Computational Linguistics and Chinese Language Processing
Vol. 10, No. 3, September 2005, pp. 303-328

Using Lexical Constraints to Enhance the Quality of Computer-Generated Multiple-Choice Cloze Items

Chao-Lin Liu*, Chun-Hung Wang* and Zhao-Ming Gao+

Abstract

Multiple-choice cloze items constitute a prominent tool for assessing students' competency in using the vocabulary of a language correctly. Without a proper estimation of students' competency in using vocabulary, it will be hard for a computer-assisted language learning system to provide course material tailored to each individual student's needs. Computer-assisted item generation allows the creation of large-scale item pools and further supports Web-based learning and assessment. With the abundant text resources available on the Web, one can create cloze items that cover a wide range of topics, thereby achieving usability, diversity and security of the item pool. One can apply keyword-based techniques like concordancing that extract sentences from the Web, and retain those sentences that contain the desired keyword to produce cloze items. However, such techniques fail to consider the fact that many words in natural languages are polysemous so that the recommended sentences typically include a non-negligible number of irrelevant sentences. In addition, a substantial amount of labor is required to look for those sentences in which the word to be tested really carries the sense of interest. We propose a novel word sense disambiguation-based method for locating sentences in which designated words carry specific senses, and apply generalized collocation-based methods to select distractors that are needed for multiple-choice cloze items. Experimental results indicated that our system was able to produce a usable cloze item for every 1.6 items it returned.

Keywords: Computer-assisted language learning, Computer-assisted item generation, Advanced authoring systems, Natural language processing, Word sense disambiguation, Collocations, Selectional preferences

* Department of Computer Science, National Chengchi University, Taipei 11605, Taiwan
E-mail: chaolin@nccu.edu.tw （劉昭麟及王俊弘，臺北市文山區，國立政治大學資訊科學系）
+ Department of Foreign Languages and Literatures, National Taiwan University, Taipei 10617, Taiwan
E-mail: zmgao@ntu.edu.tw （高照明，臺北市大安區，國立臺灣大學外國語文學系）

[Received February 3, 2005; Revised June 15, 2005; Accepted June 17, 2005]

1. Introduction

Due to the advent of modern computers and the Web, academic research on intelligent tutoring systems (ITSs) have grown in the last decade. Figure 1 shows a possible functional structure of the main components of an ITS that uses test items to assess students' competence levels. With the development of mature techniques for intelligent systems and the abundant information now available on the Internet, a computer-assisted *Authoring Component* that can help course designers construct large databases of high-quality test items and course materials has become possible [Irvine and Kyllonen 2002; Wang *et al.* 2003]. With *Test-Item* and *Course-Material Databases*, the *Tutoring Component* must find ways to provide materials appropriate for students. In the ideal case, we should be able to determine students' competence levels effectively and efficiently by means of various forms of assessment and provide course materials that are tailored to each individual student's particular needs [van der Linden and Hambleton 1997; van der Linden and Glas 2000; Liu 2005]. For this purpose, we need to have appropriate techniques and a *Student-Model Database* that together enable the *Adaptive Tester* and *Course Sequencer* to identify students' competence levels, predict their needs, and provide useful course materials. When the tutoring component cannot meet students' needs, the students should be able to feedback their requests or complaints to the course designers to facilitate future improvements.

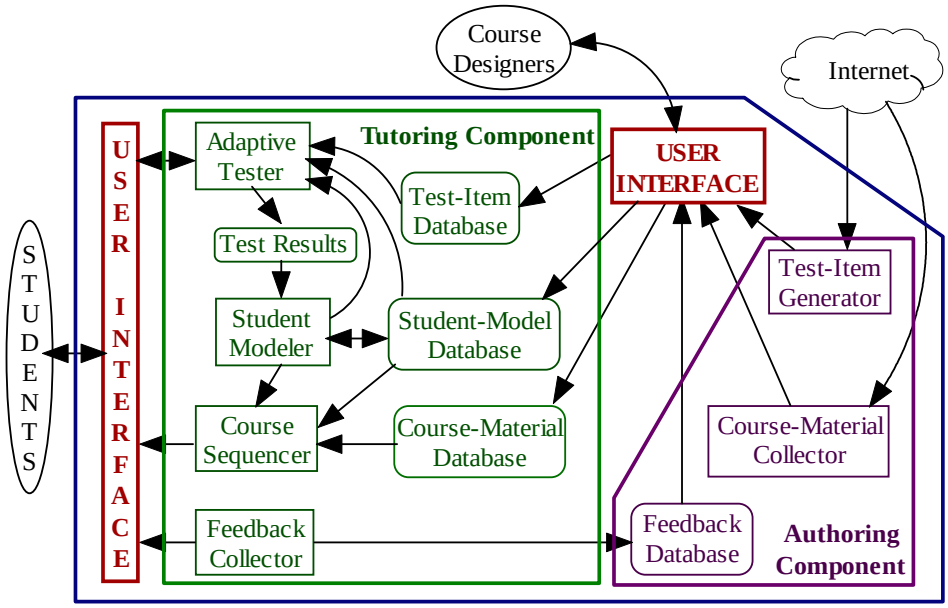

Figure 1. A functional structure of an intelligent tutoring system

As shown in Figure 1, the quality and quantity of test items are crucial to the success of the whole system, as the decisions for adaptive interactions with students depend heavily on students' responses to test items. Good test items help teachers identify students' competence levels more efficiently, and a large quantity of test items avoids the overuse of particular test items, thereby increasing the security of the item database [Dean and Sheehan 2003; Oranje 2003]. Although human experts can create test items of very high quality, the costs involved in using human experts exclusively in the authoring task can be formidable. It is thus not surprising that computer-assisted item generation (CAIG) has attracted the attention of educators and learners, who find that it offers several desirable features of generated item pools [Irvine and Kyllonen 2002]. CAIG offers the possibility of creating a large number of diverse items for assessing students' competence levels at relatively low cost, while alleviating problems related to keeping the test items secure [Dean and Sheehan 2003; Oranje 2003].

In this paper, we concern ourselves with a fundamental challenge for computer assisted language learning (CALL) and propose tools for assembling multiple-choice cloze items that are useful for assessing students' competency in the use of English vocabulary. If it is unable to determine ability to understand vocabulary, an ITS cannot choose appropriate materials for such CALL tasks as reading comprehension. To demonstrate our main ideas, we tackle the problem of generating cloze items for the college entrance examinations in Taiwan [Taiwan CEEC 2004]. (For the sake of brevity, we will henceforth use *cloze items* or *items* instead of *multiple-choice cloze items* when there is no obvious risk of confusion.) With the growth of the Web, we can search and sift online text sources for candidate sentences and come up with a list of cloze items economically with the help of natural language processing techniques [Gao and Liu 2003; Kornai and Sundheim 2003].

Techniques for natural language processing can be used to generate cloze items in different ways. One can create sentences from scratch by applying template-based methods [Dennis *et al.* 2002] or more complex methods based on some predetermined principles [Deane and Sheehan 2003]. One can also take existing sentences from a corpus and select those that meet the criteria for test items. Generating sentences from scratch provides a basis of obtaining specific and potentially well-controlled test items at the costs of more complex systems, e.g., [Sheehan *et al.* 2003]. On the other hand, since the Web puts ample text sources at our disposal, we can also filter texts to obtain candidate test items of higher quality. Administrators can then select really usable items from these candidates at relatively low cost.

Some researchers have already applied natural language processing techniques to compose cloze items. Stevens [1991] employed the concepts of concordancing and collocation to generate items using general corpora. Coniam [1997] applied factors such as word frequency in a tagged corpus to create test items of particular types. In previous works, we

considered both the frequencies and selectional preferences of words when utilizing the Web as the major source of sentences for creating cloze items [Gao and Liu 2003; Wang *et al.* 2003].

Despite the recent progress, more advanced natural language processing techniques have not yet been applied to generate cloze items [Kornai and Sundheim 2003]. For instance, many words in English carry multiple senses, and test administrators usually want to test a particular usage of a word. In this case, blindly applying a keyword matching method, such as a concordancer, may result in a long list of irrelevant sentences that will require a lot of postprocessing work. In addition, composing a cloze item requires more than just a useful sentence. Figure 2 shows a sample multiple-choice item, where we call the sentence with a gap the **stem**, the answer to the gap the **key**, and the other choices the **distractors** of the item. Given a sentence for a particular key, we still need distractors for a multiple-choice item. The selection of distractors affects the *item facility* and *item discrimination* of a cloze item and is a vital task [Poel and Weatherly 1997]. Therefore, the selection of distractors also calls for more deliberate strategies, and simple considerations alone, such as word frequency [Gao and Liu 2003; Coniam 1997], may not result in high-quality multiple-choice cloze items.

1. My sister is _____, that is, I am going to be an uncle soon.
   (A) supposing (B) assigning
   (C) expecting (D) scheduling

Figure 2. A multiple-choice cloze item for English

To remedy these shortcomings, we propose a novel integration of dictionary-based and unsupervised techniques for word sense disambiguation for use in choosing sentences in which the keys carry the senses chosen by test administrators. Our method also utilizes the techniques for computing collocations and selectional preferences [Manning and Schütze 1999] for filtering candidate distractors. Although we can find many works on word sense disambiguation in the literature [Edmonds *et al.* 2002], providing a complete overview on this field is not the main purpose of this paper. Manning and Schütze [1999] categorized different approaches into three categories: supervised, dictionary-based, and unsupervised methods. Supervised methods typically provide better chances of pinpointing the senses of polysemous words, but the cost of preparing training corpora of acceptable quality can be very high. In contrast, unsupervised methods can be more economical but might not produce high-quality cloze items for CALL applications. Our approach differs from previous dictionary-based methods in that we employ sample sentences of different senses in the lexicon as well as the definitions of polysemous words. We compare the definitions of the competing senses of the key based on a generalized notion of selectional preference. We also compare the similarities

between the candidate sentence, which may become a cloze item, and samples sentences which contain the competing senses of the key. Hence, our approach is a hybrid of dictionary-based and unsupervised approaches. Results of empirical evaluation show that our method can identify correct senses of polysemous words with reasonable accuracy and create items of satisfactory quality. In fact, we have actually used the generated cloze items in freshmen-level English classes at National Chengchi University.

We analyze the cloze items used in the college entrance examinations in Taiwan, and provide an overview of the software tools used to prepare our text corpus in Section 2. Then, we outline the flow of the item generation process in Section 3. In Section 4, we elaborate on the application of word sense disambiguation to select sentences for cloze items, and in Section 5, we delve into the application of collocations and selectional preferences to generate distractors. Evaluations, discussions and related applications of our approaches to the tasks of word sense disambiguation and item generation are presented in Section 6, which will be followed by the concluding section.

2. Data Analysis and Corpus Preparation

2.1 Cloze items for Taiwan College Entrance Examinations

Since our current goal is to generate cloze items for college entrance examinations, we analyzed the effectiveness of considering the linguistic features of cloze items with statistics collected from college entrance examinations administered in Taiwan. We collected and analyzed the properties of the test items used in the 1992-2003 entrance examinations. Among the 220 collected multiple-choice cloze items, the keys to the cloze items that were used in the examinations were only verbs (31.8%), nouns (28.6%), adjectives (23.2%) or adverbs (16.4%). For this reason, we will focus on generating cloze items for these four categories. Moreover, the cloze items contained between 6 and 28 words. Figure 3 depicts the distribution of the number of words in the cloze items. The mean was 15.98, and the standard deviation was 3.84.

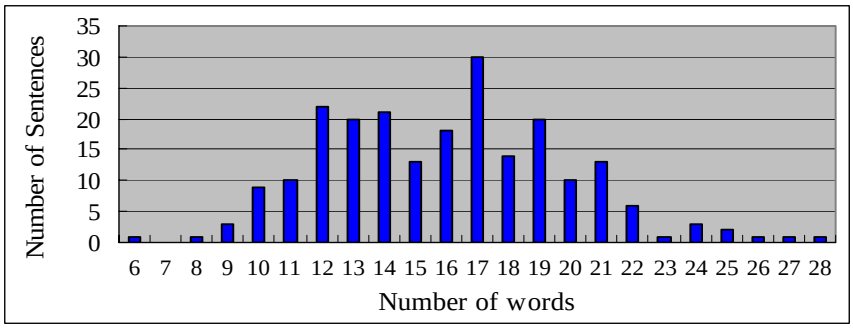

Figure 3. Distribution of the lengths of multiple-choice cloze items

In addition, the Web site of the College Entrance Examination Center provides statistics of testees' responses to a total of 40 multiple-choice cloze items that were used in college entrance examinations held in the years 2002 and 2003 [Taiwan CEEC 2004]. In each of these administrations, more than 110,000 students took the English test. The Web site contains statistics for the error rates of three different groups: ALL, HIGH, and LOW. The ALL group includes all testees, the HIGH group consists of testees whose overall English scores are among the top third, and the LOW group consists of testees whose scores are among the bottom third. Table 1 shows the correlations between the word frequency and selectional-preference (SP, henceforth) strengths of keys and distractors with the error rates observed in different student groups. We will explain how we calculated the frequencies and SP strengths of words in Sections 0 and 4.1, respectively.

From the perspective of correlation, the statistics slightly support our intuition that less frequent words make cloze items more difficult to solve. This claim holds for the ALL and HIGH groups in Table 1. However, the error rates for the LOW group do not correlate with the ranks of word frequency significantly. We suspect that this might be because examinees in the LOW group made more random guesses than average students did. We subtracted the error rates of the HIGH group from the error rates of the LOW group, and computed the correlation between the resulting differences between test items and the ranks of word frequency of the keys in the test items. The results are reported in the DIFF column. The DIFF column shows that using less frequent words in items reduced the items' ability to discriminate between students in the HIGH and LOW groups. The differences in error rates between these groups decreased when less frequent words were used in the cloze items. Figure 4 shows details of the relationships between the error rates and ranks of word frequency of the 40 items that we used to generate Table 1. Since the correlations are not very high, as shown in Table 1, clear trends are not apparent. The charts are included here to allow readers to make their own judgments as to how the error rates and ranks of word frequency are related.

In stark contrast, the correlations shown in the bottom half of Table 1 do not offer a consistent interpretation of the relationship between the error rates of different groups and the SP strengths. The negative numbers in the third row of statistics indicate that, when the SP strengths between the keys and stems increase, the error rates of all groups decrease. This is what one might expect. However, the negative statistics in the last row also suggest that as the SP strengths between the distractors and stems increase, the error rates decrease as well—a phenomenon quite hard to explain. We had expected to see the opposite trend, because distractors should be more misleading when they are more related to the stem. This surprising result might be due to the fact that selectional preference alone is not sufficient to explain students' performance in English tests. Identifying all the factors that can explain students' performance in language tests may require expertise in education, psychology, and linguistics,

which is beyond the expertise of the authors and the scope of this paper. Nevertheless, as we will show shortly, selectional preference can be instrumental in selecting sentences with desired word senses for use in the item-generation task.

Table 1. Correlations between linguistic features and (1) error rates of items for all students (ALL), (2) error rates of items for the top 33% of the students in the English tests (HIGH), (3) error rates of items of the bottom 33% of the students (LOW), and (4) the differences in error rates of items for the LOW and HIGH groups (DIFF)

		ALL	HIGH	LOW	DIFF
rank of word frequency (rank 1 is most frequent)	key	0.07	0.14	-0.07	-0.21
	distractors	0.11	0.15	0.03	-0.15
selectional-preference strength with the stem of the items	key	-0.17	-0.15	-0.07	0.13
	distractors	-0.20	-0.14	-0.21	0.00

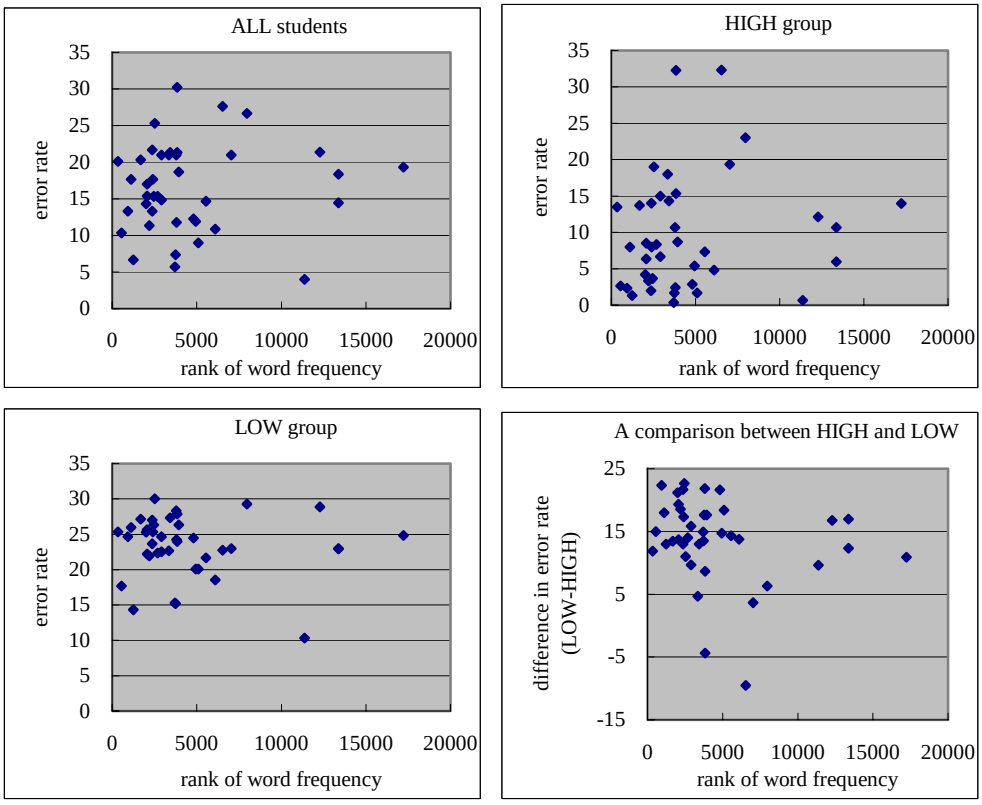

Figure 4. The relationships between error rates and rank of word frequency

2.2 Corpus Preparation and Lexicons

As indicated in Figure 1, a major step in our approach is acquiring sentences from the Web before we produce items. In this pilot study, we retrieved materials from *Studio Classroom* <www.studioclassroom.com>, the *China Post* <www.chinapost.com.tw>, *Taiwan Journal* <taiwanjournal.nat.gov.tw> and *Taiwan Review* <publish.gio.gov.tw> by using a Web crawler. We chose these online journals and news reports partially because they offer up-to-date news at a low spelling error rate and partially because that they can be downloaded at no cost. So far, we have collected in our corpus 163,719 sentences that contain 3,077,474 word tokens and 31,732 word types. Table 2 shows the statistics for verbs, nouns, adjectives, adverbs, and the whole database.

Table 2. Statistics of words in the corpus

	Verbs	Nouns	Adjectives	Adverbs	Overall
Word Tokens	484,673 (16%)	768,870 (25%)	284,331 (9%)	121,512 (4%)	3,077,474 (100%)
Word Types	5,047 (16%)	14,883 (47%)	7,690 (24%)	1,389 (4%)	31,732 (100%)

As a preprocessing step, we look for useful texts from Web pages that are encoded in the HTML format. We need to extract texts from titles, the main bodies of reports, and multimedia contents, and then segment the extracted paragraphs into individual sentences. We segment the extracted texts with the help of Reynar's MXTERMINATOR, which achieved 97.5% precision in segmenting sentences in the Brown and Wall Street Journal corpora [Reynar and Ratnaparkhi 1997]. We then tokenize words in the sentences before assigning useful tags to the tokens. Because we do not employ very precise methods for tokenization, strings may be separated into words incorrectly. Hence, although the statistics reported in Table 2 should be close to actual statistics, the numbers are not very precise.

We augment the texts with an array of tags that facilitate cloze item generation. We assign part-of-speech (POS) tags to words using Ratnaparkhi's MXPOST, which adopts the Penn Treebank tag set [Ratnaparkhi 1996]. Based on the assigned POS tags, we annotate words with their lemmas. For instance, we annotate *classified* with *classify* and *classified*, respectively, when the *classified* has *VBN* (i.e., past participle) and *JJ* (i.e., adjective) as its POS tags. We also mark the occurrences of phrases and idioms in sentences using Lin's MINIPAR [Lin 1998]. This partial parser also allows us to identify such phrases as *arrive at* and *in order to* that appear consecutively in sentences. This is certainly not sufficient for creating items for testing phrases and idioms, and we are currently looking for a better alternative.

MINIPAR mainly provides partial parses of sentences that we can use in our system. With these partial parses, words that are directly related to each other can be identified easily,

and we can apply these relationships between words in word sense disambiguation. For easy reference, we will call words that have a direct syntactic relationship with a word *W* as *W*'s **signal words** or simply **signals**.

After performing these preprocessing steps, we can calculate the word frequencies using the lemmatized texts. As explained in Section 2.1, we consider the most frequent word as the first word in the list, and order the words according to decreasing frequency. Also, as stated in Section 2.1, we focus on creating items for testing verbs, nouns, adjectives, and adverbs, we focus on the signals of words with these POS tags in sentences for disambiguating word senses, and we annotate such information in each sentence.

When we need lexical information about English words, we resort to machine readable lexicons. Specifically, we use WordNet <www.cogsci.princeton.edu/~wn/> when we need definitions and sample sentences of words for disambiguating word senses, and we consult HowNet <www.keenage.com> for information about classes of verbs, nouns, adjectives, and adverbs.

3. System Architecture

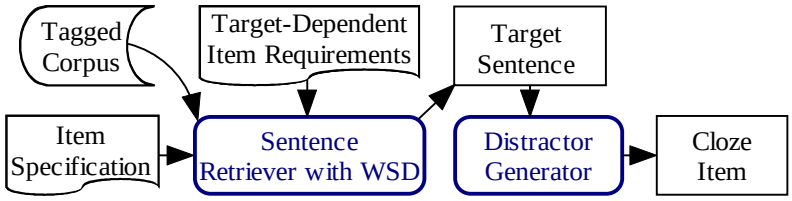

Figure 5. Main components of our cloze-item generator

We create cloze items in two major steps as shown in Figure 5. Constrained by the administrator's *Item Specification* and *Target-Dependent Item Requirements*, the *Sentence Retriever* selects a sentence for a cloze item from a *Tagged Corpus*, which we discussed its preparation in Section 0. Using the *Item Specification*, the test administrator selects the key for the desired cloze item and specifies the part-of-speech and sense of the key that will be used in the item. Figure 6 shows the interface of the *Item Specification*. Our system then attempts to create the requested items. The *Target-Dependent Item Requirements* specify general principles that should be followed in creating items for a particular test. For example, the number of words in cloze items in the college entrance examinations administered in Taiwan ranges from 6 to 28, and one may wish this as a guideline for creating drill tests. In addition, our system allows the test administrator to not specify the key and to request that the system provide a particular number of items for a particular part of speech instead.

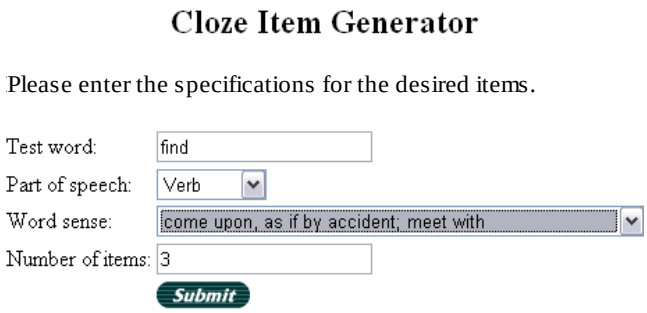

Figure 6. Interface for specifying cloze items

After retrieving the target sentences, the *Distractor Generator* considers such constraining factors as word frequency, collocations, and selectional preferences in selecting distractors. In cases where the generator cannot find sufficient distractors that go with the key and the target sentence, our system abandons the target sentence and starts the process all over again.

4. Target Sentence Retriever

The sentence retriever shown in Figure 5 extracts qualified sentences from the corpus. A sentence must contain the desired key with the requested POS to be considered as a candidate target sentence. We can easily conduct this filtering step using the MXPOS. Having identified a candidate sentence, the item generator needs to determine whether the sense of the key also meets the requirement. We conduct word sense disambiguation based on a generalized notion of selectional preferences.

4.1 Learning Strength of Generalized Selectional Preferences

Selectional preferences refer to the phenomenon that, under normal circumstances, some words constrain the meanings of other words in a sentence. A common illustration of selectional preferences is the case in which the word "chair" in the sentence "Susan interrupted the chair" must denote a person rather than a piece of furniture [Resnik 1997; Manning and Schütze 1999].

We extend this notion to the relationships between a word of interest and its signals, with the help of HowNet. HowNet provides the semantic classes of words; for instance, both *instruct* and *teach* belong to the class of *teach*, and both *want* and *demand* may belong to the class of *need*. Let *w* be a word of interest, and let π be the word class, defined in HowNet, of a signal of *w*. We denote the frequency with which both *w* and π participate in the syntactic relationship, *v*, as $f_v(w,\pi)$, and we denote the frequency with which *w* participates in the *v* relationship in all situations as $f_v(w)$. We define the strength of the selectional preference of

w and π under the relationship v as follows:

$$A_v(w,\pi) = \frac{f_v(w,\pi)}{f_v(w)}. \quad (1)$$

We consider limited types of syntactic relationships. Specifically, the signals of a verb include its subject(noun), object(noun), and the adverbs that modify the verb. Hence, the syntactic relationships for verbs include *verb-object*, *subject-verb*, and *verb-adverb*. The signals of a noun include the adjectives that modify the noun and the verb that uses the noun as its object or predicate. For instance, in "Jimmy builds a grand building," both "build" and "grand" are signals of "building." The signals of adjectives and adverbs include the words that they modify and the words that modify the adjectives and adverbs.

We obtain statistics about the strengths of selectional preferences from the tagged corpus. The definition of $f_v(w)$ is very intuitive and is simply the frequency with which the word w participates in a relationship v with any other words. We initialize $f_v(w)$ to 0 and add 1 to it every time we observe that w participates in a relationship v with any other words.

In comparison, it is more complex to obtain $f_v(w,\pi)$. Assume that s is a signal word that participates in a relationship v with w, and that the POS of s is x in this relationship. When s has only one possible sense under the POS x, and when the main class of this sole sense is π, we increase $f_v(w,\pi)$ by 1. (When HowNet uses multiple fundamental words to describe a sense, the leading word is considered the main class in our computation.) When s itself is polysemous, the learning step is a bit more involved. Assume that s has y possible senses under the POS x, and that the main classes of these senses belong to classes in $\Pi(s) = \{\pi_1,\ldots,\pi_i,\ldots,\pi_y\}$. We increase the co-occurrences of each of these classes and w, $f_v(w,\pi_i)$, i=1,…,y, by 1/y. We distribute the weight for a particular co-occurrence of π_i with w evenly, because we do not have a semantically tagged corpus. With MINIPAR, we only know what syntactic relationship holds between s and w. Without further information or disambiguating the signal words, we choose to weight each sense of s equally. Table 3 shows the statistics, collected from our corpus, for three verbs *eat*, *see* and *find* to take two classes of nouns, *Human* and *Food*, as their objects.

Table 3. Examples of the strengths of selectional preferences, $A_{verb-object}(w,\pi)$

Verb-Object	**Eat**	**See**	**Find**
Human	0.047	0.487	0.108
Food	0.441	0.005	0.057

4.2 Word Sense Disambiguation

We employ generalized selectional preferences to determine the sense of a polysemous word in a sentence. Consider the task of determining the sense of *spend* in the candidate target sentence "*They say film makers don't spend enough time developing a good story.*" The word *spend* has two possible meanings in WordNet.

1. (99) spend, pass – (pass (time) in a specific way; "How are you spending your summer vacation?")
2. (36) spend, expend, drop – (pay out; "I spend all my money in two days.")

Each definition of a possible sense includes (1) the **head words** that summarize the intended meaning, (2) a short explanation, and (3) a sample sentence. When we focus on the disambiguation of a word, we do not consider the word itself as a head word. Hence, *spend* has one head word, i.e., *pass*, in the first sense and two head words, i.e., *extend* and *drop*, in the second sense.

An intuitive method of determining the meaning of *spend* in a target sentence is to replace *spend* in the target sentence with its head words. The head words of the correct sense should fit into the target sentence better than head words of other competing senses. We judge whether a head word fits well into the position of the key based on the SP strength of the head word along with the word class of the signals of the key. Since a sense of the key may include many head words, we define the score of a sense as the average SP strength of the head words of the sense along with all the signal words of the key. This intuition leads to the first part of the total score for a sense, i.e., Ω_t, that we will present shortly.

In addition, we can compare the similarity of the contexts of *spend* in the target sentence and sample sentences, where *context* refers to the classes of the signals of the key being disambiguated. For the current example, we can compare whether the subject and object of *spend* in the target sentence belong to the same classes as the subjects and objects of *spend* in the sample sentences. The sense whose sample sentence offers a more similar context for *spend* in the target sentence receives a higher score. This intuition leads to the second part of the total score for a sense, i.e., Ω_s, that we will present below.

4.2.1 Details of Computing $\Omega_t(\theta_i \mid w,T)$: Replacing Keys with Head Words

Assume that word *w* has *n* senses in the lexicon. Let $\Theta = \{\theta_1, \ldots, \theta_i, \ldots, \theta_n\}$ be the set of senses of *w*. Assume that sense θ_i of word *w* has m_i head words in WordNet. (Note that we do not consider *w* as its own head word.) We use the set $\Lambda_i = \{\lambda_{i,1}, \lambda_{i,2}, \ldots, \lambda_{i,m_i}\}$ to denote the set of head words that WordNet provides for sense θ_i of word *w*.

When we use the partial parser to parse the target sentence *T* for a key, we obtain information about the signal words of the key. Moreover, when each of these signals is not

polysemous under their current POS tags, we look up their classes in HowNet and adopt the first listed class for each of the signals. Assume that there are $\mu(T)$ signals for the keyword *w* in a sentence *T*. We use the set $\Psi(T,w)=\{\psi_{1,T},\psi_{2,T},\ldots,\psi_{\mu(T),T}\}$ to denote the set of signals for *w* in *T*. Correspondingly, we use $\nu_{k,T}$ to denote the syntactic relationship between *w* and $\psi_{k,T}$ in *T*, and use $\Gamma(T,w)=\{\nu_{1,T},\nu_{2,T},\ldots,\nu_{\mu(T),T}\}$ to denote the set of relationships between signals in $\Psi(T,w)$ and *w*. Finally, we denote the class of $\psi_{k,T}$ as $\pi_{k,T}$ and the set of classes of the signals in $\Psi(T,w)$ as $\Pi(T,w)=\{\pi_{1,T},\pi_{2,T},\ldots,\pi_{\mu(T),T}\}$.

Recall that Equation (1) defines the strength of the selectional preference between a word and a class of the word's signal. Therefore, the following formula defines the averaged strength of the selectional preference of a head word $\lambda_{i,j}$ of sense θ_i of *w* with the signal words of *w* in *T*:

$$\frac{1}{\mu(T)}\sum_{k=1}^{\mu(T)}A_{\nu_{k,T}}(\lambda_{i,j},\pi_{k,T}).$$

When θ_i contains multiple head words, it is natural for us to compute the average strength of all the head words, excluding *w*. Hence, (2) measures the possibility of *w* taking sense θ_i in *T*. Note that $\Omega_t(\theta_i \mid w,T)$ must fall in the range [0,1] according to the definitions of (1) and (2):

$$\Omega_t(\theta_i \mid w,T)=\frac{1}{m_i}\sum_{j=1}^{m_i}\frac{1}{\mu(T)}\sum_{k=1}^{\mu(T)}A_{\nu_{k,T}}(\lambda_{i,j},\pi_{k,T}). \quad (2)$$

The computation of $\Omega_t(\theta_i \mid w,T)$ in (2) becomes more complicated when a signal, say $\psi_{k,T}$, of the key is polysemous. In this case, we face the problem of disambiguating the contextual information that we rely on for having to disambiguate the key. To terminate this mutual dependence between the senses of the key and the signal words, we use the average SP strength of the signal word in place of $A_{\nu_{k,T}}(\lambda_{i,j},\pi_{k,T})$. Specifically, we assume that $\psi_{k,T}$ has *q* senses when it participates in the $\nu_{k,T}$ relationship with the key, and we assume that the first listed classes of these *q* senses of $\psi_{k,T}$ are $\pi_{k,T,1},\pi_{k,T,2},\cdots,\pi_{k,T,q}$. We use the following definition of $A_{\nu_{k,T}}(\lambda_{i,j},\pi_{k,T})$ in Equation (2):

$$A_{\nu_{k,T}}(\lambda_{i,j},\pi_{k,T})=\frac{1}{q}\sum_{r=1}^{q}A_{\nu_{k,T}}(\lambda_{i,j},\pi_{k,T,r}). \quad (3)$$

4.2.2 Details of Computing Ω_s: Comparing the Similarity of Sample Sentences

Since WordNet provides sample sentences for important words, we can use the degrees of similarity between the sample sentences and the target sentence to disambiguate the word sense of the key in the target sentence. Let *T* and *S* be the target sentence of *w* and a sample sentence of sense θ_i of *w*, respectively. We treat it as a sign of similarity if the signal words that have the same syntactic relationships with the key in both sentences also belong to the

same class. Note specifically that we check the classes of the signal words, rather the signal words themselves. We compute this part of the score, Ω_S, for θ_i using the following three-step procedure. If there are multiple sample sentences for a given sense, say θ_i of *w*, we compute the score in (4) for each sample sentence of θ_i and use the average score as the final score for θ_i.

Procedure for computing $\Omega_S(\theta_i \mid w,T)$

1. We compute the signal words of the key and their relationships with the key in the target and sample sentences as follows:

   $\Psi(T,w) = \{\psi_{1,T}, \psi_{2,T}, \ldots, \psi_{\mu(T),T}\}$ (signal words of *w* in *T*),

   $\Psi(S,w) = \{\psi_{1,S}, \psi_{2,S}, \ldots, \psi_{\mu(S),S}\}$ (signal words of *w* in *S*),

   $\Gamma(T,w) = \{\nu_{1,T}, \nu_{2,T}, \ldots, \nu_{\mu(T),T}\}$ (syntactic relationships of signals words with *w* in *T*),

   $\Gamma(S,w) = \{\nu_{1,S}, \nu_{2,S}, \ldots, \nu_{\mu(S),S}\}$ (syntactic relationships of signals words with *w* in S).

2. We look for a pair $\psi_{j,T}$ and $\psi_{k,S}$ such that $\nu_{j,T} = \nu_{k,S}$, and check whether the matching $\psi_{j,T}$ and $\psi_{k,S}$ belong to the same word class. That is, for each signal of the key in *T*, we check the signals of the key in *S* for matching syntactic relationships (with the key) and word classes, and record the number of matched pairs in $M(\theta_i,T)$ for each θ_i. The matching process is complicated by the fact that signal words can be polysemous as well. When this situation occurs, the credit for each match is recorded proportionally. Assume that the signal word $\psi_{j,T}$ has $n_{j,T}$ possible classes, $\Pi(\psi_{j,T}) = \{\pi_{j,T,1}, \pi_{j,T,2}, ..., \pi_{j,T,n_{j,T}}\}$, when it participates in a $\nu_{j,T}$ relationship with *w* in the target sentence *T*. Assume that the signal word $\psi_{k,S}$ has $n_{k,S}$ possible classes, $\Pi(\psi_{k,S}) = \{\pi_{k,S,1}, \pi_{k,S,2}, ..., \pi_{k,S,n_{k,S}}\}$, when it participates in a $\nu_{k,S}$ relationship with *w* in a sample sentence *S*. If $\nu_{j,T} = \nu_{k,S}$, then we consider that there is a $1/n_{j,T}$ match whenever a class in $\Pi(\psi_{j,T})$ is matched by a class in $\Pi(\psi_{k,S})$. The pseudo code for computing $M(\theta_i,T)$ is as follows:

   (1) $M(\theta_i,T) = 0$;
   (2) mark all $\nu_{j,T} \in \Gamma(T,w)$, $j = 1, 2, ..., \mu(T)$, as unmatched;
   (3) for ($j = 0; j < \mu(T); j++$)
   (4) for ($k = 0; k < \mu(S); k++$)
   (5) if (($\nu_{j,T}$ unmatched) and ($\nu_{j,T} = \nu_{k,S}$))
   (6) for($l = 0; l < n_{j,T}; l++$)
   (7) for($m = 0; m < n_{k,S}; m++$)
   (8) if ($\pi_{j,T,l} = \pi_{k,S,m}$)
   (9) {

(10) mark $v_{j,T}$ as matched;

(11) $M(\theta_i,T) = M(\theta_i,T) + 1/n_{j,T}$

(12) }

3. The following score measures the proportion of matched relationships among all relationships between a sense θ_i of the key and its signals in the target sentence:

$$\Omega_s(\theta_i \mid w,T) = \frac{M(\theta_i,T)}{\mu(T)}. \quad (4)$$

4.2.3 Computing the Final Score for Each Sense

The score for *w* to take sense θ_i in a target sentence *T* is the sum of $\Omega_t(\theta_i \mid w,T)$ defined in (2) and $\Omega_s(\theta_i \mid w,T)$ defined in (4), so the sense of *w* in *T* will be set to the sense defined in (5) when the score exceeds a selected threshold. When the sum of $\Omega_t(\theta_i \mid w,T)$ and $\Omega_s(\theta_i \mid w,T)$ is too small, we avoid making arbitrary decisions about the word senses. There can be many other candidate sentences that include the key, so we can check the usability of these alternatives without having to stick to a sentence that we cannot disambiguate with sufficient confidence. We will discuss and illustrate effects of choosing different thresholds in Section 6.

$$\arg\max_{\theta_i \in \Theta} \quad \Omega_t(\theta_i \mid w,T) + \Omega_s(\theta_i \mid w,T) \quad (5)$$

5. Distractor Generation with Generalized Collocation

Distractors in multiple-choice items influence the possibility of guessing answers correctly. If we use extremely impossible distractors in the items, examinees may be able to identify the correct answers without really knowing the keys. Hence, we need to choose distractors that appear to fit the gap without having multiple possible answers to items in a typical cloze test.

There are principles and alternatives that are easy to implement and follow. Antonyms of the key are choices that average examinees will identify and ignore. The part-of-speech tags of the distractors should be the same as that of the key in the target sentence. Hence, a word will not be a good distractor if it does not have the same part of speech as the key of if it has affixes that indicate its part of speech. We may also take cultural background into consideration. Students with Chinese background tend to associate English words with their Chinese translations. Although this learning strategy works most of the time, students may find it difficult to differentiate between English words that have very similar Chinese translations. Hence, a culture-dependent strategy is to use English words that have similar Chinese translations as the key as distractors.

To generate distractors systematically, we employ word frequency ranks to select words

as candidates [Poel and Weatherly 1997; Wang *et al.* 2003]. Assume that we are generating an item for a key whose part of speech is ρ, that there are *n* word types whose parts of speech may be ρ in the dictionary, and that the word frequency rank of the key among these *n* word types is *m*. We randomly select words whose ranks fall in the range [*m*-*n*/10, *m*+*n*/10] among these *n* word types as candidate distractors. These distractors are then screened based on how well they fit into the target sentence, where *fitness* is defined based on the collocations of the word classes of the distractors and other words in the stem of the target sentence.

Since we do not examine the semantics of the target sentences, a relatively safe method for filtering distractors is to choose words that seldom collocate with important words in the target sentence. The "important words" are defined based on the parts of speech of the words and the syntactic structures of the target sentences. Recall that we have marked the words in the corpus with their signal words as discussed in Section 0. Those words that have more signal words in a sentence usually contribute more to the meaning of the sentence, so they should play a more important role in the selection of distractors. In addition, we consider words that have clausal complements in a sentence as important words. Let $T = \{t_1, t_2, \ldots, t_q\}$ denote the set of words, excluding the key, in the target sentence. We therefore define the set of important words $X \subseteq T$ such that each word in *X* either (1) has two or more signal words in *T* and is a verb, noun, adjective, or adverb, or (2) has a clausal complement.

Assume that $X \subseteq T$ is the set of important words in *T*, i.e., $X = \{x_1, x_2, \ldots, x_p\}$, $p \le q$. Let $\Pi(\kappa)$ and $\Pi(x_j)$, respectively, denote the sets of word classes of a candidate distractor κ and an important word x_j. Since we have no semantically tagged corpus, we will judge whether a candidate distractor fits the gap in the test item by checking the co-occurrence of the word class of the distractor and the word classes of the important words in the candidate sentence. A high co-occurrence score will strongly indicate that the candidate distractor is inappropriate.

Let $C = \{S_1, S_2, \ldots, S_\aleph\}$ denote the set of sentences in the corpus. We compute the pointwise mutual information between the word classes of a distractor κ and every important word in the target sentence, and take the average as the co-occurrence strength. Let $\zeta(S_i, \kappa)$ denote whether a sentence $S_i \in C$ contains a word whose word classes overlap with the word classes of κ. That is, $\zeta(S_i, \kappa)$ will be either 1 or 0, indicating whether a sense of κ is used in the sentence. Notice that it is not necessary for the word κ itself to be used. We define the probability of occurrence of any word class of κ as follows:

$$\Pr(\Pi(\kappa)) = \frac{1}{\aleph} \sum_{i=1}^{\aleph} \zeta(S_i, \kappa).$$

Analogously, we compute the probability of occurrence of any word class of an important word x_j, $\Pr(\Pi(x_j))$, as follows:

$$\Pr(\Pi(x_j)) = \frac{1}{\aleph}\sum_{i=1}^{\aleph}\zeta(S_i, x_j).$$

In addition, we let $\xi(S_i, \kappa, x_j)$ denote whether a sentence $S_i \in C$ uses a word with a word class in $\Pi(\kappa)$ and another word with a word class in $\Pi(x_j)$. Similar to $\zeta(S_i, \kappa)$, $\xi(S_i, \kappa, x_j)$ is also a Boolean variable. Using this notation, we define the co-occurrence of word classes in $\Pi(\kappa)$ and $\Pi(x_j)$ as follows:

$$\Pr(\Pi(\kappa), \Pi(x_j)) = \frac{1}{\aleph}\sum_{i=1}^{\aleph}\xi(S_i, \kappa, x_j).$$

Having obtained these probability values, we can compute the average pointwise mutual information of a candidate distractor with all of the important words in the target sentence as follows:

$$fit(\kappa) = \frac{-1}{p}\sum_{x_j \in X}\log\frac{\Pr(\Pi(\kappa), \Pi(x_j))}{\Pr(\Pi(\kappa))\Pr(\Pi(x_j))}. \quad (6)$$

We accept candidate words whose scores are better than 0.3 as distractors. The term inside the summation is the pointwise mutual information between κ and x_j, where we consider not the occurrences of the words but the occurrences of their word classes. We negate the averaged sum so that classes that seldom collocate will receive higher scores, thus avoiding multiple answers to the resulting cloze items. We set the threshold to 0.3, based on statistics about (6) that were calculated based on the cloze items administered in the 1992-2003 college entrance examinations in Taiwan.

6. Evaluations, Analyses, and Applications

6.1 Word Sense Disambiguation

Word sense disambiguation is an important topic in natural language processing research [Manning and Schütze 1999]. Different approaches have been evaluated in different setups, and a very wide range of achieved accuracy [40%, 90%] has been reported [Resnik 1997; Wilks and Stevenson 1997]. Hence, objective comparison between different approaches is not a trivial task. It requires a common test environment like SENSEVAL [ACL SIGLEX 2005]. Therefore, we will only present our own results.

Table 4. Accuracy in the WSD task

POS of the key	Baseline	Threshold = 0.4	Threshold = 0.7
Verb	38.0%(19/50)	57.1%(16/28)	68.4%(13/19)
Noun	34.0%(17/50)	63.3%(19/30)	71.4%(15/21)
Adjective	26.7%(8/30)	55.6%(10/18)	60.0%(6/10)
Adverb	36.7%(11/30)	52.4%(11/21)	58.3%(7/12)

We arbitrarily chose 160 sentences that contained polysemous words for disambiguation. A total of 50, 50, 30, and 30 samples were selected for polysemous verbs, nouns, adjectives, and adverbs, respectively. We chose these quantities of sentences based on the relative frequencies of 31.8%, 28.6%, 23.2%, and 16.4% that we discussed in Section 2. We measured the percentages of correctly disambiguated words in these 160 samples, and Table 4 shows the results. In calculating the accuracy, we used the definitions of word senses in WordNet.

The **baseline** column shows the resulting accuracy when we directly used the most common sense, as recorded in WordNet, for the polysemous words. For example, using the definitions of *spend* given in Section 4.2, the first alternative is the default sense of *spend*. The rightmost two columns show the resulting accuracy achieved with our approach when we used different thresholds for applying (5). Our system made fewer decisions when we increased the threshold, as we discussed previously in Section 4.2, and the threshold selection evidently affected the precision of word sense disambiguation evidently. Not surprisingly, a higher threshold led to higher precision, but the rejection rate increased at the same time. For instance, when the threshold was 0.4, our system judged the keys in 28 sentences for verbs, and, when the threshold increased to 0.7, only 19 judgments were made by our system. Out of these 28 and 19 judgments, 16 and 13 were correct, respectively. Sentences whose total scores did not exceed the chosen threshold were simply dropped. Since the corpus can be extended to include more and more sentences, we have the luxury of ignoring sentences and focusing on the precision rather than the rejection rate of the sentence retriever.

6.2 Cloze Item Generation

Figure 7 shows a sample output for the specification shown in Figure 6. Given the generated items, the test administrator can choose the best items via the interface for compiling test questions. Although we have not implemented the post-editing component completely, teachers will be allowed to change the words in the recommended test items and organize the test items according to each teacher's preferences.

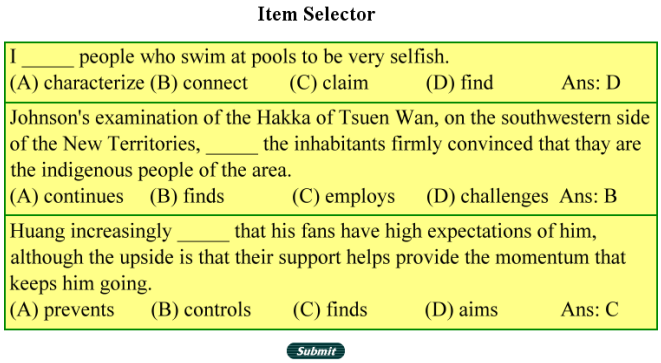

Figure 7. Items generated by the session shown in Figure 6

We asked the item generator to create 200 items in the evaluation. To mimic the distribution of real world examinations, we allocated different numbers of items for verbs, nouns, adjectives, and adverbs based on the proportions of 31.8%, 28.6%, 23.2%, and 16.4% that we reported in Section 2. Hence, we used 77, 62, 35, and 26 items for verbs, nouns, adjectives, and adverbs, respectively, in the evaluation.

Table 5. Correctness of the generated sentences (with the chosen POS tags and senses)

POS of the key	Number of items	% of correct sentences
Verb	77	66.2%
Noun	62	69.4%
Adjective	35	60.0%
Adverb	26	61.5%
Overall		65.5%

In the evaluation, we requested one item at a time and examined whether the sense and part of speech of the key in the generated item really met the requirements. The threshold for using (5) to disambiguate word sense was set to 0.7. The results of this experiment, shown in Table 5, do not differ significantly from those reported in Table 4. For all four major targets of cloze tests, our system was able to return one correct sentence for less than two target sentences it generated. This result is not surprising, as the WSD task is the bottleneck. Putting constraints on the POS would not change the performance significantly. Notice that we generated two different sets of sentences to collect the statistics shown in Tables 4 and 5, so the statistics vary for the same POS.

In addition, we checked the quality of the distractors and marked those items that had only one correct answer as good items. We asked our system to generate another 200 test items and manually determined whether the generated items each had one solution. Table 6 shows that our system was able to create items with unique answers most of the time. It appears that choosing good distractors for adverbs is the most challenging task. Using different adverbs to modify a sentence affects the meaning of the resulting sentence, but it is relatively less likely that using different adverbs as the modifiers will affect the correctness of the sentence. Hence, it is more likely to have multiple possible answers to test items whose keys are adverbs.

Table 6. Uniqueness of answers to the composed test items

Item category	POS of the key	Number of items	Results
Cloze	**Verb**	64	90.6%
	Noun	57	94.7%
	Adjective	46	93.5%
	Adverb	33	84.8%
		overall	91.5%

6.3 Discussion

The head words and sample sentences in the entries of lexicons provide good guidance for word sense disambiguation. Florian and Wicentowski's unsupervised methods that apply information in WordNet and unlabeled corpora are similar to our method, but only the best performer among their methods offers results that are comparable to our results [Florian and Wicentowski 2002]. (We have to note that the comparison made here is based on reported statistics, and that a fair comparison would require using both systems to disambiguate the same set of test data.) Hence, we are quite encouraged by the current performance of our system. Nevertheless, our approach to word sense disambiguation does have the following problems.

We note that not every sense of every word has sample sentences in WordNet. When a sense does not have any sample sentence, this sense will receive no credit, i.e., 0, for $\Omega_s(\theta_i \mid w,T)$. Consequently, our current reliance on sample sentences in the lexicon causes us to discriminate against senses that do not have sample sentences. This is an obvious drawback in our current design, but this problem is not really detrimental or unsolvable. There are usually sample sentences for important and commonly-used senses of polysemous words, so we hope that this discrimination problem will not occur frequently. To solve this problem once and for all, we could customize WordNet by adding sample sentences for all the senses of important words, though we do not imagine that this is a trivial task.

MINIPAR gives only one parse for a sentence, and we have no guarantee of obtaining the correct parses for our sentences. However, this might not be a big problem as our sentences are relatively short. Recall that our system attempts to choose sentences that contained between 6 and 28 words (with an average of about 16 words). Although such short sentences can still be parsed in multiple ways syntactically, short sentences are usually not syntactically ambiguous, and MINIPAR may work satisfactorily.

Using contextual information to disambiguate words is not as easy as we expected. The method reported in this paper is not perfect, and the resulting precision leaves large room for improvement. When we use selectional preference to compute $\Omega_t(\theta_i \mid w,T)$ in (2), we do not attempt to disambiguate the polysemous signal words of the key. We choose to assume that a polysemous signal word will take on each of the possible senses with equal chances in (3). We allow ourselves to avoid the disambiguation of polysemous signal words by this simplifying decision, so introduce errors in the recommended cloze items when the signal words are polysemous. Were the main goal of our research word sense disambiguation, we would have to resort to a more fully-fledged mechanism when a sentence contained multiple ambiguous words. Identifying the topic or the discourse information about the texts from where the target sentences are extracted are possible ways for disambiguating the signal words, and there are quite a few such work in the literature [Manning and Schütze 1999].

An individual sentence that is extracted from a larger context, e.g., a paragraph, may not contain sufficient information for students to understand the extracted sentence. If understanding the target sentence requires information not contained in the target sentence, it will not be a good idea to use this sentence as a test item, because this extra factor may introduce unnecessary noise that prevents students from answering the test item correctly.

Dr. Lee-Feng Chien of Academia Sinica has pointed out that our use of the sense definitions in WordNet may have demanded unnecessary quality for the word sense disambiguation task. WordNet includes more fine-grained differentiations of senses that may exceed the needs of ordinary learners of English.

The aforementioned weaknesses should not overshadow the viability of our approach. The experimental results obtained in our pilot study indicate that, with our method, one can implement a satisfactory cloze item generator at relatively low cost. Although we must admit that the weaknesses of our approach could become problems if we targeted at a fully automatic item generation [Bejar *et al.* 2003], we suspect that a fully automatic item generator would offer items of appropriate quality for our current application. In our approach to generating cloze items, a reasonable error rate in the word sense disambiguation task is tolerable because human experts will review and select the generated items anyway. As long as we can confine the error rates within a limited range, the computer-assisted generation process will increase the overall productivity.

6.4 More Applications

We have used the generated items in real tests in a freshman-level English class at National Chengchi University, and have integrated the reported item generator in a Web-based system for learning English [Gao and Liu 2003]. In this system, we have two major subsystems: an authoring subsystem and an assessment subsystem. Using the authoring subsystem, test administrators can select items through the interface shown in Figure 7, save the selected items to an item pool, edit the items, including their stems if necessary, and finalize the selection of items for a particular examination. Using the assessment subsystem, students can answer the test items via the Internet, and receive scores immediately if the administrators choose to provide them. Student's answers are recorded for student modeling and for the analysis of item facility and item discrimination.

In addition to supporting cloze tests, our system also can create items for testing idioms and phrases. Figure 8 shows the output of this function. However, we can only support consecutive phrases at this moment. Moreover, we are currently expanding our system to help students with listening and dictation in English [Huang *et al.* 2005]. Our long-term plans are to expand our system to support more aspects of learning English and to enable our system to adapt to students' competency [Liu 2005].

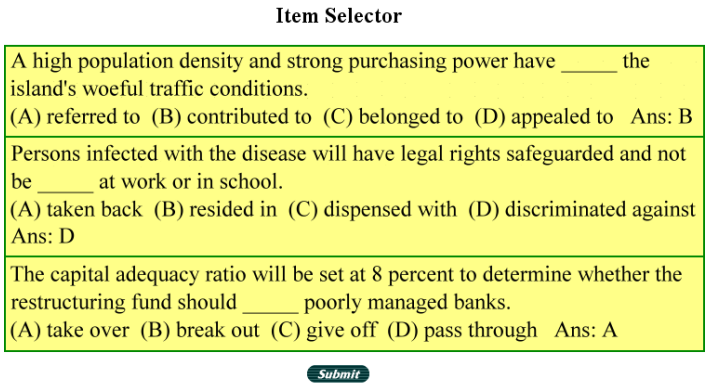

Figure 8. Sample items for testing English phrases

7. Concluding Remarks

Natural language processing techniques prove to be instrumental for creating multiple-choice cloze items that meet very specific needs of test administrators. By introducing word sense disambiguation into the item generation process, we enable each generated cloze item to include words that carry the desired senses. Word sense disambiguation itself is not a trivial task and has been studied actively for years. Although our approach does not lead to perfect selections of the word senses in target sentences, its performance is comparable to that of some modern methods for word sense disambiguation, and we have shown that it can provide crucial aid in the item generation task. After all, it is well known that word sense disambiguation may require information about contexts that cover more than just individual sentences, and that high-quality disambiguation within an individual sentence can be very difficult, if not impossible to achieve [Manning and Schütze 1999].

We have also proposed a new approach to selecting distractors for multiple-choice cloze items. Using the proposed collocation-based measure and word frequencies, we are able to identify distractors that are similar in challenge level with the key of the item, while guaranteeing that there is only one answer to the item about 90% of the time.

Since test administrators can request our system to return multiple items and manually select the best ones for composing tests, it is not absolutely necessary for us to create a perfect item generator. Our system currently generates one usable cloze item for every 1.6 generated items. Nevertheless, we intend to improve this result by considering more advanced linguistic features in sense disambiguation, and will update the results in the near future.

Acknowledgements

We thank Dr. Lee-Feng Chien, the editors of this special issue, and the anonymous reviewers for their invaluable comments on previous versions of this paper. We must admit that we have

not been able to follow all their suggestions for improving this short article, and we are responsible for any remaining problems in this paper. The authors would also like to thank Professor I-Ping Wan of the Graduate Institute of Linguistics of National Chengchi University for adopting test items generated by our system in her English classes in the 2003 Fall semester. This research was supported in part by grants 91-2411-H-002-080, 92-2213-E-004-004, 92-2411-H-002-061, 93-2213-E-004-004, and 93-2411-H-002-013 from the National Science Council of Taiwan. This paper is an expanded version of [Wang *et al.* 2004a] and [Wang *et al.* 2004b].

References

ACL SIGLEX, SESEVAL homepage, http://www.senseval.org/, 2005.

Bejar, I. I., R. R. Lawless, M. E. Wagner, R. E. Bennett and J. Revuelta, "A Feasibility Study of On-the-Fly Item Generation in Adaptive Testing," *Journal of Technology, Learning and Assessment*, 2, 2003, http://www.jtla.org/.

Computational Linguistics, Special issue on word sense disambiguation, 24(1), 1998.

Coniam, D., "A Preliminary Inquiry into Using Corpus Word Frequency Data in the Automatic Generation of English Language Cloze Tests," *Computer Assisted Language Instruction Consortium*, 16(2–4), 1997, pp. 15–33.

Deane, P. and K. Sheehan, "Automatic Item Generation via Frame Semantics," Educational Testing Service: http://www.ets.org/research/dload/ncme03-deane.pdf, 2003.

Dennis, I., S. Handley, P. Bradon, J. Evans and S. Nestead, "Approaches to Modeling Item Generative Tests," *Item Generation for Test Development*, ed. by Irvine and Kyllonen, 2002, pp. 53–72.

Edmonds, P., R. Mihalcea, and P. Saint-Dizier, editors, *Proceedings of the ACL 2002 Workshop on Word Sense Disambiguation: Recent Successes and Future Directions*, Association for Computational Linguistics, 2002.

Gao, Z.-M. and C.-L. Liu, "A Web-Based Assessment and Profiling System for College English," In *Proceedings of the Eleventh International Conference on Computer Assisted Instruction*, 2003, CD–ROM.

Huang, S.-M., C.-L. Liu and Z.-M. Gao, "Computer-Assisted Item Generation for Listening Cloze and Dictation Practice in English," In *Proceedings of the Fourth International Conference on Web-Based Learning*, 2005, forthcoming.

Irvine, S. H. and P. C. Kyllonen, editors, *Item Generation for Test Development*, Lawrence Erlbaum Associates, 2002.

Kornai, A. and B. Sundheim, editors, *Proceedings of HLT-NAACL 2003 Workshop on Building Educational Applications Using Natural Language Processing*, Association for Computational Linguistics, 2003.

Liu, C.-L., "Using Mutual Information for Adaptive Item Comparison and Student Assessment," *Journal of Educational Technology & Society*, 8(4), 2005, forthcoming.

Lin, D., "Dependency-Based Evaluation of MINIPAR," In *Proceedings of the Workshop on the Evaluation of Parsing Systems in the First International Conference on Language Resources and Evaluation*, 1998.

Manning, C. D. and H. Schütze, *Foundations of Statistical Natural Language Processing*, The MIT Press, Cambridge, 1999.

Oranje, A., "Automatic Item Generation Applied to the National Assessment of Educational Progress: Exploring a Multilevel Structural Equation Model for Categorized Variables," Educational Testing Service: http://www.ets.org/research/dload/ncme03-andreas.pdf, 2003.

Poel, C. J. and S. D. Weatherly, "A Cloze Look at Placement Testing," *Shiken: JALT* (Japanese Association for Language Teaching) *Testing & Evaluation SIG Newsletter*, 1(1), 1997, pp. 4–10.

Ratnaparkhi, A., "A Maximum Entropy Part-of-Speech Tagger," In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 1996, pp. 133–142.

Resnik, P., "Selectional Preference and Sense Disambiguation," In *Proceedings of the Applied Natural Language Processing Workshop on Tagging Text with Lexical Semantics: Why, What and How*, 1997, pp. 52–57.

Reynar, J. C. and A. Ratnaparkhi, "A Maximum Entropy Approach to Identifying Sentence Boundaries," In *Proceedings of the Conference on Applied Natural Language Processing*, 1997, pp. 16–19.

Sheehan, K. M., P. Deane, and I. Kostin, "A Partially Automated System for Generating Passage-Based Multiple-Choice Verbal Reasoning Items," paper presented at the National Council on Measurement in Education Annual Meeting, 2003.

Stevens, V., "Classroom Concordancing: Vocabulary Materials Derived from Relevant Authentic Text," *English for Specific Purposes*, 10(1), 1991, pp. 35–46.

Taiwan College Entrance Examination Center (CEEC): Statistics about 2002 and 2003 entrance examinations, 2004, http://www.ceec.edu.tw/exam/e_index.htm/.

van der Linden, W. J. and R. K. Hambleton, editors, *Handbook of Modern Item Response Theory*, Springer, New York, USA, 1997.

van der Linden, W. J. and C. A. W. Glas, editors, *Computerized Adaptive Testing: Theory and Practice*, Kluwer, Dordrcht, Netherlands, 2000.

Wang, C.-H., C.-L. Liu and Z.-M. Gao, "Toward Computer Assisted Item Generation for English Vocabulary Tests," In *Proceedings of the 2003 Joint Conference on Artificial Intelligence, Fuzzy Systems, and Grey Systems*, 2003, CD–ROM.

Wang, C.-H., C.-L. Liu and Z.-M. Gao, "利用自然語言處理技術自動產生英文克漏詞試題之研究," In *Proceedings of the Sixteenth Conference on Computational Linguistics and Speech Processing*, 2004a, pp. 111–120. (in Chinese)

Wang, C.-H., C.-L. Liu and Z.-M. Gao, "Using Lexical Constraints for Corpus-Based Generation of Multiple-Choice Cloze Items," In *Proceedings of the Seventh IASTED*

International Conference on Computers and Advanced Technology in Education, 2004b, pp. 351–356.

Wilks, Y. and M. Stevenson, "Combining Independent Knowledge Sources for Word Sense Disambiguation," In *Proceedings of the Conference on Recent Advances in Natural Language Processing*, 1997, pp. 1–7.