessarily coincide between the two languages, but the dependency structure between words will be preserved. We believe that dependency relations offer richer linguistic clues (syntactic information) and are effective for language pairs with different word ordering constraints.

Secondly, statistical dependency parsers are used to obtain candidate patterns. Previous methods mostly use rule-based parsers for preprocessing (Matsumoto et al., 1993), (Kitamura and Matsumoto, 1995). The progress in parsing technology are noteworthy, and in particular, various statistical dependency models have been proposed (Collins, 1997), (Ratnaparkhi, 1997), (Charniak, 2000). It has an advantage over the rule-based counterpart in that it achieves wider coverage, does not need to care for consistency in rule writing, and is robust to domain changes. We conjecture that our approach improves coverage and robustness by use of statistical dependency parsers.

In this paper, we aim to investigate the efficacy of statistically probable dependency structure in finding phrase-level bilingual correspondence. Though our discussion will proceed for English-Japanese phrasal correspondence, the proposed approach is applicable to any pair of languages.

This paper is organised as follows: In the next section, we present the overview of our approach. In Sections 3 and 4, components are elaborated in detail. In Section 5, experiment and results are given. In Section 6, we compare our approach with related works, and finally our findings are concluded in Section 7.

2 Overview of Our Approach
Our approach presupposes a sentence-aligned parallel corpora. The task is divided into two steps: a monolingual step in which candidate patterns are generated by use of dependency relations, and a bilingual step in which these candidate patterns from each language are paired
Our primary aim is to investigate the effectiveness of dependency structures in the monolingual candidate generation step. For this reason, the bilingual step borrows the weighted Dice coefficient and greedy determination from (Kitamura and Matsumoto, 1996).

In the following sections, we explain each step in detail.

3 Dependency-Preserving Candidate Patterns

Dependency grammar or related paradigm (Hudson, 1984) focuses on individual words and their relationships. In this framework, every phrase is regarded as consisting of a governor and dependants, where dependants may be optionally classified further. The syntactically dominating word is selected as the governor, with modifiers and complements acting as dependants. Dependency structures are suitably depicted as a directed acyclic graph (DAG), where arrows direct from dependants to governors.

We use a maximum likelihood model proposed in (Fujio and Matsumoto, 1998) where the dependency probability between segments are determined based on its co-occurrence and distance. It has constraints that (a) dependencies do not cross, (b) each segment has at least one governor\(^1\). Furthermore, the model has an option to allow multiple dependencies whose probabilities are above certain confidence. It is useful for cases where phrasal dependencies cannot be determined correctly using only syntactic information. It has an effect of improving recall by sacrificing precision and may contain more partially correct results useful for our candidate pattern generation.

We apply the following notions as units of segments: For English, (a) a preposition or conjunction is grouped into the succeeding baseNPs\(^2\), (b) auxiliary verbs are grouped into the succeeding main verb. For Japanese, one (or a sequence of) content word(s) optionally followed by function words\(^3\).

Having chunked into suitable segments, sentences are parsed to obtain dependency relations. We have setup the following three models:

1. **best-one model**: uses only the most likely (statistically best) dependency relations. At most one dependency is allowed for each segment.

2. **ambiguous model**: uses dependency relations above the certain confidence score 0.5\(^4\). Multiple dependencies may be considered for each segment.

3. **adjacent model**: uses only adjacency relations between segments. A segment is adjacent to the previous segment.

In the ambiguous model, we expect that more likely dependency relations will appear frequently given in a large corpus, thereby increasing the correlation score. Hence, ambiguity at parsing phase will hopefully resolved in the following bilingual pairing phase. As for the adjacent model, only chunking and its adjacency are used.

Finally, dependency relations between segments is used to generate candidate patterns.

---

\(^1\)except for the 'root' segment. For Japanese, the 'root' segment is the rightmost segment. For English, the segment that contains the main verb is regarded as the 'root' segment.

\(^2\)a baseNP or 'minimal' NP is non-recursive NP, i.e. none of its child constituents are NPs.

\(^3\)often referred as a bunsetsu.

\(^4\)statistically-not-the-best dependencies are also included if

\[\text{prob}(\text{kth ranked dependency}) \geq 0.5\]
In this paper, dependency size of a candidate pattern designates the number of segments connected through dependency relations. Figures 2, 3, and 4 illustrate examples of English candidate patterns of dependency size 1, 2, and 3 for the proposed dependency models.

In a dependency-connected candidate pattern, function words of the governor segment is dropped. This is to cope with data sparseness in generated candidate patterns. Moreover, two types of DACs can be generated from patterns of size 3, and we use DAG-type tags ('I' and 'T') to distinguish their types. We also note that candidate patterns do not necessarily follow the word ordering of original sentences.

The algorithm is as follows:

**Input:** a corpus, the minimum occurrence threshold in a corpus $f_{\text{min}}$ and the dependency size $d_{\text{wp}}$.

For each sentence in a corpus, process the following:

1. Part-of-Speech Tagging
2. Chunking: Rules are written as regular expressions defined over POS word sequences.
3. Dependency Analysis
4. Candidate Pattern Generation: Candidate patterns are generated and stored with their sentence ID. Dependency-connected patterns of less than or equal to the size $d_{\text{wp}}$ are extracted.

**Output:** a hash-table that maps from candidate patterns appearing at least the minimum occurrence $f_{\text{min}}$ to their sentence IDs found in the corpus.

### 4 Phrase-level Correspondence Acquisition

Pairing of candidate patterns is a combinatorial problem and we take the following tactics to reduce the search space. First, our algorithm works in a greedy manner. This means that a translation pair determined in the early stage of the algorithm will never be considered again.

Secondly, filtering process is incorporated. Figure 5 illustrates filtering for a sentence pair “I saw a girl in the park/*見た上次の少女を見た”。 A set of candidate patterns derived from English is depicted on the left, while that from Japanese is depicted on the right. Once a pair “I_girl_saw(T)/&…~'~ ~ ~ k (T)” is determined as a translation pair, then the algorithm assumes that “girl_saw(T)” will not be paired with candidate patterns related to “girl_saw(T)” (cancelled by diagonal lines in Figure 5) for the sentence pair. The operation effectively discards the found pairs and causes recalculation of correlation scores in the proceeding iterations.

As mentioned in Section 2, our correlation score is calculated by the weighted Dice Coefficient defined as:

$$\text{sim}(p_e,p_j) = \left(\log_2 f_{ej}\right) \frac{2f_{ej}}{f_e + f_j}$$

where $f_j$ and $f_e$ are the number of occurrences in Japanese and English corpora respectively and $f_{ej}$ is the number of co-occurrences.

The algorithm is as follows:

**Input:** hash-tables of candidate patterns for each language, the initial threshold of frequency $f_{\text{curr}}$ and the final threshold of frequency $f_{\text{min}}$. 

---

**Figure 2:** best-one model

| [I] | [saw] | [a girl] | [in the park] |
|-----|-------|----------|--------------|
| size 1) | {I, saw, girl, park} |
| size 2) | {I_saw, girl_saw, in-park_saw} |
| size 3) | {I_girl_saw(T), I_in-park_saw(T)} |

**Figure 3:** ambiguous model

| [I] | [saw] | [a girl] | [in the park] |
|-----|-------|----------|--------------|
| size 1) | {I, saw, girl, park} |
| size 2) | {I_saw, girl_saw, in-park_saw, in-park_girl} |
| size 3) | {I_girl_saw(T), I_in-park_saw(T), in-park_girl_saw(T)} |

**Figure 4:** adjacent model

| [I] | [saw] | [a girl] | [in the park] |
|-----|-------|----------|--------------|
| size 1) | {I, saw, girl, park} |
| size 2) | {saw, girl_saw, in-park_girl} |
| size 3) | {girl_saw(L), in-park_girl_saw(L)} |
Repeat the following until \( f_{\text{curr}} \) reaches \( f_{\text{min}} \).

1. For each pair of English candidate \( p_e \) and Japanese candidate \( p_j \) appearing at least \( f_{\text{curr}} \) times, identify the most likely correspondences according to the correlation scores.

   - For an English pattern \( p_e \), obtain the correspondence candidate set \( \mathcal{P}_e = \{ p_{j1}, p_{j2}, \ldots, p_{jn} \} \) such that \( \text{sim}(p_e, p_{jk}) > \log_2 f_{\text{min}} \) for all \( k \). Similarly, obtain the correspondence candidate set \( \mathcal{P}_j \) for a Japanese pattern \( p_j \).

   - Register \((p_e, p_j)\) as a translation pair if \( p_j = \arg\max_{p_{jk} \in \mathcal{P}_j} \text{sim}(p_e, p_{jk}) \) and \( p_e = \arg\max_{p_{ek} \in \mathcal{P}_e} \text{sim}(p_j, p_{ek}) \). The correlation score of \((p_e, p_j)\) is the highest among \( \mathcal{P}_j \) for \( p_e \) and \( \mathcal{P}_e \) for \( p_j \).

2. Filter out the co-occurrence positions for \( p_e, p_j \), and related candidate patterns.

3. Lower the threshold of frequency if no more pairs are found with \( f_{\text{curr}} \).

5 Experiment and Result

5.1 Experimental Setting

We use a business expression corpus (Takubo and Hashimoto, 1995) containing 10000 sentences pairs which are pre-aligned.

NLP tools are summarised in Table 1.

| preprocessing       | tool          | precision     |
|---------------------|---------------|---------------|
| POS(E)              | ChaSen2.0     | 96% precision |
| POS(J)              | ChaSen2.0     | 97% precision |
| chunking(E)         | SNPlex1.0     | rule-based    |
| chunking(J)         | Unit          | rule-based    |
| dependency(E)       | edep          | trial system  |
| dependency(J)       | jdep          | 85–87% precision |

Table 1: NLP tools

decremented by 1. If the number of registered translation pairs is less than 10, then \( f_{\text{curr}} \) is lowered in the next iteration. All parameters are empirically chosen.

5.2 Result

Our approach is evaluated by the metrics defined below:

\[
\text{precision} = \frac{\text{count}(p_t)}{\text{count}(p_e)}
\]

\[
\text{coverage} = \frac{\sum_{p_l}(\text{length}(p_l) \ast \text{cofreq}(p_l))}{\sum_{p_t}\text{occur}(p_t)}
\]

Precision measures the correctness of extracted translation pairs, while coverage measures the proportion of correct translation pairs in the parallel corpora. Let \( X \) be a pattern. \( \text{count}(X) \) gives the number of \( X \) returned, \( \text{occur}(X) \) gives the number of occurrences of \( X \) in each corpus, \( \text{length}(X) \) gives the dependency size of \( X \) and \( \text{cofreq}(X) \) gives the number of co-occurrences in the parallel corpora. \( p_e \) means extracted patterns, and of which correct patterns are designated as \( p_t \). \( p_l \) means the candidate patterns generated from each side of parallel corpora. Coverage is calculated for English
and Japanese separately and then their mean is taken.

Precision for each model is summarised in Tables 2, 3, and 4, while coverage is shown in Table 5. To examine the characteristics of each model, we expand correspondence candidate sets PE and Pj so that patterns with the correlation score ≥ \( \log_2 2 \) (≥ 1) are also considered. These are marked by asterisks "*" in Tables.

Random samples of correct and near-correct translation pairs are shown in Table 6, Table 7 respectively. Extracted translation pairs are matched against the original corpora to restore their word ordering. This restoration is done manually this time, but can be automated with little modification in our algorithm.

\[ f_{ij} = f_c = f_1 = f_{\text{min}} = 2 \]

### 5.3 Discussion

As we see from Table 2 and 3, the best-one model achieves better precision than the adjacent model. Upon inspecting the results, nearly the same translation patterns are extracted for higher thresholds. This is because our dependency parsers use the distance feature in determining dependency. Consequently, nearer segments are likely to be dependency-related. Experiment data shows that the exact overlaps are found in 9348 out of 14705 (63.55%) candidate patterns for English and 6625 out of 11566 (57.27%) for Japanese.

However, the difference appears when the threshold reaches 3 and patterns such as "not hesitate to contact" which is not found in the adjacent model are extracted. Moreover, the best-one model is better in terms of coverage. These results support that the dependency relations appear useful clues than just being linearly ordered.

Comparisoning the best-one model with the ambiguous model, the ambiguous model achieves a higher precision except for *2. This indicates...
that the accuracy of dependency parsers currently achieves are insufficient, and therefore, better to expand the possibilities of candidate patterns by allowing redundant dependency relations. As the dependency parsers improve, the best-one model will outperform the ambiguous model. However, as the result of *2 shows, candidates from redundant dependency relations are mostly extracted at the low threshold. The overall trend reveals that redundant relations act as noise at low thresholds, but help to scale up the correlation score at higher thresholds.

As shown in Table 6, a domain-specific disambiguation sample ("Thank you/ありがとうございます" vs. "Thank you in advance/前もってお願い申し上げます") is found. As for long-distance dependency-related translation patterns, "は-"-case (nominative) and verb patterns (consultations include/協議に含める) are extracted. Other types of long-distance translation patterns such as "で"-case (accusative) and verb patterns (be held at X/X で開催する) are not extracted even candidate patterns from each corpus are generated.

Generally speaking, acquiring long-distance translation patterns is a hard problem. We still require further investigation examining under what circumstance the dependency relations are really effective. So far, we use relatively "clean" business expression corpora which is a collection of standard usage. However, in the real world setting, more repetitions and variations will be observed. Adjuncts can be placed in less constrained way and the adjacent model cannot deal with if they are apart. In such cases, availability of robust dependency parsers become essential, dependency relations plays a key role in finding the long-distance translation patterns.

---

### Table 6: random samples of correct translation patterns in best-one model. “+” indicates a segment-separator and “*” indicates a morpheme-separator.

| English | Japanese | score |
|---------|----------|-------|
| thank-you | ありがとう | 4.7037 |
| consultations+include | 協議に含める | 2.3219 |
| apply+for.the_position | 報に応募いたします | 2.2157 |
| thank+you+in_advance | 前もってお願い申し上げる | 1.6000 |
| not+hesitate+to.contact | 遠慮なくご連絡 | 1.6000 |
| be+enclosed+a+copy | 1部同封いたします | 1.0566 |
| be+writing+to+let+know | 書状をもってお知らせいたします | 1.0566 |
| applications+include | 用途に+ある | 1.0000 |
| upcoming.board+of+director+s+meeting | 次回の+取締役会 | 1.0000 |
| will_have+to+cancel | 中止ざるを得なくなる | 1.0000 |
| have+high_hope | 大いに+期待する | 1.0000 |
| business+is+expanded | 商売が+発展する | 1.0000 |
| we+have+learned+from+your+fax | 貴ファックスで+知る | 1.0000 |
| leaving+in+about+ten+days | 約1ヶ月後に+発送 | 1.0000 |
| get+you+in+close+business+relationship | お付き合いを+築く | 1.0000 |
| we+are+inquiring+regarding | に関し+お尋ねいたします | 1.0000 |
| pay+special+attention | 特別の+注意を+払う | 1.0000 |

### Table 7: random samples of near-correct translation patterns where score is 1.000. Segments to be deleted to become correct patterns are embraced by "()". Segments to be added are embraced by "[]".

| English | Japanese |
|---------|----------|
| (have Been pleased)+to+serve+as+their+main+banker | 王立銀行+となる | |
| [be held]+at+hotel+new+ohaha | ホテルニューオーバーで+開催する |
| assets+position+(in+good+shape) | 資産+状態 |
| (have been placed)+into+our+file | 私どもの+ファイル |
| (put)+one+month+limit | 1ヶ月の+期限 |
| [passed]+on+past+tuesday | 火曜日に+亡くなりる |

---

6A typical Japanese sentence follows S-O-V structure, while the English counterpart follows S-V-O structure.
6 Related Works

Smadja et al. (1996) finds rigid and flexible collocations. They first identify candidate collocations in English, and subsequently, find the corresponding French collocations by gradually expanding the candidate word sequences. Kitamura et al. (1996) enumerates word sequences of arbitrary length (n-gram of content words) that appear more than the minimum threshold from English and Japanese and attempts to find the correspondence based on the prepared candidate lists.

Difference from Smadja et al. (1996) is that our method is bi-directional and difference from Kitamura et al. (1996) is that we use dependency relations which leads to "structured" phrasal correspondence as opposed to "flat" adjacent correspondence.

On the other hand, Matsumoto et al. (1993), Kitamura et al. (1995) and Meyers et al. (1996) use dependency structure for structural matching of sentences to acquire translation rules. Their methods employ grammar-based parsers and only work for declarative sentences. Their objectives are complete matching of dependency trees of two languages.

Instead, our method uses statistical dependency parsers and are not restricted to simple sentences for input. Furthermore, we are concerned with partial matching of dependency trees so that the overall robustness and coverage will be improved.

7 Conclusion

In this paper, we propose a method to find phrase-level bilingual correspondence using dependency structure from parallel corpora. We have conducted a preliminary experiment with 10000 business sentence pairs of English and Japanese and achieved approximately 90% precision.

Though a fuller investigation still requires, our finding shows that the dependency relations serve as useful linguistic clues in the task of phrase-level bilingual correspondence acquisition.

References

P.F. Brown, J.C. Lai, and R.L. Mercer. 1991. Aligning sentences in parallel corpora. In ACL-92: 29th Annual Meeting of the Association for Computational Linguistics, pages 169–176.

E. Charniak. 2000. A maximum-entropy-inspired parser. In NAACL-2000: 1st Meeting of the North American Chapter of the Association for Computational Linguistics, pages 132–139.

M.J. Collins. 1997. Three generative, lexicalised models for statistical parsing. In ACL-97: 35th Annual Meeting of the Association for Computational Linguistics, pages 16–23.

I. Dagan, K. Church, and W. Gale. 1992. Robust bilingual word alignment for machine aided translation. In Proc. of the Workshop on Very Large Corpora, pages 1–8.

M. Fujio and Y. Matsumoto. 1998. Japanese dependency structure analysis based on lexicalized statistics. In Proc. of 3rd Conf. on Empirical Methods in Natural Language Processing, pages 88–96.

R. Hudson. 1984. Word Grammar. Blackwell.

M. Kitamura and Y. Matsumoto. 1995. A machine translation system based on translation rules acquired from parallel corpora. In Proc. of Recent Advances in Natural Language Processing, pages 27–44.

M. Kitamura and Y. Matsumoto. 1996. Automatic extraction of word sequence correspondences in parallel corpora. In Proc. 4th Workshop on Very Large Corpora, pages 79–87.

Y. Matsumoto, H. Ishimoto, and T. Utsuro. 1993. Structural matching of parallel texts. In ACL-93: 31st Annual Meeting of the Association for Computational Linguistics, pages 23–30.

I.D. Melamed. 1995. Automatic evaluation and uniform filter cascades for inducing n-best translation lexicons. In Proc. of 3rd Workshop on Very Large Corpora, pages 184–198.

A. Ratnaparkhi. 1997. A linear observed time statistical parser based on maximum entropy models. In Proc of 2nd Conf. on Empirical Methods in Natural Language Processing, pages 1–10.

K. Takubo and M. Hashimoto. 1995. A Dictionary of English Business Letter Expressions. Nihon Keizai Shimbun, Inc.