Ambiguity Resolution for Vt-N Structures in Chinese

Yu-Ming Hsieh[1,2] **Jason S. Chang**[2] **Keh-Jiann Chen**[1]
[1] Institute of Information Science, Academia Sinica, Taiwan
[2] Department of Computer Science, National Tsing-Hua University, Taiwan
morris@iis.sinica.edu.tw, jason.jschang@gmail.com
kchen@iis.sinica.edu.tw

Abstract

The syntactic ambiguity of a transitive verb (Vt) followed by a noun (N) has long been a problem in Chinese parsing. In this paper, we propose a classifier to resolve the ambiguity of Vt-N structures. The design of the classifier is based on three important guidelines, namely, adopting linguistically motivated features, using all available resources, and easy integration into a parsing model. The linguistically motivated features include semantic relations, context, and morphological structures; and the available resources are treebank, thesaurus, affix database, and large corpora. We also propose two learning approaches that resolve the problem of data sparseness by auto-parsing and extracting relative knowledge from large-scale unlabeled data. Our experiment results show that the Vt-N classifier outperforms the current PCFG parser. Furthermore, it can be easily and effectively integrated into the PCFG parser and general statistical parsing models. Evaluation of the learning approaches indicates that world knowledge facilitates Vt-N disambiguation through data selection and error correction.

1 Introduction

In Chinese, the structure of a transitive verb (Vt) followed by a noun (N) may be a verb phrase (VP), a noun phrase (NP), or there may not be a dependent relation, as shown in (1) below. In general, parsers may prefer VP reading because a transitive verb followed by a noun object is normally a VP structure. However, Chinese verbs can also modify nouns without morphological inflection, e.g., 養殖/*farming* 池/*pond*. Consequently, parsing Vt-N structures is difficult because it is hard to resolve such ambiguities without prior knowledge. The following are some typical examples of various Vt-N structures:

1)
解決/*solve* 問題/*problem* → VP
解決/*solving* 方案/*method* → NP
解決/*solve* 人類/*mankind* (問題/*problem*)→None

To find the most effective disambiguation features, we need more information about the Vt-N → NP construction and the semantic relations between Vt and N. Statistical data from the Sinica Treebank (Chen et al., 2003) indicates that 58% of Vt-N structures are verb phrases, 16% are noun phrases, and 26% do not have any dependent relations. It is obvious that the semantic relations between a Vt-N structure and its context information are very important for differentiating between dependent relations. Although the verb-argument relation of VP structures is well understood, it is not clear what kind of semantic relations result in NP structures. In the next sub-section, we consider three questions: What sets of nouns accept verbs as their modifiers? Is it possible to identify the semantic types of such pairs of verbs and nouns? What are their semantic relations?

1.1 Problem Analysis

Analysis of the instances of NP(Vt-N) structures in the Sinica Treebank reveals the following four types of semantic structures, which are used in the design of our classifier.

Type 1. Telic(Vt) + Host(N): Vt denotes the telic function (purpose) of the head noun N, e.g.,

Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 928–937,
October 25-29, 2014, Doha, Qatar.

研究/*research* 工具/*tool*; 探測/*explore* 機/*machine*; 賭/*gamble* 館/*house*; 搜尋/*search* 程式/*program*. The telic function must be a salient property of head nouns, such as tools, buildings, artifacts, organizations and people. To identify such cases, we need to know the types of nouns which take telic function as their salient property. Furthermore, many of the nouns are monosyllabic words, such as 員/*people*, 器/*instruments*, 機/*machines*.

Type 2. Host-Event(Vt) + Attribute(N): Head nouns are attribute nouns that denote the attributes of the verb, e.g., 研究/*research* 方法/*method* (*method of research*); 攻擊/*attack* 策略/*strategy* (*attacking strategy*); 書寫/*write* 內容/*context* (*context of writing*); 賭/*gamble* 規/*rule* (*gambling rules*). An attribute noun is a special type of noun. Semantically, attribute nouns denote the attribute types of objects or events, such as *weight*, *color*, *method*, and *rule*. Syntactically, attribute nouns do not play adjectival roles (Liu, 2008). By contrast, object nouns may modify nouns. The number of attributes for events is limited. If we could discover all event-attribute relations, then we can solve this type of construction.

Type 3. Agentive + Host: There is only a limited number of such constructions and the results of the constructions are usually ambiguous, e.g., 炒飯/*fried rice* (NP), 叫聲/*shouting sound*. The first example also has the VP reading.

Type 4. Apposition + Affair: Head nouns are event nouns and modifiers are verbs of apposition events, e.g. 追撞/*collide* 事故/*accident*, 破壞/*destruct* 運動/*movement*, 憤恨/*hate* 行為/*behavior*. There is finite number of event nouns.

Furthermore, when we consider verbal modifiers, we find that verbs can play adjectival roles in Chinese without inflection, but not all verbs play adjectival roles. According to Chang et al. (2000) and our observations, adjectival verbs are verbs that denote event types rather than event instances; that is, they denote a class of events which that are concepts in an upper-level ontology. One important characteristic of adjectival verbs is that they have conjunctive morphological structures, i.e., the words are conjunct with two nearly synonymous verbs, e.g., 研/*study* 究/*search* (*research*), 探/*explore* 測/*detect* (*explore*), and 搜/*search* 尋/*find* (*search*). Therefore, we need a morphological classifier that can detect the conjunctive morphological structure of a verb by checking the semantic parity of two morphemes of the verb.

Based on our analysis, we designed a Vt-N classifier that incorporates the above features to solve the problem. However, there is a data sparseness problem because of the limited size of the current Treebank. In other words, Treebank cannot provide enough training data to train a classifier properly. To resolve the problem, we should mine useful information from all available resources.

The remainder of this paper is organized as follows. Section 2 provides a review of related works. In Section 3, we describe the disambiguation model with our selected features, and introduce a strategy for handling unknown words. We also propose a learning approach for a large-scale unlabeled corpus. In Section 4, we report the results of experiments conducted to evaluate the proposed Vt-N classifier on different feature combinations and learning approaches. Section 5 contains our concluding remarks.

2 Related Work

Most works on V-N structure identification focus on two types of relation classification: modifier-head relations and predicate-object relations (Wu, 2003; Qiu, 2005; Chen, 2008; Chen et al., 2008; Yu et al., 2008). They exclude the independent structure and conjunctive head-head relation, but the cross-bracket relation does exist between two adjacent words in real language. For example, if "遍佈/*all over* 世界/*world*" was included in the short sentence "遍佈/*all over* 世界/*world* 各國/*countries*", it would be an independent structure. A conjunctive head-head relation between a verb and a noun is rare. However, in the sentence "服務 設備 都 甚 周到" (Both service and equipment are very thoughtful.), there is a conjunctive head-head relation bewteen the verb 服務/*service* and the noun 設備/*equipment*. Therefore, we use four types of relations to describe the V-N structures in our experiments. The symbol 'H/X' denotes a predicate-object relation; 'X/H' denotes a modifier-head relation; 'H/H' denotes a conjunctive head-head relation; and 'X/X' denotes an independent relation.

Feature selection is an important task in V-N disambiguation. Hence, a number of studies have suggested features that may help resolve the ambiguity of V-N structures (Zhao and Huang, 1999; Sun and Jurafsky, 2003; Chiu et al., 2004; Qiu, 2005; Chen, 2008). Zhao and Huang used lexicons, semantic knowledge, and word length in-

formation to increase the accuracy of identification. Although they used the Chinese thesaurus CiLin (Mei et al., 1983) to derive lexical semantic knowledge, the word coverage of CiLin is insufficient. Moreover, none of the above papers tackle the problem of unknown words. Sun and Jurafsky exploit the probabilistic rhythm feature (i.e., the number of syllables in a word or the number of words in a phrase) in their shallow parser. Their results show that the feature improves the parsing performance, which coincides with our analysis in Section 1.1. Chiu et al.'s study shows that the morphological structure of verbs influences their syntactic behavior. We follow this finding and utilize the morphological structure of verbs as a feature in the proposed Vt-N classifier. Qiu's approach uses an electronic syntactic dictionary and a semantic dictionary to analyze the relations of V-N phrases. However, the approach suffers from two problems: (1) low word coverage of the semantic dictionary and (2) the semantic type classifier is inadequate. Finally, Chen proposed an automatic VN combination method with features of verbs, nouns, context, and the syllables of words. The experiment results show that the method performs reasonably well without using any other resources.

Based on the above feature selection methods, we extract relevant knowledge from Treebank to design a Vt-N classifier. However we have to resolve the common problem of data sparseness. Learning knowledge by analyzing large-scale unlabeled data is necessary and proved useful in previous works (Wu, 2003; Chen et al., 2008; Yu et al., 2008). Wu developed a machine learning method that acquires verb-object and modifier-head relations automatically. The mutual information scores are then used to prune verb-noun whose scores are below a certain threshold. The author found that accurate identification of the verb-noun relation improved the parsing performance by 4%. Yu et al. learned head-modifier pairs from parsed data and proposed a head-modifier classifier to filter the data. The filtering model uses the following features: a PoS-tag pair of the head and the modifier; the distance between the head and the modifier; and the presence or absence of punctuation marks (e.g., commas, colons, and semi-colons) between the head and the modifier. Although the method improves the parsing performance by 2%, the filtering model obtains limited data; the recall rate is only 46.35%. The authors also fail to solve the problem of Vt-N ambiguity.

Our review of previous works and the observations in Section 1.1 show that lexical words, semantic information, the syllabic length of words, neighboring PoSs and the knowledge learned from large-scale data are important for Vt-N disambiguation. We consider more features for disambiguating Vt-N structures than previous studies. For example, we utilize (1) four relation-classification in a real environment, including 'X/H', 'H/X', 'X/X' and 'H/H' relations; (2) unknown word processing of Vt-N words (including semantic type predication and morph-structure predication); (3) unsupervised data selection (a simple and effective way to extend knowledge); and (4) supervised knowledge correction, which makes the extracted knowledge more useful.

3 Design of the Disambiguation Model

The disambiguation model is a Vt-N relation classifier that classifies Vt-N relations into 'H/X' (predicate-object relations), 'X/H' (modifier-head relations), 'H/H' (conjunctive head-head relations), or 'X/X' (independent relations). We use the Maximum Entropy toolkit (Zhang, 2004) to construct the classifier. The advantage of using the Maximum Entropy model is twofold: (1) it has the flexibility to adjust features; and (2) it provides the probability values of the classification, which can be easily integrated into our PCFG parsing model.

In the following sections, we discuss the design of our model for feature selection and extraction, unknown word processing, and world knowledge learning.

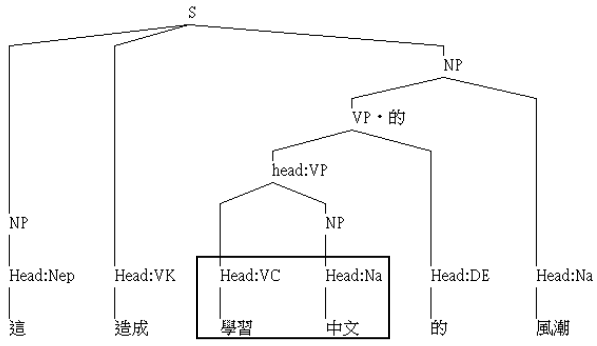
3.1 Feature Selection and Extraction

We divide the selected features into five groups: PoS tags of Vt and N, PoS tags of the context, words, semantics, and additional information. Table 1 shows the feature types and symbol notations. We use symbols of t_1 and t_2 to denote the PoS of Vt and N respectively. The context feature is neighboring PoSs of Vt and N: the symbols of t_{-2} and t_{-1} represent its left PoSs, and the symbole t_3 and t_4 represent its right PoSs. The semantic feature is the lexicon's semantic type extracted from E-HowNet sense expressions (Huang et al., 2008). For example, the E-HowNet expression of "車輛/*vehicles*" is {LandVehicle|車:quantity={mass|眾}}, so its semantic type is {LandVehicle|車}. We discuss the model's performance with different feature combinations in Section 4.

Feature	Feature Description
PoS	PoS of Vt and N t_1; t_2
Context	Neighboring PoSs t_{-2}; t_{-1}; t_3; t_4
Word	Lexical word w_1; w_2
Semantic	Semantic type of word st_1; st_2
Additional Information	Morphological structure of verb *Vmorph*
	Syllabic length of noun *Nlen*

Table 1. The features used in the Vt-N classifier

The example in Figure 1 illustrates feature labeling of a Vt-N structure. First, an instance of a Vt-N structure is identified from Treebank. Then, we assign the semantic type of each word without considering the problem of sense ambiguity for the moment. This is because sense ambiguities are partially resolved by PoS tagging, and the general problem of sense disambiguation is beyond the scope of this paper. Furthermore, Zhao and Huang (1999) demonstrated that the retained ambiguity does not have an adverse impact on identification. Therefore, we keep the ambiguous semantic type for future processing.



zhe zaochen xuexi zhongwen DE fongchao
this cause learn Chinese trend
"This causes the trend of learning Chinese."

Figure 1. An example of a tree with a Vt-N structure

Table 2 shows the labeled features for "學習/*learn* 中文/*Chinese*" in Figure 1. The column *x* and *y* describe relavant features in "學習/*learn*" and "中文/*Chinese*" respectively. Some features are not explicitly annotated in the Treebank, e.g., the semantic types of words and the morphological structure of verbs. We propose labeling methods for them in the next sub-section.

Feature Type	x	y
Word	w_1=學習	w_2=中文
PoS	t_1=VC	t_2=Na
Semantic	st_1=study|學習	st_2=language|語言
Context	t_{-2}=Nep; t_{-1}=VK; t_3=DE; t_4=Na	
Additional Information	Vmorph=VV	Nlen=2
Relation Type	rt = H/X	

Table 2. The feature lables of Vt-N pair in Figure 1

3.2 Unknown Word Processing

In Chinese documents, 3% to 7% of the words are usually unknown (Sproat and Emerson, 2003). By 'unknown words', we mean words not listed in the dictionary. More specifically, in this paper, unknown words means words without semantic type information (i.e., E-HowNet expressions) and verbs without morphological structure information. Therefore, we propose a method for predicting the semantic types of unknown words, and use an affix database to train a morph-structure classifier to derive the morphological structure of verbs.

Morph-Structure Predication of Verbs: We use data analyzed by Chiu et al. (2004) to develop a classifier for predicating the morphological structure of verbs. There are four types of morphological structures for verbs: the coordinating structure (VV), the modifier-head structure (AV), the verb-complement structure (VR), and the verb-object structure (VO). To classify verbs automatically, we incorporate three features in the proposed classifier, namely, the lexeme itself, the prefix and the suffix, and the semantic types of the prefix and the suffix. Then, we use training data from the affix database to train the classifier. Table 3 shows an example of the unknown verb "傳播到/disseminate" and the morph-structure classifier shows that it is a 'VR' type.

Feature	Feature Description
Word=傳播到	Lexicon
PW=傳播	Prefix word
PWST={disseminate|傳播}	Semantic Type of Prefix Word 傳播
SW=到	Suffix Word
SWST={Vachieve|達成}	Semantic Type of Suffix Word 到

Table 3. An example of an unknown verb and feature templates for morph-structure predication

Semantic Type Provider: The system exploits *WORD*, *PoS*, *affix* and *E-HowNet* information to obtain the semantic types of words (see Figure 2). If a word is known and its PoS is given, we can usually find its semantic type by searching the E-HowNet database. For an unknown word, the semantic type of its head morpheme is its semantic type; and the semantic type of the head morpheme is obtained from E-HowNet[1]. For example, the unknown word "傳播到/*disseminate*", its prefix word is "傳播/*disseminate*" and we learn that its semantic type is {disseminate|傳播} from E-HowNet. Therefore, we assign {disseminate|傳播} as the semantic type of "傳播到/*disseminate*". If the word or head morpheme does not exist in the affix database, we assign a general semantic type based on its PoS, e.g., nouns are {thing|萬物} and verbs are {act|行動}. In this matching procedure, we may encounter multiple matching data of words and affixes. Our strategy is to keep the ambiguous semantic type for future processing.

Input: *WORD, PoS*
Output: *Semantic Type (ST)*

```
procedure STP(WORD, PoS)
    (* Initial Step *)
    ST := null;
    (* Step 1: Known word *)
    if WORD already in E-HowNet then
        ST := EHowNet(WORD, PoS);
    else if WORD in Affix database then
        ST := EHowNet(affix of WORD, PoS);
    (* Step 2 : Unknown word *)
    if ST is null and PoS is 'Vt' then
        ST := EHowNet(prefix of WORD, PoS);
    else if ST is null and PoS is 'N' then
        ST := EHowNet(suffix of WORD, PoS);
    (* Step 3 : default *)
    if ST is null and PoS is 'Vt' then
        ST := 'act|行動';
    else if ST is null and PoS is 'N' then
        ST := 'thing|萬物'
    (* Finally *)
    STP := ST;
end;
```

Figure 2. The Pseudo-code of the Semantic Type Predication Algorithm.

[1] The E-HowNet function in Figure 2 will return a null ST value where words do not exist in E-HowNet or Affix database.

3.3 Learning World Knowledge

Based on the features discussed in the previous sub-section, we extract prior knowledge from Treebank to design the Vt-N classifier. However, the training suffers from the data sparseness problem. Furthermore most ambiguous Vt-N relations are resolved by common sense knowledge that makes it even harder to construct a well-trained system. An alternative way to extend world knowledge is to learn from large-scale unlabeled data (Wu, 2003; Chen et al., 2008; Yu et al., 2008). However, the unsupervised approach accumulates errors caused by automatic annotation processes, such as word segmentation, PoS tagging, syntactic parsing, and semantic role assignment. Therefore, how to extract useful knowledge accurately is an important issue.

To resolve the error accumulation problemm, we propose two methods: unsupervised NP selection and supervised error correction. The NP selection method exploits the fact that an intransitive verb followed by a noun can only be interpreted as an NP structure, not a VP structure. It is easy to find such instances with high precision by parsing a large corpus. Based on the selection method, we can extend contextual knowledge about NP(V+N) and extract nouns that take adjectival verbs as modifiers. The error correction method involves a small amount of manual editing in order to make the data more useful and reduce the number of errors in auto-extracted knowledge. The rationale is that, in general, high frequency Vt-N word-bigram is either VP or NP without ambiguity. Therefore, to obtain more accurate training data, we simply classify each high frequency Vt-N word bigram into a unique correct type without checking all of its instances. We provide more detailed information about the method in Section 4.3.

4 Experiments and Results

4.1 Experimental Setting

We classify Vt-N structures into four types of syntactic structures by using the bracketed information (tree structure) and dependency relation (head-modifier) to extract the Vt-N relations from treebank automatically. The resources used in the experiments as follows.

Treebank: The Sinica Treebank contains 61,087 syntactic tree structures with 361,834 words. We extracted 9,017 instances of Vt-N structures from the corpus. Then, we randomly

selected 1,000 of the instances as test data and used the remainder (8,017 instances) as training data. Labeled information of word segmentation and PoS-tagging were retained and utilized in the experiments.

E-HowNet: E-HowNet contains 99,525 lexical semantic definitions that provide information about the semantic type of words. We also implement the semantic type predication algorithm in Figure 2 to generate the semantic types of all Vt and N words, including unknown words.

Affix Data: The database include 13,287 examples of verbs and 27,267 examples of nouns, each example relates to an affix. The detailed statistics of the verb morph-structure categorization are shown in Table 4. The data is used to train a classifier to predicate the morph-structure of verbs. We found that verbs with a conjunctive structure (VV) are more likely to play adjectival roles than the other three types of verbs. The classifier achieved 87.88% accuracy on 10-fold cross validation of the above 13,287 verbs.

	VV	VR	AV	VO
Prefix	920	2,892	904	662
Suffix	439	7,388	51	31

Table 4. The statistics of verb morph-structure categorization

Large Corpus: We used a Chinese parser to analyze sentence structures automatically. The auto-parsed tree structures are used in Experiment 2 (described in the Sub-section 4.3). We obtained 1,262,420 parsed sentences and derived 237,843 instances of Vt-N structure as our dataset (called as ASBC).

4.2 Experiment 1: Evaluation of the Vt-N Classifier

In this experiment, we used the Maximum Entropy Toolkit (Zhang, 2004) to develop the Vt-N classifier. Based on the features discussed in Section 3.1, we designed five models to evaluate the classifier's performance on different feature combinations.

The features and used in each model are described below. The feature values shown in brackets refer to the example in Figure 1.

- **M1** is the baseline model. It uses PoS-tag pairs as features, such as (t_1=VC, t_2=Na).
- **M2** extends the M1 model by adding context features of (t_{-1}=VK, t_1=VC), (t_2=Na, t_3=DE), (t_{-2}=Nep, t_{-1}=VK, t_1=VC), (t_2=Na, t_3=DE, t_4=Na) and (t_{-1}=VK, t_3=DE).
- M3 extends the M2 model by adding lexicon features of (w_1=學習, t_1=VK, w_2=中文, t_2=Na), (w_1＝學習, w_2=中文), (w_1=學習) and (w_2=中文).
- M4 extends the M3 model by adding semantic features of (st_1=study|學習, t_1=VK , st_2=language|語言, t_2=Na), (st_1=study|學習, t_1=VK) and (st_2=language|語言, t_2=Na).
- M5 extends the M4 model by adding two features: the morph-structure of verbs; and the syllabic length of nouns (Vmorph='VV') and (Nlen=2).

Table 5 shows the results of using different feature combinations in the models. The symbol P1(%) is the 10-fold cross validation accuracy of the training data, and the symbol P2(%) is the accuracy of the test data. By adding contextual features, the accuracy rate of M2 increases from 59.10% to 72.30%. The result shows that contextual information is the most important feature used to disambiguate VP, NP and independent structures. The accuracy of M2 is approximately the same as the result of our PCFG parser because both systems use contextual information. By adding lexical features (M3), the accuracy rate increases from 72.30% to 80.20%. For semantic type features (M4), the accuracy rate increases from 80.20% to 81.90%. The 1.7% increase in the accuracy rate indicates that semantic generalization is useful. Finally, in M5, the accuracy rate increases from 81.90% to 83.00%. The improvement demonstrates the benefits of using the verb morph-structure and noun length features.

Models	Feature for Vt-N	P1(%)	P2(%)
M1	(t_1,t_2)	61.94	59.10
M2	+ (t_{-1},t_1) (t_2,t_3) (t_{-2},t_{-1},t_1) (t_2,t_3,t_4) (t_{-1},t_3)	76.59	72.30
M3	+ (w_1,t_1,w_2,t_2) (w_1,w_2) (w_2) (w_1)	83.55	80.20
M4	+ (st_1,t_1,st_2,t_2) (st_1,t_1) (st_2, t_2)	84.63	81.90
M5	+ (Vmorph) (Nlen)	85.01	83.00

Table 5. The results of using different feature combinations

Next, we consider the influence of unknown words on the Vt-N classifier. The statistics shows that 17% of the words in Treebank lack semantic type information, e.g., 留在/*StayIn*, 填飽/*fill*, 貼出/*posted*, and 綁好/*tied*. The accuracy of the Vt-N classifier declines by 0.7% without semantic type information for unknown words. In other words, lexical semantic information improves the accuracy of the Vt-N classifier. Regarding the problem of unknown morph-structure of words, w observe that over 85% of verbs with more than 2 characters are not found in the affix database. If we exclude unknown words, the accuracy of the Vt-N prediction decreases by 1%. Therefore, morph-structure information has a positive effect on the classifier.

4.3 Experiment 2: Using Knowledge Obtained from Large-scale Unlabeled Data by the Selection and Correction Methods.

In this experiment, we evaluated the two methods discussed in Section 3, i.e., unsupervised NP selection and supervised error correction. We applied the data selection method (i.e., *distance*=1, with an intransitive verb (Vi) followed by an object noun (Na)) to select 46,258 instances from the ASBC corpus and compile a dataset called Treebank+ASBC-Vi-N. Table 6 shows the performance of model 5 (M5) on the training data derived from Treebank and Treebank+ASBC-Vi-N. The results demonstrate that learning more nouns that accept verbal modifiers improves the accuracy.

	Treebank+ ASBC-Vi-N	Treebank
size of training instances	46,258	8,017
M5 - P2(%)	83.90	83.00

Table 6. Experiment results on the test data for various knowledge sources

We had also try to use the auto-parsed results of the Vt-N structures from the ASBC corpus as supplementary training data for train M5. It degrades the model's performance by too much error when using the supplementary training data. To resolve the problem, we utilize the supervised error correction method, which manually correct errors rapidly because high frequency instances (w_1, w_2) rarely have ambiguous classifications in different contexts. So we designed an editing tool to correct errors made by the parser in the classification of high frequency Vt-N word pairs. After the manual correction operation, which takes 40 man-hours, we assign the correct classifications (w_1, t_1, w_2, t_2, rt) for 2,674 Vt-N structure types which contains 10,263 instances to creates the ASBC+Correction dataset. Adding the corrected data to the original training data increases the precision rate to 88.40% and reduces the number of errors by approximately 31.76%, as shown in the Treebank+ASBC+Correction column of Table 7.

	Treebank+ ASBC+Correction	Treebank+ ASBC-Vi-N	Treebank
size of training instances	56,521	46,258	8,017
M5 - P2(%)	88.40	83.90	83.00

Table 7. Experiment results of classifiers with different training data

We also used the precision and recall rates to evaluate the performance of the models on each type of relation. The results are shown in Table 8. Overall, the Treebank+ASBC+Correction method achieves the best performance in terms of the precision rate. The results for Treebank+ASBC-Vi-N show that the unsupervised data selection method can find some knowledge to help identify NP structures. In addition, the proposed models achieve better precision rates than the PCFG parser. The results demonstrate that using our guidelines to design a disambiguation model to resolve the Vt-N problem is successful.

		H/X	X/H	X/X
Treebank	R(%)	91.11	67.90	74.62
	P(%)	84.43	78.57	81.86
Treebank+ ASBC-Vi-N	R(%)	91.00	72.22	71.54
	P(%)	84.57	72.67	85.71
Treebank+ ASBC+Correction	R(%)	98.62	60.49	83.08
	P(%)	86.63	88.29	93.51
PCFG	R(%)	90.54	23.63	80.21
	P(%)	78.24	73.58	75.00

Table 8. Performance comparison of different classification models.

4.4 Experiment 3: Integrating the Vt-N classifier with the PCFG Parser

Identifying Vt-N structures correctly facilitates statistical parsing, machine translation, infor-

mation retrieval, and text classification. In this experiment, we develope a baseline PCFG parser based on feature-based grammar representation by Hsieh et al. (2012) to find the best tree structures (*T*) of a given sentence (*S*). The parser then selects the best tree according to the evaluation score *Score(T,S)* of all possible trees. If there are *n* PCFG rules in the tree *T*, the *Score(T,S)* is the accumulation of the logarithmic probabilities of the *i*-th grammar rule (*RPi*). Formula 1 shows the baseline PCFG parser.

$$Score(T,S) = \sum_{i=1}^{n} (RP_i) \quad (1)$$

The Vt-N models can be easily integrated into the PCFG parser. Formula 2 represents the integrated structural evaluation model. We combine RPi and VtNPi with the weights w_1 and w_2 respectively, and set the value of w_2 higher than that of w_1. VtNPi is the probability produced by the Vt-N classifier for the type of the relation between Vt-N bigram determined by the PCFG parsing. The classifier is triggered when a [Vt, N] structure is encountered; otherwise, the Vt-N model is not processed.

$$Score(T,S) = \sum_{i=1}^{n} (w_1 \times RP_i + w_2 \times VtNP_i) \quad (2)$$

The results of evaluating the parsing model incorporated with the Vt-N classifier (see Formula 2) are shown in Table 9 and Table 10. The P2 is the accuracy of Vt-N classification on the test data. The bracketed *f*-score (BF^2) is the parsing performance metric. Based on these results, the integrated model outperforms the PCFG parser in terms of Vt-N classification. Because the Vt-N classifier only considers sentences that contain Vt-N structures, it does not affect the parsing accuracies of other sentences.

	PCFG + M5 (Treebank)	PCFG
P2(%)	80.68	77.09
BF(%)	83.64	82.80

Table 9. The performance of the PCFG parser with and without model M5 from Treebank.

[2] The evaluation formula is (BP*BR*2) / (BP+BR), where BP is the precision and BR is the recall.

	PCFG + M5 (Treebank+ASBC+Correction)	PCFG
P2(%)	87.88	77.09
BF(%)	84.68	82.80

Table 10. The performance of the PCFG parser with and without model M5 from Treebank+ASBC+Correction data set.

4.5 Experiment 4: Comparison of Various Chinese Parsers

In this experiment, we give some comparison results in various parser: 'PCFG Parser' (baseline), 'CDM Parser' (Hsieh et al., 2012), and 'Berkeley Parser' (Petrov et al., 2006). The CDM parser achieves the best score in Traditional Chinese Parsing task of SIGHAN Bake-offs 2012 (Tseng et al., 2012). Petrov's parser (as Berkeley, version is 2009 1.1) is the best PCFG parser for non-English language and it is an open source. In our comparison, we use the same training data for training models and parse the same test dataset based on the gold standard word segmentation and PoS tags. We have already discussed the PCFG parser in Section 4.4. As for CDM parser, we retrain relevant model in our experiments. And since Berkeley parser take different tree structure (Penn Treebank format), we transform the experimental data to Berkeley CoNLL format and re-train a new model with parameters "-treebank CHINESE -SMcycles 4" [3] from training data. Moreover we use "-useGoldPOS" parameters to parse test data and further transform them to Sinica Treebank style from the Berkeley parser's results. The different tree structure formats of Sinica Treebank and Penn Treebank are as follow:

```
Sinica Treebank:
S(NP(Head:Nh:他們)|Head:VC:散播
|NP(Head:Na:熱情))

Penn Treebank:
( (S (NP (Head:Nh (Nh 他們))) (Head:VC
(VC 散播)) (NP (Head:Na (Na 熱情)))))
```

The evaluation results on the testing data, i.e. in P2 metric, are as follows. The accuracy of PCFG parser is 77.09%; CDM parser reaches 78.45% of accuracy; and Berkeley parser is 70.68%. The results show that the problem of Vt-

[3] The "-treebank CHINESE -SMcycles 4" is the best training parameter in Traditional Chinese Parsing task of SIGHAN Bake-offs 2012.

N cannot be well solved by any general parser including CDM parser and Berkeley's parser. It is necessary to have a different approach aside from the general model. So we set the target for a better model for Vt-N classification which can be easily integrated into the existing parsing model. So far our best model achieved the P2 accuracy of 87.88%.

5 Concluding Remarks

We have proposed a classifier to resolve the ambiguity of Vt-N structures. The design of the classifier is based on three important guidelines, namely, adopting linguistically motivated features, using all available resources, and easy integration into parsing model. After analyzing the Vt-N structures, we identify linguistically motivated features, such as lexical words, semantic knowledge, the morphological structure of verbs, neighboring parts-of-speech, and the syllabic length of words. Then, we design a classifier to verify the usefulness of each feature. We also resolve the technical problems that affect the prediction of the semantic types and morph-structures of unknown words. In addition, we propose a framework for unsupervised data selection and supervised error correction for learning more useful knowledge. Our experiment results show that the proposed Vt-N classifier significantly outperforms the PCFG Chinese parser in terms of Vt-N structure identification. Moreover, integrating the Vt-N classifier with a parsing model improves the overall parsing performance without side effects.

In our future research, we will exploit the proposed framework to resolve other parsing difficulties in Chinese, e.g., N-N combination. We will also extend the Semantic Type Predication Algorithm (Figure 2) to deal with all Chinese words. Finally, for real world knowledge learning, we will continue to learn more useful knowledge by auto-parsing to improve the parsing performance.

Acknowledgments

We thank the anonymous reviewers for their valuable comments. This work was supported by National Science Council under Grant NSC99-2221-E-001-014-MY3.

Reference

Li-li Chang, Keh-Jiann Chen, and Chu-Ren Huang. 2000. Alternation Across Semantic Fields: A Study on Mandarin Verbs of Emotion. *Internal Journal of Computational Linguistics and Chinese Language Processing (IJCLCLP)*, 5(1):61-80.

Keh-Jiann Chen, Chu-Ren Huang, Chi-Ching Luo, Feng-Yi Chen, Ming-Chung Chang, Chao-Jan Chen, , and Zhao-Ming Gao. 2003. Sinica Treebank: Design Criteria, Representational Issues and Implementation. In *(Abeille 2003) Treebanks: Building and Using Parsed Corpora*, pages 231-248. Dordrecht, the Netherlands: Kluwer.

Li-jiang Chen. 2008. Autolabeling of VN Combination Based on Multi-classifier. *Journal of Computer Engineering*, 34(5):79-81.

Wenliang Chen, Daisuke Kawahara, Kiyotaka Uchimoto, Yujjie Zhang, and Hitoshi Isahara. 2008. Dependency Parsig with Short Dependency Relations in Unlabeled Data. In *Proceedings of the third International Joint Conference on Natural Language Processing (IJCNLP)*. pages 88-94..

Chih-ming Chiu, Ji-Chin Lo, and Keh-Jiann Chen. 2004. Compositional Semantics of Mandarin Affix Verbs. In *Proceedings of the Research on Computational Linguistics Conference (ROCLING)*, pages 131-139.

Yu-Ming Hsieh, Ming-Hong Bai, Jason S. Chang, and Keh-Jiann Chen. 2012. Improving PCFG Chinese Parsing with Context-Dependent Probability Re-estimation, In *Proceedings of the Second CIPS-SIGHAN Joint Conference on Chinese Language Processing*, pages 216–221.

Shu-Ling Huang, You-Shan Chung, Keh-Jiann Chen. 2008. E-HowNet: the Expansion of HowNet. In *Proceedings of the First National HowNet workshop*, pages 10-22, Beijing, China.

Chunhi Liu, *Xiandai Hanyu Shuxing Fanchou Yianjiu (現代漢語屬性範疇研究)*. Chengdu: Bashu Books, 2008.

Jiaju Mei, Yiming Lan, Yunqi Gao, and Yongxian Ying. 1983. *A Dictionary of Synonyms*. Shanghai Cishu Chubanshe.

Slav Petrov, Leon Barrett, Romain Thibaux and Dan Klein. 2006. Learning Accurate, Compact, and Interpretable Tree Annotation. In *Proceesings of COLING/ACL*, pages 433-400.

Likun Qiu. 2005. Constitutive Relation Analysis for V-N Phrases. *Journal of Chinese Language and Computing*, 15(3):173-183.

Richard Sproat and Thomas Emerson, 2003. The first International Chinese Word Segmentation Bakeoff. In *Proceedings of the Second SIGHAN Workshop on Chinese Language Processing*, pages 133-143.

Honglin Sun and Dan Jurafsky. 2003. The Effect of Rhythm on Structural Disambiguation in Chinese. In *Proceedings of the Second SIGHAN Workshop on Chinese Language Processing*, pages 39-46.

Yuen-Hsieh Tseng, Lung-Hao Lee, and Liang-Chih Yu. 2012. Tranditional Chinese Parsing Evaluation at SIGHAN Bake-offs 2012. In *Proceedings of the Second CIPS-SIGHAN Joint Conference on Chinese Language Processing*, pages 199-205.

Andi Wu. 2003. Learning Verb-Noun Relations to Improve Parsing. In *Proceedings of the Second SIGHAN workshop on Chinese Language Processing*, pages 119-124.

Kun Yu, Daisuke Kawahara, and Sadao Kurohashi. 2008. Chinese Dependency Parsing with Large Scale Automatically Constructed Case Structures, In *Proceedings of the 22nd International Conference on Computational Linguistics (COLING2008)*, pages 1049-1056.

Jun Zhao and Chang-ning Huang. 1999. The Complex-feature-based Model for Acquisition of VN-construction Structure Templates. *Journal of Software*, 10(1):92-99.

Le Zhang. 2004. Maximum Entropy Modeling Toolkit for Python and C++. Reference Manual.