Extending Bilingual WordNet via Hierarchical Word Translation Classification

Tzu-yi Nien[a], Tsun Ku[b], Chung-chi Huang[a], Mei-hua Chen[a], and Jason S. Chang[a]

[a] Institute of Information Systems and Applications, National
Tsing Hua University, HsinChu, Taiwan 300, R.O.C.
{zinien, u901571, chen.meihua, jason.jschang}@gmail.com
[b] Institute for Information Industry, 11F, No. 106, Section 2,
Heping East Road, Taipei, Taiwan 106, R.O.C.
cujing@iii.org.tw

Abstract. We introduce a method for learning to assign word senses to translation pairs. In our approach, this sense assignment or disambiguation problem is transformed into one on how to navigate through a sense network like WordNet aimed at distinguishing the more adequate senses from others. The method involves automatically constructing classification models for branching nodes in the network, and automatically learning to reject less probable senses, based on the translation characteristics of word senses and semantically-related word groups (e.g., lexicographer files) respectively. At run-time, translation pairs are expanded with their synonyms and sense ambiguity is resolved using a greedy algorithm choosing the most likely branches based on the trained classification models. Evaluation shows that our method significantly outperforms the strong baseline of assigning most frequent sense to the translation pairs and effectively determines suitable word senses for given translation pairs, suggesting the possibility of employing our method as a computer-assisted tool for speeding up the process of lexicography or of using our method to assist machine translation systems in word selection.

Keywords: Word sense disambiguation, word translation classification, WordNet, machine-learning technique, maximum entropy model.

1 Introduction

Many words (e.g., *plant*) have different senses in different contexts (e.g., *green plant* and *nuclear plant*), usually leading to different translations in another language (e.g., 植物 and 工廠 respectively). On the other hand, different words (e.g., *plant* and *factory*) may express very similar meanings (e.g., *the working place for industrial labors*). Therefore, WordNet (Miller *et al*., 1990), a sense inventory encoding with semantic relatedness of words, has been a valuable resource in the field of natural language processing since its introduction. In WordNet, nominal, verbal, adjective, and adverbial words are grouped into synonym sets, or so-called synsets and synsets are interlinked with various semantic relations (e.g., *hypernym*, *hyponym* and etc). Its rich and well-defined lexical semantic relations have made WordNet an important knowledge source for various research areas: word sense disambiguation; computer-assisted language learning; information retrieval.

The well-established lexical hierarchy residing in English WordNet has prompted researchers to construct WordNet-like sense inventory for other languages. Take the Chinese language for example. Efforts have been made on automatic construction and on manual translation from English WordNet into Chinese. Chinese translations in the latter case, however, may not sufficiently cover the scope. It would be more efficient and cost-effective if translations of various word senses could be automatically integrated from bilingual sources (e.g., dictionaries and phrase tables in machine translation systems).

Consider the word "plant" with the sense of "*buildings for carrying on industrial labor*" and the sense of "*a living organism lacking the power of locomotion*", and 廠房 (manufactory), one of its Chinese translations. Assume that "廠房" is unseen in a bilingual WordNet (e.g., English WordNet with Chinese translations). The best way to incorporate such a new translation is probably not blindly assigning it to all the senses of "plant". A good way might be to identify the most appropriate sense for the translation, in this case, "plant#1" (i.e., *buildings for carrying on industrial labor*). Intuitively, by leveraging sense-to-translation relations, such sense ambiguity could be resolved.

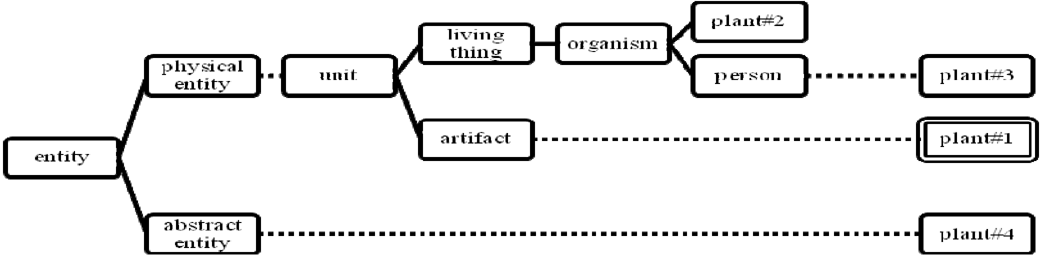

Figure 1: An example classifying diagram for (plant, 廠房). The correct sense for this translation pair is "plant#1". Note that our classification models are applied on three branching[1] synsets (i.e., "entity", "unit", and "organism") and that the WordNet hierarchy shown here is simplified: non-branching synsets are hidden and represented by dashed lines.

We present a hierarchical word translation classification (WTC) model that automatically learns to attach translations of English words to the adequate word senses. An example classification diagram for (plant, 廠房) is shown in Figure 1. Paths from the root, "entity", to the four nominal senses of "plant" are highlighted and our goal is to find the suitable sense for the <word, translation> pair. Our model learns to navigate through the lexical hierarchy in WordNet (to determine the sense for the given translation) during training by analyzing a collection of translation pairs in bilingual WordNet. We describe the training process of hierarchical WTC Model in more detail in Section 3.

At run-time, our model starts with a <word, translation> pair (e.g., (plant, 廠房)) from a bilingual knowledge resource and then transforms the disambiguation problem to a hierarchical classification problem. In our prototype, features extracted from the translation are exploited to find its adequate sense. Additional translations of word senses provided by our model can be used to broaden the scope of an existing bilingual WordNet. Alternatively, our model can be embedded into machine translation (MT) systems in order to help choose more appropriate word translations.

2 Related Work

Word Sense Disambiguation (WSD) has been an area of active research. WSD is to determine the meaning of a word in current context, which is an important component in language understanding or MT systems.

WSD models have been developed using machine learning techniques. They may train on sets of sense-annotated data for predefined words (Hearst, 1991; Leacock *et al*., 1993; Bruce and Wiebe, 1994). To avoid the labor-intensive and time-consuming process of sense-tagging, Yarowsky (1995) propose a semi-supervised model to bootstrap from raw data based on some confident and unambiguous seeds.

[1] With respect to the sense disambiguation task.

Another direction is to base WSD models on dictionaries or lexical semantic knowledge resources. Lesk (1986) is the first to leverage the definitions of words in machine readable dictionaries to predict word senses. On the other hand, WordNet, a valuable knowledge source encoded with hyponym, hypernym, and synonym semantic relations, is used to measure semantic distances among word senses to help sense disambiguation (Agirre and Rigau, 1996; Galley and McKeown, 2003). An interesting approach presented by Mihalcea (2005) describes how to apply a graph-based algorithm (i.e., random walk algorithm) and WordNet semantic relations to solve all-word WSD task.

Recently, WSD not only has been approached from bilingual perspective, but has been applied to bilingual applications. Li and Li (2002) introduce "bilingual bootstrapping" making use of a small number of sense-annotated data to further bootstrap two languages' discerning or effective context words in disambiguation. Gale *et al.* (1992) and Diab and Resnik (2002) also leverage bilingual information in WSD. WSD or word translation disambiguation (WTD), aimed at improving word selection in MT, has been proved to have positive influence on bilingual application like statistical MT systems (Chan *et al.*, 2007; Carpuat and Wu, 2007).

In our work, word senses are assigned to given translations, which is the opposite of WTD, choosing translations for senses, in view of extending the translation coverage of an existing bilingual WordNet such as Sinica Bilingual Ontological WordNet (Huang *et al.*, 2004), Sinica BOW for short. Such bilingual WordNet may be constructed manually by translation (Huang *et al.*, 2004) or automatically (Chang *et al.*, 2003). Our work can be thought of as (Chang *et al.*, 2003)'s follow-up research which enriches the translations in bilingual WordNet.

3 Hierarchical Word Translation Classification

3.1 Problem Statement

We focus on the essential step of extending bilingual WordNet: determining the appropriate word senses for unseen translation pairs from bilingual knowledge resources (e.g., dictionaries or phrase tables). Using bilingual WordNet which provides a hierarchical structure on tree nodes (i.e., synsets) and translations, we train a classifier at each branching node that estimates associations between given translations and branching node's children (i.e., inherited hyponyms). Then, the problem of sense disambiguation is transformed to a hierarchical classification problem. We now formally state the problem we are addressing.

Problem Statement: We are given a bilingual WordNet (e.g., Sinica BOW) and a word-translation pair ($\boldsymbol{e}$, $\boldsymbol{f}$). Our goal is to assign the most adequate and relevant sense $\boldsymbol{s}_\mathrm{i}$ to $\boldsymbol{f}$ where $\boldsymbol{s}_\mathrm{i} \in \boldsymbol{S} = \{\boldsymbol{s}_1, ..., \boldsymbol{s}_\mathrm{n}\}$, a set of word senses $\boldsymbol{e}$ has. For this, we traverse the WordNet from top abstract synsets to the bottom word senses (i.e., $\boldsymbol{s}_1$, …, $\boldsymbol{s}_\mathrm{n}$) and identify all related branching nodes such that the probabilities of the branching paths associated with the translation $\boldsymbol{f}$ can be estimated and the most likely sense, $\boldsymbol{s}_\mathrm{i}$, can therefore be pinpointed.

3.2 Learning to Classify Translations

(1) Propagate Translations to Generate the Training Data
(2) Train Hierarchical Word Translation Classification Models
(3) Train Filtering Model

Figure 2: Outline of the training process.

We attempt to resolve the sense ambiguity by learning lexical characteristics from a collection of translation pairs in a bilingual WordNet. Our learning process is shown in Figure 2.

Propagating Translations. In the first stage of the learning process (Step (1) in Figure 2), we propagate translations of each word sense (i.e., synsets) to its inherited hypernyms (i.e., ancestors) in WordNet. Then, the word translation classification (WTC) models described in the following stage (Step (2) in Figure 2) can exploit this information to learn to classify translations into appropriate senses. Here, propagating means incorporating translations of synsets into translation lists (TL's) of their hypernyms. The rationale behind propagating translations to their hypernyms is to establish additional associations between translations and hypernyms. For instance, higher-level concept "artifact" (a man-made object taken as a whole) will be related to some common translation features (e.g., unigram character of "廠", "房" and "器") shared among the translations of its hyponyms, after translation propagation.

The input to this stage is a bilingual WordNet, a collection of <word, translation> pairs with word senses. These pairs constitute our training data along with lexical hierarchy (i.e., hypernym/hyponym relations). We also take into account the frequency, i.e. *tag_count*, of word senses provided in WordNet. Higher *tag_count* value implies more frequently occurring sense. The output of this stage is a collection of TL's associated with WordNet synsets.

```
procedure PropagateTranslation(e, Sense, f)
(1) Synset = GetSynset(e, Sense)
(2) Cnt = GetTagCount(e, Sense)
(3) Hypernyms = GetHypernyms(Synset)
(4) AddToList(Synset, f, Cnt)
    for each hi in Hypernyms
(5)   AddToList(hi, f, Cnt)
```

Figure 3: Algorithm of translation propagation.

Figure 3 shows the algorithm for propagating a translation in the WordNet hierarchy. This procedure applies to each translation pair in the bilingual WordNet.

In Step (1) of the algorithm we identify the synset of the English word *e* and its word sense *Sense* in that synsets are the basic units for any available semantic relations in WordNet. Then, we look up the frequency count, *Cnt*, of *e* and *Sense* (Step (2)). In Step (3) we identify the hypernyms of the synset *Synset* for translation propagation. The hypernyms, *Hypernyms*, express more abstract or more general concepts than *Synset* does.

Finally, we integrate the translation *f* into TL's of *Synset* and *Hypernyms* (Step (4) and (5)). Note that we also populate sense frequency (*Cnt*) to TL's such that WTC models described in following stage can leverage the frequency information.

Training Hierarchical Word Translation Classification Models. In the second stage of the learning algorithm (Step (2) in Figure 2), we train translation classification models for *branching* synsets with more than one direct hyponym in WordNet. To navigate from the top, general concepts, to the bottom, specific word senses, and to find the right class for a translation in WordNet hierarchy, we utilize machine learning technique to construct hierarchical word translation classification models. See Figure 1 for the example of branching synsets.

The input of this stage is the propagated translation data obtained from the previous stage, a collection of <*WordNet synset*, *TranslationList*> pairs. The output of this stage is a set of WTC models which estimate associations between given translation and one of the direct hyponyms of the branching synset in question.

In this paper, we employ Maximum Entropy (ME) as our machine learning model. As a statistical model, ME offers a neat way to incorporate any potential features for outcome prediction.

During the training of our ME-based WTC model, all direct hyponyms of a synset constitute the outcome space of the classification model, and features, as we will describe in detail later, are derived from instances in TL's. Specifically, the association between a direct hyponym (an *outcome*) of a synset and the translation *f* is governed by the conditional probability as

$$p\left(outcome \middle| f\right) = \frac{\exp\left(\sum_i \lambda_i \times feature_i\left(outcome, f\right)\right)}{\sum_{o \in outcomes} \exp\left(\sum_i \lambda_i \times feature_i\left(o, f\right)\right)} \qquad (1)$$

where *outcomes* is a set of all direct hyponyms of the synset, *feature*$_i$ is a binary-valued function, and λ_i is the weight of the feature function *feature*$_i$. Note that λ_i's are tuned to reflect the significance of the features in determining the hyponym-translation association and that, during training, sense frequency (*tag_count*) is used to indicate the importance of the translation being associated with the word sense. In our implementation, we perform add-one smoothing technique to deal with zero *tag_count*.

Now, we describe the features (i.e., *feature*$_i$) used in our model. Inspired by the observation that translations of a semantic synset (e.g., "artifact#1") are likely to share some common words or characters (e.g., "器", "廠" and "房"), n-gram features, referred to as literal features, of translations are leveraged. Following describes three types of literal features for Chinese translations:

- **Unigram Feature**: Chinese characters tend to carry some sort of semantic meanings. Therefore, we split Chinese translations into characters and collect their corresponding features. For instance, "核", "能", "發", "電", and "廠" are the literal unigram features of the translation "核能發電廠" (nuclear plant).
- **Bigram Feature**: Since consecutive two Chinese characters, which we refer to as bigrams, might convey more specific meaning than unigrams, bigrams are also used as features. For the above instance, there are four bigram features, "核能", "能發", "發電", and "電廠".
- **Head Word Feature**: The head word of a Chinese translation may occur at the beginning or in the end and the length of a head is uncertain. As a result, both ends of the translation and a preset character limit on head word are used to generate our head word features. For instance, "核", "廠", "核能", and "電廠" are selected as the head word features of "核能發電廠" if character limit is set to two.

Notice that although alternative machine learning approaches can be exploited to train the WTC models, using ME has a number of advantages. Firstly, ME provides an easy way to incorporate potential feature functions so that research efforts can be focused on selecting representative features to characterize the problems. In addition, features in ME models are assigned with highly-tuned weights and ME models are trained without the assumption of feature independence, one of the issues facing Naïve Bayesian model.

Training Filtering Model. In the third and final stage of the learning process, we train *a* filtering model at the so-called lexicographer file level of WordNet to prune unlikely starting synsets, leading to the word senses, for the given translation. More specifically, instead of dealing with very general and abstract concepts at the top level, the model classifies the given translation to some more specific and concrete semantic categories. This filtering aims to accelerate the process of word translation classification and to boost the performance by reducing the probability that the hierarchical WTC models set out on the wrong foot.

In principle, hierarchical WTC models described in the previous section alone could resolve the sense ambiguity if implemented with a greedy path-finding algorithm. Nonetheless, it is likely that during the classification process, WTC models fail to make the correct branch prediction for the *first* few branching synsets at the higher level of WordNet hierarchy because their immediate hyponyms convey too general concepts.

The <*WordNet synset*, *TranslationList*> pairs from Step (1) in Figure 2 are utilized to train the filtering model. The filtering model, a ME-based classification model, estimates associations between features of a given translation and some predefined outcome, in this case, the lexicographer files. Lexicographer files are semantic categories organized during the development of WordNet. In total, there are forty-five lexicographer files: twenty-six for nouns, fifteen for verbs, three for adjectives, and one for adverbs. As for features, we use the same feature sets previously described.

While we construct a WTC model for each branching synset, we only train a single filtering model at the level of the lexicographer files in WordNet to filter out synsets whose associations with the given translations are smaller than θ (a threshold to be determined). Moreover, we also use the smoothed sense frequency to reflect the importance of the translations of the frequent sense.

3.3 Run-Time Translation Classification

Once the WTC and filtering models are constructed, we are ready to classify translations to corresponding word senses in WordNet. We associate adequate senses with given translations using the procedure in Figure 4.

```
procedure ClassifyTranslation(e, f)
(1)  Senses = GetSenses(e)
(2)  TranslationSynonyms = GetSynonyms(f, N) ∪ {f}
//Remove unlikely senses via filtering model
     for each sense s in Senses
(3)     c = GetCategory(s)
(4a)    score = 0
        for each t in TranslationSynonyms
(4b)       score += FilteringModel(c, GetFeatures(t))
        if score/#TranslationSynonyms < θ
(4c)       Remove sense s from Senses
//Greedily select the most probable branch at each branching node via hierarchical WTC models
     while #Senses > 1
(5)     BS = GetBranchingSynset(Senses)
(6)     WTCModel = Load WTC model associated with BS
        for each h_i in hyponyms of BS
(7a)       score_i = 0
           for each t in TranslationSynonyms
(7b)          score_i += WTCModel(h_i, GetFeatures(t))
(7c)    ChosenBranch = h_i with max score
        for each sense s in Senses
           if ChosenBranch is not an inherited hypernym of s
(8)           Remove sense s from Senses
(9) Return the only sense in Senses
```

Figure 4: Run-time classification algorithm.

In Step (1) we retrieve the senses of the given English word *e* in WordNet as the candidate senses for disambiguation. Then, we expand translation *f* with its synonyms (at most *N* synonyms) by looking up a synonym thesaurus (Step (2)). The motivation of synonym expansion is to reduce the impact of rare translations (e.g., “寒玉” a translation of “moon”) on system performance. The features of the more frequently used translations (e.g., “月亮” and “月” for “moon”) usually are more effective and useful in classification because of their commonness and lexical characteristics. All of the translation synonyms will be considered in filtering (Step (4b)) and branch prediction (Step (7b)).

In Step (3) and (4), we prune less likely senses via filtering model using the lexicographer files, or semantic categories, associated with them. The filtering model predicts the relatedness between features of the given translation (as well as each of its synonyms) and the semantic categories (Step (4b)). Since the given translation and its synonyms basically express similar concept, their predicted scores are weighted equally. The senses with *averaged* score less than a threshold θ are removed from the sense set in Step (4c).

The remaining sense ambiguity is resolved using hierarchical WTC models (from Step (5) to (8)). In Step (5) branching synset, *BS* , whose immediate hyponyms each cover a subset of the remained candidate senses, is identified by examining the network of WordNet. As the algorithm proceeds, *BS*’s move downwards in the WordNet hierarchy as the ambiguity at upper levels is being resolved. The WTC model associated with *BS* is loaded in Step (6) to estimate the hyponym-translation association and predict the most likely branch *ChosenBranch*, satisfying $\arg\max_{h_i}\{score_i\}$ for *BS* (Step (7b) and (7c)). In Step (8), we discard the sense whose inherited hypernyms do not include *ChosenBranch* in a greedy manner. The algorithm continues until only one sense remains, then assigned to the given translation pair as the most relevant sense (Step (9)).

4 Experimental Setting

4.1 Data Sets

We used the latest version of WordNet (i.e., WordNet 3.0) as our lexical hierarchy and trained our classification models on Sinica BOW, a English-Chinese WordNet. In our experiments, we focused on nominal synsets and hyponym/hypernym semantic relation defined in WordNet. On the other hand, we looked up 同義詞詞林, a Chinese thesaurus for run-time synonym expansion.

We randomly selected 500 nouns from SEMCOR, a subset of Brown Corpus, and manually translated them into Chinese via Longman English-Chinese Dictionary of Contemporary English. After removing the <word, translation> pairs already existing in the bilingual WordNet, 300 translation pairs were randomly selected as our evaluation data, and 100 of them made up of our development data set for tuning system parameters and the rest our (outside) testing data.

4.2 Models Compared

In view of extending the existing bilingual WordNet, we propose a classification framework for categorizing the given translations from bilingual knowledge resources into suitable word senses, in which we deploy a filtering model (**FM**) and hierarchical WTC models (**HM**) obtained using the learning process in Section 3. In addition, translation synonym expansion (**TS**) is applied to reduce the impact of rare translations on system performance. To inspect the effectiveness of these modules, a baseline model and the models using our three main modules, **HM**, **FM**, and **TS**, are evaluated. Models compared are described as follows:

- **Baseline**: For any given translation pair, the most frequent sense is returned.
- **HM**: The translation is classified using only hierarchical WTC models. That is, the filtering threshold θ and the number of allowed translation synonyms N are both set to zero.
- **HM+FM**: Unlikely word senses are pruned by **FM** prior to **HM**. θ is set according to the tuning process described in Section 4.4 and N is set to zero.
- **HM+TS**: N additional translation synonyms are used in **HM**. No prior sense filtering is applied (θ is set to zero).
- **HM+FM+TS**: The complete version of the proposed system, using all three components.

4.3 Evaluation Metrics

In this subsection, we introduce the metrics, *Hit Rate* and *Mean Reciprocal Rank* (*MRR*), for evaluating the performance of our system.

Definition: The *Top-n Hit Rate* of a system S for a set of query translation pairs Q is the percentage of the pairs for which S returned at least one accurate sense (*hit*) among the top n returned senses.

Example: Consider an example where, among the 10 sets of the returned senses (i.e., 10 query translation pairs), 6 top-ranked and 2 second-place senses are confirmed accurate. The *Top-2 Hit Rate* of this system is then (6+2)/10 = 80%.

Besides, to measure the effort needed for a user to locate a correct sense in the returned sense lists, systems are evaluated using *MRR*. *MRR* is a real number lying between 0 and 1, in which 1 denotes the accurate senses always occur at the first places. We report the *MRR* results to examine the possibility of our system being used to help lexicographers bridge new translations to word senses.

Definition: The *Reciprocal Rank* of a system, for a translation pair p is defined as R_p^{-1}, where R_p is the smallest rank of the correct sense assigned to p. The *Mean Reciprocal Rank* of the system is the average of the *Reciprocal Rank* values over all evaluated translation pairs.

4.4 Tuning Parameters

We carried out pilot experiments on the development data set to tune the two parameters in our system: the filtering threshold, θ, and the number of allowed translation synonyms, N.

The filtering threshold in our model influences the degree of pruning. To select a suitable θ, the performance, in this case, the accuracy of senses being rejected by the filtering model (P), the coverage of our rejected senses (R), and the combination of the two (F-measure[2]), of our filtering model was evaluated at different thresholds.

Figure 5 summarizes the results. Based on the statistics in Figure 5, we set our filtering threshold to 0.04, at which the filtering model achieved highest F-measure, that is, most balanced performance between P (0.86) and R (0.59).

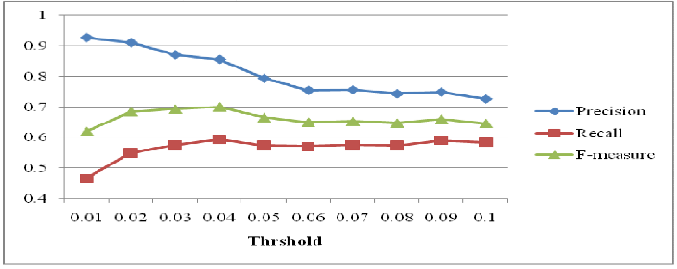

Figure 5: Precision (P), recall (R), and F-measure at different filtering thresholds.

To select an appropriate number for synonym expansion, we examined *MRR* of our model with respect to the number of translation synonyms expanded. Figure 6 shows that our model performed the best when at most two translation synonyms are allowed and that more synonyms did not lead to better results probably due to the noise introduced. In sum, we set the filtering threshold, θ, to 0.04, and the maximal number of allowed translation synonyms, N, to 2 in our experiments.

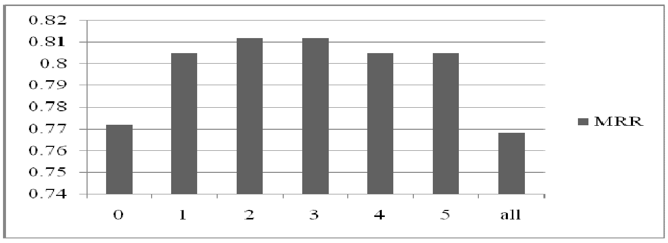

Figure 6: *MRR* of different *N*'s on developing data set.

5 Evaluation Results and Discussion

5.1 Experimental Results

In experiments, 200 testing translation pairs were classified using the models described in Section 4.2. Table 1 summarizes the performance of different combinations of the three main system modules (i.e., **HM**, **FM**, and **TS**).

Table 1: The evaluation results of different systems.

System	*Top-1 Hit Rate (%)*	*MRR*
Baseline	65	0.79
HM	74	0.84
HM+FM	75	0.83
HM+TS	75	0.84
HM+FM+TS	77	0.84

[2] 2*P*R/(P+R)

As suggested in Table 1, our proposed systems significantly outperformed the baseline in terms of *Top-1 Hit Rate*, which indicates that our classification strategy effectively and correctly assigned suitable word senses to given translation pairs. Among the four combinations of our system components, **HM+FM+TS**, a system with hierarchical WTC models, a filtering model, and synonym expansion, achieved the highest *Top-1 Hit Rate*, 77%, suggesting the WTC models benefited from sense pre-pruning and synonym expanding. On the other hand, the high *MRR* (0.84) pointed out that users (e.g., lexicographers) could often find the suitable word sense for the translation by looking at the first two senses in the ranked sense list generated by our model.

We further examine the *Top-1 Hit Rates* for words with different numbers of senses (See Figure 7). As we can see, *Top-1 Hit Rate* declines against the number of sense per word, and our model outperformed the baseline at all sense counts and remained at 70% accuracy. Also, excluding the 30 monosemous words in the test set enlarged the difference between our system and the baseline (72% vs. 58%).

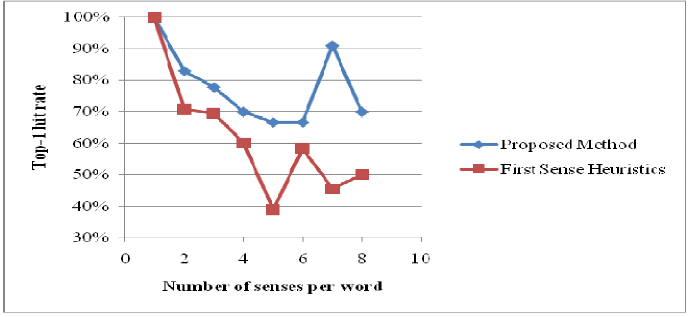

Figure 7: *Top-1 Hit Rates* of words with different numbers of word senses.

5.2 Error Analysis

In the experiment of **HM+FM+TS**, 47 translation pairs out of 200 were wrongly classified. And errors can be mainly grouped into three types: one related to high word sense ambiguity, one descriptive translations, and one transliterations.

Over 50% of the mislabeled translation pairs have more than 4 English word senses (the average number of senses per word was 4.4 in our test set), indicating that it is more difficult to assign correct senses to translation pairs with high degree of sense ambiguity. Nonetheless, our system still achieved much higher *Top-1 Hit Rate* (64%) in classifying the 73 translation pairs with more than 4 senses than the baseline (45%).

Another major type of errors results from descriptive translations, referring to the cases where words are not translated but, to some extent, defined in another language. Take "factory" for example. Its common Chinese translation is "工廠". However, "從事工業生產的場所" (a place for manufacturing) may be another, a descriptive one actually. Tokens of descriptive translations are likely to introduce noise (e.g., "的" in "從事工業生產的場所"), subsequently degrading the performance of our classification model. These descriptive translations might be correctly sense-labeled if more concise expression or translation is provided. For example, (recovery, "恢復健康") originally mislabeled will be correctly assigned to the sense "gradual healing (through rest) after sickness or injury" if provided with "康復", parts of "恢復健康".

Other errors are related to the fact that the given translations are transliterations. Our classification model aims to build relations between WordNet senses and *translations*, not *transliterations*. Transliterations usually reflecting the sound not the meanings of the words therefore hinder the model from functioning properly and accurately. An example transliteration in our test data is (trust, 托拉斯), and its adequate sense is "a consortium of independent organizations formed to limit competition by controlling the production and distribution of a product or service".

6 Future Work and Summary

Many avenues exist for future research and improvement of our system. For example, other potential features can be integrated into the classification framework, such as the translations of the glosses or the definitions of the word senses. Also, a simple procedure, which extracts content words or essential terms, our classifiers better at, from the explanatory or descriptive translations, can be employed prior to sense disambiguation or branch finding. Another interesting direction to explore is to further consider the context information of the given translations. For instance, the contexts of “植物” (green “plant”), e.g. “植物標本” and “有機植物”, and “工廠” (manufacturing “plant”), e.g. “模型工廠” and “機械工廠”, are very different and they may be informative for sense determination.

In summary, we have introduced a method for classifying a <word, translation> pair into an appropriate word sense in WordNet. Our goal is to automatically extend the scope of an existing bilingual WordNet by incorporating new translation pairs probably from dictionaries or parallel corpora. The method involves sense pre-filtering, hierarchical classification using ME-based models, and translation synonym expansion. We have implemented and thoroughly evaluated the method as applied to word sense assignment. In our evaluation, we have shown that the method outperforms the baseline in terms of *Top-1 Hit Rate* and *MRR*, an indicator of a system’s potential in accelerating the process of lexicography.

References

Agirre, E. and G. Rigau. 1996. Word Sense Disambiguation using Conceptual Density. *Conference on Computational Linguistics*, pp. 16-22.

Bruce, R. and J. Wiebe. 1994. Word-Sense Disambiguation Using Decomposable Models. *ACL*, pp. 139-146.

Carpaut, M. and D. Wu. 2007. Improving Statistical Machine Translation using Word Sense Disambiguation. *EMNLP*, pp. 61-72.

Chan, Y.-S., H.-T. Ng, and D. Chiang. 2007. Word Sense Disambiguation Improves Statistical Machine Translation. *ACL*, pp. 33-40.

Chang, J. S., T. Lin, G.-N. You, T. C. Chuang, and C.-T. Hsieh. 2003. Building a Chinese WordNet via Class-based Translation Model. *Journal of CLCLP*, pp. 61-76.

Diab, M. and P. Resnik. 2002. An Unsupervised Method for Word Sense Tagging using Parallel Corpora. *ACL*, pp. 255-262.

Gale, W. A., K. W. Church, and D. Yarowsky. 1992. Using Bilingual Materials to Develop Word Sense Disambiguation Methods. *Conference on Theoretical and Methodological Issues in Machine Translation*, pp. 101-112.

Galley, M. and K. McKeown. 2003. Improving Word Sense Disambiguation in Lexical Chaining. *Conference on Artificial Intelligence*.

Hearst, M. A. 1991. Noun Homograph Disambiguation using Local Context in Large Corpora. *Conference of the University of Waterloo Centre for the New OED and Text Research*, pp. 1-15.

Leacock, C., G. Towell, and E. Voorhees. 1993. Corpus-based Statistical Sense Resolution. *ARPA Human Language Technology Workshop*, pp. 260-265.

Lesk, M. 1986. Automatic Sense Disambiguation using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone. *Conference on Systems Documentation* pp. 24-26.

Li C. and H. Li. 2002. Word Translation Disambiguation Using Bilingual Boostrapping. *ACL*.

Miller, G. A., R. Beckwith, C. Fellbaum, D. Gross, and K. J. Miller. 1990. Introduction to WordNet: An On-line Lexical Database. *International Journal of Lexicography*, pp. 235-244.

Mihalcea, R. 2005. Unsupervised Large-Vocabulary Word Sense Disambiguation with Graph-based Algorithms for Sequence Data Labeling. *Conference on EMNLP*.

Yarowsky, D. 1995. Unsupervised Word Sense Disambiguation Rivaling Supervised Methods. *Annual Meeting of the ACL*.