Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 401943, 10 pages
http://dx.doi.org/10.1155/2014/401943

Research Article

Unsupervised Chunking Based on Graph Propagation from Bilingual Corpus

Ling Zhu, Derek F. Wong, and Lidia S. Chao

Natural Language Processing & Portuguese-Chinese Machine Translation Laboratory, Department of Computer and Information Science, University of Macau, Macau

Correspondence should be addressed to Ling Zhu; mb05448@umac.mo

Received 30 August 2013; Accepted 8 December 2013; Published 19 March 2014

Academic Editors: J. Shu and F. Yu

This paper presents a novel approach for unsupervised shallow parsing model trained on the unannotated Chinese text of parallel Chinese-English corpus. In this approach, no information of the Chinese side is applied. The exploitation of graph-based label propagation for bilingual knowledge transfer, along with an application of using the projected labels as features in unsupervised model, contributes to a better performance. The experimental comparisons with the state-of-the-art algorithms show that the proposed approach is able to achieve impressive higher accuracy in terms of *F*-score.

1. Introduction

Shallow parsing, is also called text chunking, plays a significant role in natural language processing (NLP) community. It can be regarded as a classification task, as a process of training a classifier to segment a sentence and labeling each partitioned phrase with an accurate chunk tag. The classifier takes the words, their POS tags, and surrounding context as features of an instance, whereas the chunk tag is the class label. The goal of shallow parsing is to divide a sentence into labeled, nonoverlapping, and nonrecursive chunks based on different methodologies.

Plenty of classification algorithms have been applied in the field of shallow parsing. The models are broadly categorized into three types: rule-based chunking model, machine learning-based model, and memory-based model. Particularly, in recent decades we have witnessed the remarkable advancement of the state of the art on chunking task by applying supervised learning approaches. Supervised chunking model, for example, MEMs (maximum entropy models), which is employed by [1] solves the ambiguous tagging problem by training on a corpus to compute the probability information of each word in the input context and get a good performance.

Although supervised learning algorithms have resulted in the state of the art and high accuracy systems on varieties of tasks in the NLP domain, the performance in source-poor language is still unreasonable. A fundamental obstacle of statistical shallow parsing for the quantities of world's language is the shortage of annotated training data. Furthermore, the work of well-understand hand annotation has proved to be expansive and time consuming. For example, over one million dollars have been invested in the development of Penn Treebank [2], and the lack of developed Treebank and tagged corpus in the majority of languages illustrate that it is difficult to raise the investment for annotation projects. However, unannotated parallel text data is broadly available because of the explosive growth of multilingual website, news streams, and human translations of books and documents. These suggest that unsupervised methods appear to be a useful solution for inducing chunking taggers, as they do not need any annotated text for training. Unfortunately, the existing unsupervised chunking systems do not have a reasonable performance to make its practical usability questionable at best.

To bridge the gap of the accuracy between source-rich languages and source-poor languages, we would like to leverage the existing resources of source-rich languages like English when doing the shallow parsing for the source-poor foreign languages such as Chinese. In this work, the system assumes that there is no labeled data available for training but

that we have access to parallel English data. This part of work is closest to [3], but there are two main differences to improve the weakness of dealing with the unaligned words.

First, a novel graph-based framework is applied to project syntactic information across language boundaries for several reasons as follows.

(i) Graphs can indicate complicated relationships between classes and instances. For instance, an ambiguous instance interest could belong to the class of both NP and VP.

(ii) Chunks from multiple sources, such as different tree banks, web sites,and texts can be represented in a single graph.

(iii) The paths of information propagation of graphs make it possible to explicit the potential common information among different instances. That is to say, the algorithm relies on the fact that neighbor cases should belong to the same class, and the relationships between data points are captured in the form of a similarity graph with vertices corresponding to the cases and edge weights to their similarity.

In the end, a bilingual graph is constructed to represent the relationships between English and Chinese.

Second, rather than directly using these projected labels as features in supervised training phases, we prefer to employ them in unsupervised model. Additionally, to facilitate bilingual unsupervised chunking research and standardize best practice, a tag set consists of eight universal chunk categories is proposed in this work.

The paper is organized as follows. Section 2 introduces the overall approach of our work. Section 3 focuses on how to establish the graph and carry out the label propagation. Section 4 describes the unsupervised chunk induction and feature selection in detail. In Section 5, the evaluation results are presented followed by a conclusion to end this paper.

2. Approach Overview

The central aim of our work is to build a chunk tagger for Chinese sentences, assuming that there is an English chunk tagger and some parallel corpuses between these two languages. Hence, we can conclude that the emphasis of our approach is how to build a bilingual graph from the sentence-aligned Chinese-English corpus. Two types of vertices will be employed in our graph: trigram types are used on the Chinese side corresponding to different vertices, while on the English side the vertices are individual word types. During the graph construction, no labeled data is needed but does require two similarity functions. A cooccurrence based similarity function is applied to compute edge weights between Chinese trigrams. This function is designed to indicate the syntactical similarity of the middle words of the neighbor trigrams. A second similarity function, which leverages standard unsupervised word alignment statistics, is employed to establish a soft correspond between Chinese and English.

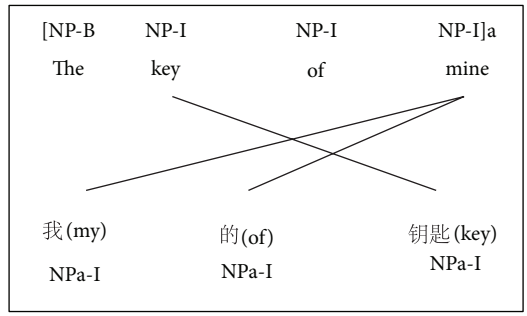

Figure 1: Direct label projection from English to Chinese with position information.

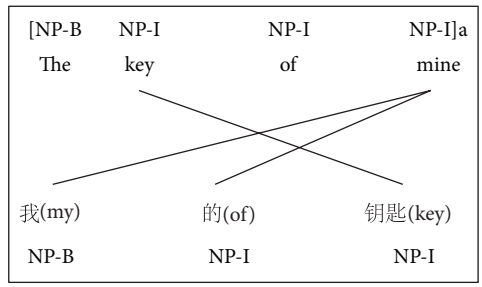

Figure 2: Adjust the Cchunk tag based on position information.

Based on the reality that we do not have labeled Chinese data, we aim to project the syntactic information from the English side to the Chinese side. Before initializing the graph, a supervised chunking model is used to tag the English side of the parallel text. The label distributions for the English vertices are generated by aggregating the chunk labels of the English tokens to types. Consider the crossover problem and different word orders between Chinese and English, the position information is also considered in our work. Based on this idea, the chunk tags are projected directly from English side to Chinese along with the position as shown in Figure 1. Based on the position information, we can get the exact boundary of each chunk at the Chinese side. Then an adjustment is made to assign the correct label as shown in Figure 2. After graph initialization, label propagation is applied to transfer the chunk tags to the peripheral Chinese vertices firstly (i.e., the ones which are adjacent to the English vertices), followed by further propagating among all the Chinese vertices. It is worth mentioning that the chunk distributions over the Chinese trigram types are regarded as features for learning a better unsupervised chunking tagger. The following sections will explain these steps in detail (see Algorithm 1).

3. Graph Construction

In graph learning approaches, one constructs a graph whose vertices are labeled and unlabeled examples and whose weighted edges encode the degree to which examples they link have the same label [4]. Notice that graph construction used for the problems of structured prediction such as

Bilingual Chunking Induction.
Require: Parallel English and foreign language data D^e and D^c; unlabeled foreign training data Γ^c; English tagger
Ensure: θ^c, a set of parameters learned using a constrained unsupervised model.
(1) $D^{e \leftrightarrow c} \leftarrow$ **word-align-bitext** (D^e, D^c)
(2) $\widehat{D}^e \leftarrow$ **chunk-tag-supervised** (D^e)
(3) A $(D^{e \leftrightarrow c}, \widehat{D}^e)$
(4) $G \leftarrow$ **construct-graph** (Γ^c, D^c, A)
(5) $\widetilde{G} \leftarrow$ **graph-propagate** (G)
(6) $\Delta \leftarrow$ **extract-word-constraints** $(\widetilde{G})$
(7) $\theta^f \leftarrow$ **chunk-induce-constraints** (Γ^c, Δ)
(8) return θ^f

Algorithm 1: Graph-based unsupervised chunking approach.

shallow parsing is nontrivial for the following two reasons. First, it is necessary for resolving ambiguous problem by employing individual words as the vertices instead of the context. Besides, if we use the whole sentences as the vertices during graph construction, it is unclear how to calculate the sequence similarity. Altun et al. [5] proposed a graph-based semisupervised learning approach by using the similarity between labeled and unlabeled structured data in a discriminative framework. Recently, a graph over the cliques in an underlying structured model was defined by [6]. In their work, a new idea has been proposed that one can use a graph over trigram types, combing with the weights based on distribution similarity, to improve the supervised learning algorithms.

3.1. Graph Vertices. In this work, we extend the intuitions of [6] to set up the bilingual graph. Two different types of vertices are applied for each language because the information transformation in our graph is asymmetric (from English to Chinese). On the Chinese side, the vertices (denoted by D^c) correspond to trigram types, which are the same as in [6]. The English vertices (denoted by D^e), however, correspond to word types. The reason we do not need the trigram types in English side to disambiguate them is that each English word are going to be labeled. Additionally, in the label propagation or graph construction, we do not connect the vertices between any English words but only to the Chinese vertices.

Furthermore, there are two kinds of vertices consisting of ones extracted from the different sides of word aligned bitext (D^e, D^c) and an additional unlabeled Chinese monolingual corpus Γ^c, which will be used later in the training phase. Such as noted, two different similarities will be employed to define the edge weights between vertices from these two languages and among the Chinese vertices.

3.2. Monolingual Similarity Function. In this section, a brief introduction is given as follows in terms of computing the monolingual similarity between the connecting pairs of Chinese trigram type. Specifically, the set V^c consists of all the word n-grams that occur in the text. Given a symmetric similarity function between types defined below, we link types m_i and m_j $(m_i, m_j \in V_c)$ with an edge weight $w_{m_i m_j}$ as follows:

$$w_{m_i m_j} = \begin{cases} \text{sim}\left(m_i, m_j\right) & \text{if } m_i \in \kappa\left(m_j\right) \text{ or } m_j \in \kappa\left(m_i\right) \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

where $\kappa(m_j)$ is the set of k-nearest neighbors of m_i according to the given similarity. For all the parameters in this paper, we define $k = 5$.

To define the similarity function for each trigram type as $m_i \in V_c$, and we rely on the cooccurrence statistics of ten features illustrated in Table 1.

Table 1: Various features for computing edge weights between Chinese trigram types.

Description	Feature
Trigram + context	$x_1x_2x_3x_4x_5$
Trigram	$x_2x_3x_4$
Left context	x_1x_2
Right context	x_4x_5
Center word	x_3
Trigram − center word	x_2x_4
Left word + right context	$x_2x_4x_5$
Right word + left context	$x_1x_2x_4$
Suffix	Has suffix (x_3)
Prefix	Has prefix (x_3)

Based on this monolingual similarity function, a nearest neighbor graph could be achieved. In this graph, the edge weight for the n most similar trigram types is set to the PMI values and is 0 for all other ones. Finally, we apply the function (m) to denote the neighborhood of vertex m and set the maximum number of 5 in our experiments.

3.3. Bilingual Similarity Function. We rely on high-confidence word alignments to define a similarity function

between the English and Chinese vertices. Since the graph vertices are extracted from a parallel corpus, a standard word alignment technique GIZA++ [7] is applied to align the English sentences D^e and their Chinese translations D^c. Based on the idea that the label propagation process in the graph will provide coverage and high recall, only a high confidence alignment $D^{e \leftrightarrow c} > 0.9$ is considered.

Therefore, we can extract tuples of the form $[t \leftrightarrow u]$ based on these high-confidence word alignments, where the Chinese trigram types that the middle word aligns to an English unigram type u. Relying on computing the proportion of these tuple counts, we set the edge weights between the two languages by our bilingual similarity function.

3.4. Graph Initialization. So far, we have introduced how to construct a graph. But the graph here is unlabeled completely. For the sake of label propagation, we have to use a supervised English chunking tagger to label the English side of the parallel corpus in the graph initialization phase. Afterwards we simply count the individual labels of each English token and then normalize the counts to get the tag distribution over the English unigram types. These distributions are in the use of initializing the English vertices' distribution in the graph. Considering that we extract the English vertices from a bitext, all vertices in English word types are assigned with an initial label distribution.

3.5. Graph Example. Figure 3 shows a simple small excerpt from a Chinese-English graph. We see that only the Chinese trigrams [一项 重要 选择], [因素 是 这样] and [不断 深入] are connected to the English translated words. All these English vertices are labeled by a supervised tagger. In this particular case, the neighborhoods can be more diverse and a soft label distribution is allowed over these vertices. As one can see, Figure 3 is composed of three subgraphs. In each subgraph, it is worth noting that the middle of Chinese trigram types has the same chunking types (with the labeled one). This exhibits the fact that the monolingual similarity function guarantees the connected vertices having the same syntactic category. The label propagation process then spreads the English words' tags to the corresponding Chinese trigram vertices. After that, labels are further propagated among the Chinese vertices. This kind of propagation is used to convey these tags inwards and results in tag distributions for the middle word for each Chinese trigram type. More details on how to project the chunks and propagate the labels will be described in the following section.

3.6. Label Propagation. Label propagation operates on the constructed graph. The primary objective is to propagate labels from a few labeled vertices to the entire graph by optimizing a loss function based on the constraints or properties derived from the graph, for example, smoothness [4, 8] or sparsity [9]. State-of-the-art label propagation algorithms include LP-ZGL [4], adsorption [10], MAD [8], and sparsity-inducing penalties [9].

Label propagation is applied in two phases to generate soft label distributions over all the vertices in the bilingual graph. In the first stage, the label propagation is used to project the English chunk labels to the Chinese side. In detail, we simply transfer the label distributions from the English word types to the connected Chinese vertices (i.e., V_c^l) at the periphery of the graph. Note that not all the Chinese vertices are fully connected with English words if we consider only the high-confidence alignments. In this stage, a tag distribution d_i over the labels y is generated, which represents the proportion of times the center word $c_i \in V_c$ aligns to English words e_y tagged with label y:

$$d_i(y) = \frac{\sum_{e_y} \#\left[c_i \longleftrightarrow e_y\right]}{\sum_{y'} \sum_{e_{y'}} \#\left[c_i \longleftrightarrow e_{y'}\right]}. \tag{2}$$

The second phase is the conventional label propagation to propagate the labels from the peripheral vertices V_c^l to all Chinese vertices in the constructed graph. The optimization on the similarity graph is based on the objective function:

$$\begin{aligned} P(q) = & \sum_{c_i \in V_c \setminus C_c^l} w_{ij} \left\| q_i - q_j \right\|^2 + \lambda \left\| q_i - U \right\|^2 \\ \text{s.t.} \quad & \sum_y q_i(y) = 1 \quad \forall c_i \\ & q_i(y) \geq 0 \quad \forall c_i, y \\ & q_i = d_i \quad \forall c_i \in V_c^l, \end{aligned} \tag{3}$$

where q_i $(i = 1, \ldots, |V_c|)$ are the label distributions over all Chinese vertices and λ is the hyperparameter that will be discussed in Section 4. Consider that $\|q_i - q_j\|^2 = \sum_y (q_i(y) - q_j(y))^2$ is a squared loss, which is used to penalize the neighbor vertices to make sure that the connected neighbors have different label distributions. Furthermore, the additional second part of the regulation makes it possible that the label distribution over all possible labels y is towards the uniform distribution U. All these show the fact that this objective function is convex.

As we know, the first term in (3) is a smoothness regularizer which can be used to encourage the similar vertices, which have the large w_{ij} in between, to be much more similar. Moreover, the second part is applied to regularize and encourage all marginal types to be uniform. Additionally, this term also ensures that the unlabeled converged marginal vertices will be uniform over all tags if these types do not have a path to any labeled vertices. This part makes it possible that the middle word of this kind unlabeled trigram takes on any possible tag as well.

However, although a closed form solution can be derived through the objective function mentioned above, it would be impossible without the inversion of the matrix of order $|V_c|$. To solve this problem, we rely on an iterative update based algorithm instead. The formula is as follows:

$$q_i^{(m)}(y) = \begin{cases} c_i(y), & \text{if } c_i \in V_c^l \\ \dfrac{\gamma_i(y)}{\kappa_i}, & \text{otherwise,} \end{cases} \tag{4}$$

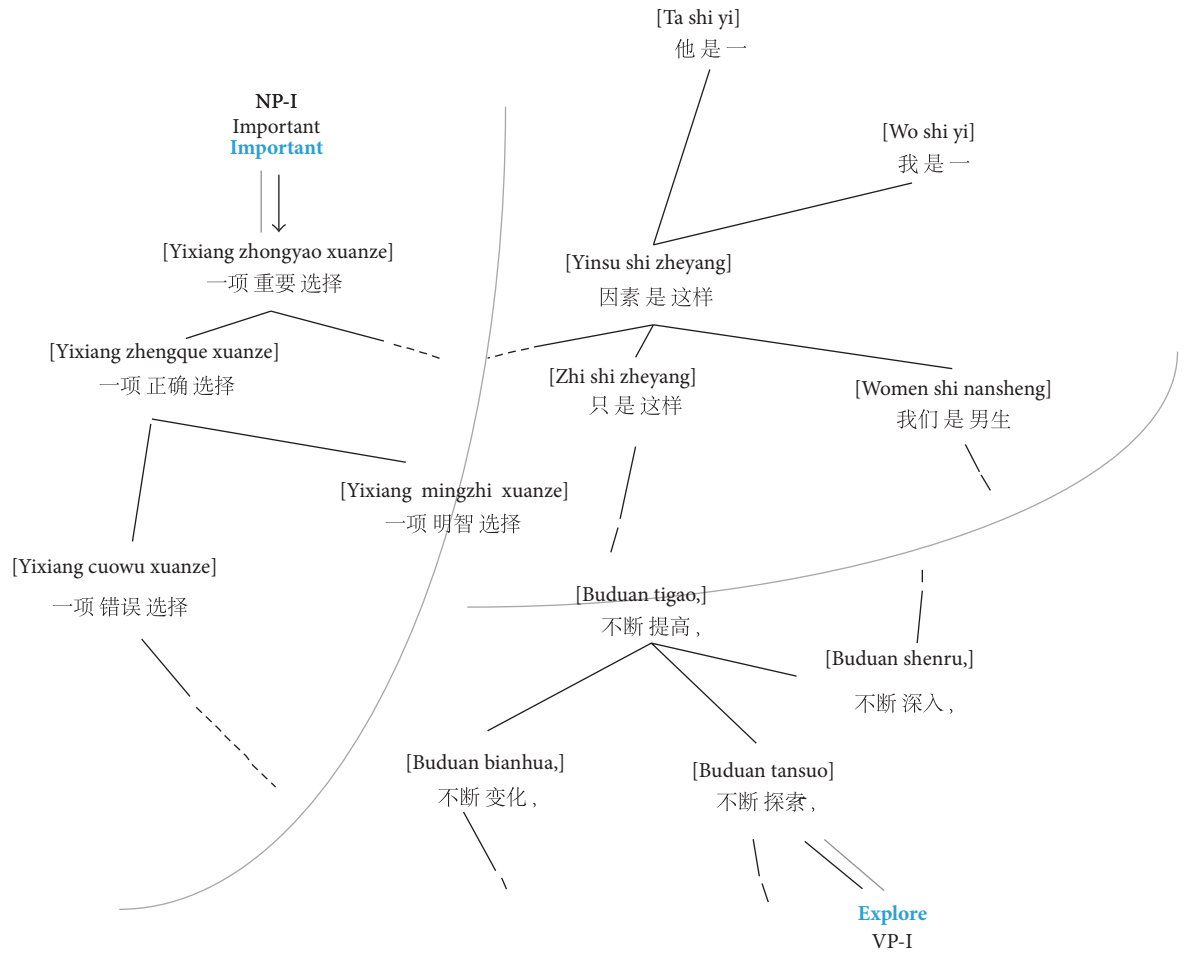

Figure 3: An example of similarity graph over trigram on labeled and unlabeled data.

where $\forall c_i \in V_c \setminus V_c^l$, $\gamma_i(y)$, and k_i are defined as follows:

$$\gamma_i(y) = \sum_{c_j \in N(c_i)} w_{ij} q_i^{(m-1)}(y) + \lambda U(y),$$
$$k_i = \lambda + \sum_{c_j \in N(c_i)} w_{ij}. \tag{5}$$

This procedure will be processed 10 iterations in our experiment.

4. Unsupervised Chunk Induction

In Section 3, the bilingual graph construction is described such that English chunk tags can be projected to Chinese side by simple label propagation. This relies on the high-confidence word alignments. Many Chinese vertices could not be fully projected from the English words. To complement, label propagation is used to further propagate among the Chinese trigram types. This section introduces the usage of unsupervised approach to build a practical system in the task of shallow parsing. The implementation of how to establish the chunk tag identification system will be discussed and presented in detail.

4.1. Data Representation

4.1.1. Universal Chunk Tag. The universal tag was first proposed by [11] that consists of twelve universal part-of-speech categories for the sake of evaluating their cross-lingual POS projection system for six different languages. In this work, we follow the idea but focus on the universal chunk tags between English and Chinese for several reasons. First, it is useful for building and evaluating unsupervised and cross-lingual chunk taggers. Additionally, taggers constructed based on universal tag set can result in a more reasonable comparison across different supervised chunking approaches. Since two kinds of tagging standards are applied for the different languages, it is vacuous to state that "shallow parsing for language *A* is much harder than that for language *B*" when the tag sets are incomparable. Finally, it also permits our model to train the chunk taggers with a common tag set across multiple languages which does not need to maintain language

specific unification rules due to the differences in Treebank annotation guidelines.

The following compares the definition of tags used in the English Wall Street Journal (WSJ) [2] and the Chinese Penn Treebank (CTB) 7.0 [12] that are used in our experiment.

(i) The Phrasal Tag of Chinese. In the present corpus CTB7, each chunk is labeled with one syntactic category. A tag set of 13 categories is used to represent shallow parsing structural information as follows.

ADJP—adjective phrase: phrase headed by an adjective.

ADVP—adverbial phrase: phrasal category headed by an adverb.

CLP—classifiers phrase: for example, (QP (CD 一) (CLP (M 系列))) (a series).

DP—determiner phrase: in CTB7 corpus, a DP is a modifier of the NP if it occurs within a NP. For example, (NP (DP (DT 任何) (NP (NN 人))) (any people).

DNP—phrase formed by XP plus (DEG 的) ('s) that modifies a NP; the XP can be an ADJP, DP, QP, NP, PP, or LCP.

DVP—phrase formed by XP plus "地 (-ly)" that modifies a VP.

FRAG—fragment used to mark fragmented elements that cannot be built into a full structure by using null categories.

LCP—used to mark phrases formed by a localizer and its complement. For example, (LCP (NP (NR 皖南) (NN 事变)) (LC 中)) (in the Southern Anhui Incident).

LST—list marker: numbered list, letters, or numbers which identify items in a list and their surrounding punctuation is labeled as LST.

NP—noun phrases: phrasal category that includes all constituents depending on a head noun.

PP—preposition phrase: phrasal category headed by a preposition.

QP—quantifier phrase: used with NP, for example, (QP (CD 五百) (CLP (M 辆))) (500).

VP—verb phrase: phrasal category headed by a verb.

(ii) The Phrasal Tag of English. In the respect English WSJ corpus, there are just eight categories: NP (noun phrase), PP (prepositional phrase), VP (verb phrase), ADVP (adverb phrase), ADJP (adjective phrase), SBAR (subordinating conjunction), PRT (particle), and INTJ (interjection).

As we can see, the criterion of Chinese CTB7 has a lot of significant differences from English WSJ corpus. Unlike the English Treebank, the fragment of Chinese continents is normally smaller. A chunk which aligned to a base-NP phrase in the English side could be divided into a QP tag, a CLP tag, and an NP tag. For example, after chunking, "五百辆车 (500 cars)" will be tagged as (NP (QP (CD 五百) (CLP (M 辆))) (NN 车)). But at the English side, "500 cars" will be just denoted with an NP tag. This nonuniform standard will result in a mismatch projection during the alignment phase and the difficulty of evaluation in the cross-lingual setting. To fix these problems, mappings are applied in the universal chunking tag set. The categoryies CLP, DP, and QP are merged into an NP phrase. That is to say, the phrase such as (NP (QP (CD 五百) (CLP (M 辆))) (NN 车)), which corresponds to the English NP chunk "500 cars," will be assigned with an NP tag in the universal chunk tag. Additionally, the phrases DNP and DVP could be included in the ADJP and ADVP tags, respectively.

Table 2: The description of universal chunk tags.

Tag	Description	Words	Example
NP	Noun phrase	DET + ADV + ADJ + NOUN	The strange birds
PP	Preposition phrase	TO + IN	In between
VP	Verb phrase	ADV + VB	Was looking
ADVP	Adverb phrase	ADV	Also
ADJP	Adjective phrase	CONJ + ADV + ADJ	Warm and cozy
SBAR	Subordinating conjunction	IN	Whether or not

On the English side, the occurrence of INTJ is 0% according to statistics. This evidence shows that we can ignore the INTJ chunk tag. Additionally, the words belong to a PRT tag that is always regarded as a VP in Chinese. For example, (PRT (RP up)) (NP (DET the) (NNS stairs)) is aligned to "上楼梯" in Chinese where "上" is a VP.

4.1.2. IBO2 Chunk Tag. There are many ways to encode the phrase structure of a chunk tag, such as IBO1, IBO2, IOE1, and IOE2 [13]. The one used in this study is the IBO2 encoding. This format ensures that all initial words of different base phrases will receive a B tag, which is able to identify the boundaries of each chunk. In addition, two boundary types are defined as follows:

(i) B-*X*: represents the beginning of a chunk *X*;

(ii) I-*X*: indicates a noninitial word in a phrase *X*;

(iii) O: any words that are out of the boundary of any chunks.

Hence, the input Chinese sentence can be represented using these notations as follows:

去年 (NP-B) 实现 (VP-B) 进出口 (NP-B) 总值 (NP-I) 达 (VP-B) 一千零九十八点二亿 (NP-B) 美元 (NP-I) 。(O)

Based on the above discussion, a tag set that consists of six universal chunk categories is proposed as shown in Table 2. While there are a lot of controversies about universal tags and what the exact tag set should be used, these six

chunk categories are sufficient to embrace the most frequent chunk tags that exist in English and Chinese. Furthermore, a mapping from fine-grained chunk tags for these two kind languages to the universal chunk set has been developed. Hence, a dataset consisting of common chunk tag for English and Chinese is achieved by combining with the original Treebank data and the universal chunk tag set and mapping between these two languages.

4.2. Chunk Induction. The label distribution of Chinese word types x can be computed by marginalizing the chunk tag distribution of trigram types $c_i = x_{-1}x_0x_{+1}$ over the left and right context as follows:

$$p(y \mid x) = \frac{\sum_{x_{-1}x_{+1}} q_i(y)}{\sum_{x_{-1}x_{+1},y'} q_i(y')}. \quad (6)$$

Then a set of possible tags $t_x(y)$ is extracted through a way that eliminates labels whose probability is below a threshold value τ:

$$t_x(y) = \begin{cases} 1 & \text{if } p(y \mid x) \geq \tau \\ 0 & \text{otherwise.} \end{cases} \quad (7)$$

The way how to choose τ will be described in Section 5. Additionally, the vector t_x will cover every word in the Chinese trigram types and will be employed as features in the unsupervised Chinese shallow parsing.

Similar to the work of [14, 15], our chunk induction model is constructed on the feature-based hidden Markov model (HMM). A chunking HMM generates a sequence of words in order. In each generation step, based on the current chunk tag z_i conditionally, an observed word x_i and a hidden successor chunk tag z_{i+1} are generated independently. There are two types of conditional distributions in the model, emission and transition probabilities, which are both multinomial probability distributions. Given a sentence x and a state sequence z, a first-order Markov model defines a joint distribution as

$$\begin{aligned} P_\theta = (X = x, Z = z) &= P_\theta(Z_1 = z_1) \\ &\cdot \prod_{i=1}^{|x|} P_\theta(Z_{i+1} = z_{i+1} \mid Z_i = z_i) \\ &\cdot P_\theta(X_i = x_i \mid Z_i = z_i), \end{aligned} \quad (8)$$

where the second part represents the transition probability and the following part is emission probability, which is different from the conventional Markov model where the feature-based model replaces the emission part with a log-linear model, such that

$$P_\theta(X = x, Z = z) = \frac{\exp \theta^{\mathrm{T}} f(x, z)}{\sum_{x' \in \mathrm{Val}(X)} \exp \theta^{\mathrm{T}} f(x, z)}, \quad (9)$$

which corresponds to the entire Chinese vocabulary.

The advantage of using this locally normalized log-linear model is the ability of looking at various aspects of the observation x incorporating features of the observations. Then the model is trained by applying the following objective function:

$$\mathscr{L}(\theta) = \sum_{i=1}^{N} \log \sum_{Z} P(\theta)\left(X = x^{(i)}, Z = z^{(i)}\right) - C\|\theta\|^2. \quad (10)$$

It is worth noting that this function includes marginalizing out all the sentence x's and all possible configurations z, which leads to a nonconvex objective. We optimize this objective function by using L-BFGS, a quasi-Newton method proposed by Liu and Nocedal [16]. The evidences of past experiments show that this direct gradient algorithm performed better than using a feature-enhanced modification of the EM (expectation-maximization).

Moreover, this state-of-the-art model also has an advantage that makes it easy to experiment with various ways of incorporating the constraint feature into the log-linear model. This feature function f_t consists of the relevant information extracted from the smooth graph and eliminates the hidden states which are inconsistent with the threshold vector t_x.

Table 3: Feature template used in unsupervised chunking.

Basic:	(x=, z =)
Contains digit:	check if x contains digit and conjoin with z (contains digit(x)=, z =)
Contains hypen:	contains hypen(x)=, z =
Suffix:	indicator features for character suffixes of up to length 1 present in x
Prefix:	indicator features for character prefixes of up to length 1 present in x
Pos tag:	indicator feature for word POS assigned to x

4.3. Unsupervised Chunking Features. The feature selection process in the unsupervised chunking model does affect the identification result. Feature selection is the task to identify an optimal feature subset that best represents the underlying classification problem, through filtering the irrelevant and redundant ones. Unlike the CoNLL-2000, we shared supervised chunking task which is using WSJ (the Wall Street Journal) corpus as background knowledge to construct the chunking model. Our corpus is composed of only words which do not contain any chunk information that enables us to form an optimal feature subset for the underlying chunk pattern. We use the set of features as the following feature templates. These are all coarse features on emission contexts that are extracted from words with certain orthographic properties. Only the basic feature is used for transitions. For any emission context with word x and tag z, we construct the following feature templates as shown in Table 3.

Box 1 illustrates the example about the feature subset. The feature set of each word in a sentence is a vector of 6 dimensions, which are the values of the corresponding features and the label indicates which kind of chunk label should the word belong to.

Sentence: 中国/NR/NP-B 建筑业/NN/NP-I 对/P/PP-B 外/NN/NP-B 开放/VV/VP-B 呈现/VV/VP-B 新/JJ/NP-B 格局/NN/NP-I 。
Word: 建筑业
Instance: (建筑业, NP-I), (N, NP-I), (N, NP-I), (建), (业), NN

Box 1: Example of feature template.

5. Experiments and Results

Before presenting the results of our experiments, the evaluation metrics, the datasets, and the baselines used for comparisons are firstly described.

5.1. Evaluation Metrics. The metrics for evaluating NP chunking model constitute precision rate, recall rate, and their harmonic mean F_1-score. The tagging accuracy is defined as

$$\text{Tagging Accuracy} = \frac{\text{The number of correct tagged words}}{\text{The number of words}}. \quad (11)$$

Precision measures the percentage of labeled NP chunks that are correct. The number of correct tagged words includes both the correct boundaries of NP chunk and the correct label. The precision is therefore defined as

$$\text{Precision} = \frac{\text{The number of correct proposed tagged words}}{\text{The number of correct chunk tags}}. \quad (12)$$

Recall measures the percentage of NP chunks presented in the input sentence that are correctly identified by the system. Recall is depicted as below

$$\text{Recall} = \frac{\text{The number of correct proposed tagged words}}{\text{The number of current chunk tags}}. \quad (13)$$

The F-measure illustrates a way to combine previous two measures into one metric. The formula of F-sore is defined as

$$F_{\beta\text{-score}} = \frac{\left(\beta^2 + 1\right) \times \text{Recall} \times \text{Precision}}{\beta^2 \times \text{Recall} + \text{Precision}}, \quad (14)$$

where β is a parameter that weights the importance of recall and precision; when $\beta = 1$, precision and recall are equally important.

5.2. Dataset. To facilitate bilingual graph construction and unsupervised chunking experiment, two kinds of data sets are utilized in this work: (1) monolingual Treebank for Chinese chunk induction and (2) large amounts of parallel corpus of English and Chinese for bilingual graph construction and label propagation. For the monolingual Treebank data we rely on Penn Chinese Treebank 7.0 (CTB7) [12]. It consists of over one million words of annotated and parsed text from Chinese newswire, magazines, various broadcast news, and broadcast conversation programs, web newsgroups, and weblogs. The parallel data came from the UM corpus [17], which contains 100,000 pair of English-Chinese sentences. The training and testing sets are defined for the experiment and their corresponding statistic information is shown in Table 4. In addition, Table 4 also shows the detailed information of data used for bilingual graph construction including the parallel corpus for the tag projection and the label propagation of Chinese trigram.

Table 4: (a) English-Chinese parallel corpus. (b) Graph instance. (c) Chinese unlabeled dataset: CTB7 corpora.

(a)

Number of sentence pairs	Number of seeds	Number of words
10,000	27,940	31,678

(b)

Number of sentences	Number of vertices
17,617	185,441

(c)

Dataset	Source	Number of sentences
Training dataset	Xinhua 1–321	7,617
Testing dataset	Xinhua 363–403	912

5.3. Chunk Tagset and HMM States. As described, the universal chunk tag set is employed in our experiment. This set U consists of the following six coarse-grained tags: NP (noun phrase), PP (prepositional phrase), VP (verb phrase), ADVP (adverb phrase), ADJP (adjective phrase), and SBAR (subordinating conjunction). Although there might be some controversies about the exact definition of such a tag set, these 6 categories cover the most frequent chunk tags that exist in one form or another in both of English and Chinese corpora.

For each kind of languages under consideration, a mapping from the fine-grained language specific chunk tags in the Treebank to the universal chunk tags is provided. Hence, the model of unsupervised chunking is trained on the datasets labeled with the universal chunk tags.

Table 5: Chunk tagging evaluation results for various baselines and proposed graph-based model.

Tag	Feature-HMM			Projection			Graph-based		
	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
NP	0.60	0.67	0.63	0.67	0.72	0.68	0.81	0.82	0.81
VP	0.56	0.51	0.53	0.61	0.57	0.59	0.78	0.74	0.76
PP	0.36	0.28	0.32	0.44	0.31	0.36	0.60	0.51	0.55
ADVP	0.40	0.46	0.43	0.45	0.52	0.48	0.64	0.68	0.66
ADJP	0.47	0.53	0.50	0.48	0.58	0.51	0.67	0.71	0.69
SBAR	0.00	0.00	0.00	0.50	1.0	0.66	0.50	1.0	0.66
All	0.49	0.51	0.50	0.57	0.62	0.59	**0.73**	**0.75**	**0.74**

In this work, the number of latent HMM states for Chinese is fixed to be a constant, which is the number of the universal chunk tags.

5.4. Various Models. To comprehensively probe our approach with a thorough analysis, we evaluated two baselines in addition to our graph-based algorithm. We were intentionally lenient with these two baselines.

(i) Feature-HMM. The first baseline is the vanilla feature-HMM proposed by [9], which apply L-BFGS to estimate the model parameters and use a greedy many-to-1 mapping for evaluation.

(ii) Projection. The direct projection serves as a second baseline. It incorporates the bilingual information by projecting chunk tags from English side to the aligned Chinese texts. Several rules were added to fix the unaligned problem. We trained a supervised feature-HMM with the same number of sentences from the bitext as it is in the Treebank.

(iii) Our Graph-Based Model. The full model uses both stages of label propagation (3) before extracting the constrain features. As a result, the label distribution of all Chinese word types was added in the constrain features.

5.5. Experimental Setup. The experimental platform is implemented based on two toolkits: Junto [18] and Ark [14, 15]. Junto is a Java-based label propagation toolkit for the bilingual graph construction. Ark is a Java-based package for the feature-HMM training and testing.

In the feature-HMM training, we need to set a few hyperparameters to minimize the number of free parameters in the model. $C = 1.0$ was used as regularization constant in (10) and trained L-BFGS for 1,000 iterations. Several threshold values for τ to extract the vector t_x were tried and it was found that 0.2 works best. It indicates that $\tau = 0.2$ could be used for Chinese trigram types.

5.6. Results. Table 5 shows the complete set of results. As expected, the projection baseline is better than the feature-HMM for being able to benefit from the bilingual information. It greatly improves upon the monolingual baseline by 12% on F_1-score. Comparing among the unsupervised approaches, the feature-HMM achieves only 50% of F_1-score on the universal chunk tags. Overall, the graph-based model outperforms these two baselines. That is to say, the improvement of feature-HMM with the graph-based setting is statistically significant with respect to other models. Graph-based modal performs 24% better than the state-of-the-art feature-HMM and 12% better than the direct projection model.

6. Conclusion

This paper presents an unsupervised graph-based Chinese chunking by using label propagation for projecting chunk information across languages. Our results suggest that it is possible for unlabeled text to learn accurate chunk tags from the bitext, the data which has parallel English text. In the feature, we propose an alternative graph-based unsupervised approach on chunking for languages that are lacking ready-to-use resource.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors are grateful to the Science and Technology Development Fund of Macau and the Research Committee of the University of Macau (Grant nos. MYRG076(Y1-L2)-FST13-WF and MYRG070(Y1-L2)-FST12-CS) for the funding support for their research.

References

[1] R. Koeling, "Chunking with maximum entropy models," in *Proceedings of the 2nd Workshop on Learning Language in Logic and the 4th Conference on Computational Natural Language Lerning*, vol. 7, pp. 139–141, 2000.

[2] M. P. Marcus, M. A. Marcinkiewicz, and B. Santorini, "Building a large annotated corpus of English: the Penn Treebank," *Computational Linguistics*, vol. 19, no. 2, pp. 313–330, 1993.

[3] D. Yarowsky and G. Ngai, "Inducing multilingual POS taggers and NP bracketers via robust projection across aligned corpora," in *Proceedings of the 2nd Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL '01)*, pp. 200–207, 2001.

[4] X. Zhu, Z. Ghahramani, and J. Lafferty, "Semi-supervised learning using Gaussian fields and harmonic functions," in *Proceedings of the 20th International Conference on Machine Learning (ICML '03)*, vol. 3, pp. 912–919, August 2003.

[5] Y. Altun, M. Belkin, and D. A. Mcallester, "Maximum margin semi-supervised learning for structured variables," in *Advances in Neural Information Processing Systems*, pp. 33–40, 2005.

[6] A. Subramanya, S. Petrov, and F. Pereira, "Efficient graph-based semi-supervised learning of structured tagging models," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 167–176, October 2010.

[7] F. J. Och and H. Ney, "A systematic comparison of various statistical alignment models," *Computational Linguistics*, vol. 29, no. 1, pp. 19–51, 2003.

[8] A. Subramanya and J. Bilmes, "Semi-supervised learning with measure propagation," *The Journal of Machine Learning Research*, vol. 12, pp. 3311–3370, 2011.

[9] D. Das and N. A. Smith, "Semi-supervised frame-semantic parsing for unknown predicates," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pp. 1435–1444, June 2011.

[10] S. Baluja, R. Seth, D. S. Sivakumar et al., "Video suggestion and discovery for you tube: taking random walks through the view graph," in *Proceedings of the 17th International Conference on World Wide Web*, pp. 895–904, April 2008.

[11] S. Petrov, D. Das, and R. McDonald, "A universal part-of-speech tagset," 2011, http://arxiv.org/abs/1104.2086.

[12] N. Xue, F. Xia, F.-D. Chiou, and M. Palmer, "The Penn Chinese TreeBank: phrase structure annotation of a large corpus," *Natural Language Engineering*, vol. 11, no. 2, pp. 207–238, 2005.

[13] E. F. Tjong, K. Sang, and H. Déjean, "Introduction to the CoNLL- shared task: clause identification," in *Proceedings of the Workshop on Computational Natural Language Learning*, vol. 7, p. 8, 2001.

[14] T. Berg-Kirkpatrick, A. Bouchard-Côté, J. de Nero, and D. Klein, "Painless unsupervised learning with features," in *Proceedings of the Human Language Technologies Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 582–590, Los Angeles, Calif, USA, June 2010.

[15] A. B. Goldberg and X. Zhu, "Seeing stars when there aren't many stars: graph-based semi-supervised learning for sentiment categorization," in *Proceedings of the 1st Workshop on Graph Based Methods for Natural Language Processing*, pp. 45–52, 2006.

[16] D. C. Liu and J. Nocedal, "On the limited memory BFGS method for large scale optimization," *Mathematical Programming B*, vol. 45, no. 3, pp. 503–528, 1989.

[17] L. Tian, F. Wong, and S. Chao, "An improvement of translation quality with adding key-words in parallel corpus," in *Proceedings of the International Conference on Machine Learning and Cybernetics (ICMLC '10)*, vol. 3, pp. 1273–1278, July 2010.

[18] P. P. Talukdar and F. Pereira, "Experiments in graph-based semi-supervised learning methods for class-instance acquisition," in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL '10)*, pp. 1473–1481, July 2010.