

## FAST- An Automatic Generation System for Grammar Tests

Chia-Yin Chen

Inst. of Info. Systems & Applications

National Tsing Hua University

101, Kuangfu Road,

Hsinchu, 300, Taiwan

G936727@oz.nthu.edu.tw

Hsien-Chin Liou

Dept. of Foreign Lang. & Lit.

National Tsing Hua University

101, Kuangfu Road,

Hsinchu, 300, Taiwan

hcliou@mx.nthu.edu.tw

Jason S. Chang

Dept. of Computer Science

National Tsing Hua University

101, Kuangfu Road,

Hsinchu, 300, Taiwan

jschang@cs.nthu.edu.tw

### Abstract

Testing has long been acknowledged as an integral part of language teaching; however, manually-designed tests are not only time consuming but also labor intensive. Lately, due to the remarkable progress of computer technology, computer-assisted item generation (CAIG) has drawn considerable attention and becomes one of the core subjects in CALL (Computer Assisted Language Learning). CAIG provides an alternative way to automatically generate questions in relatively short time, effectively establish item banks in a large scale, and possibly support adaptive testing for incremental language learning, solving the underlying problems of time and expenditure. Previous work has explored the generation of reading comprehension, vocabulary, and cloze questions, but very little has been done on grammar tests. The purpose of this paper is to address the issue of the automatic creation of English grammar tests.

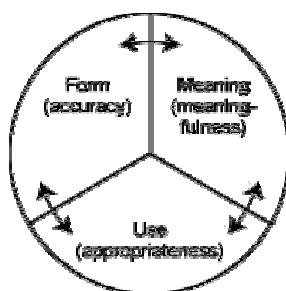
We introduce a method to automatically generate grammar test items by applying Natural Language Processing (NLP) techniques. Based on test patterns adapted from TOEFL (Test of English as Foreign Language), sentences gathered from the Web are transformed into tests on grammaticality. The method involves representing test writing knowledge as structural patterns, acquiring authentic sentences on the Web, and applying patterns to transform sentences into items. At runtime, sentences are converted into two types of TOEFL-style question: multiple-choice and error detection. A prototype system *FAST* (Free Assessment of Structural Tests) with initial evaluation on a set of generated questions indicates that the proposed method has great potential as an educational application. Our methodology provides a promising approach and offers significant potential for computer assisted language learning and assessment.

Key word: computer-assisted item generation, computer assisted language learning, grammar

### Introduction

Language testing, aiming to assess learners' language ability, is an essential part both for language teaching and learning. Language tests provide information of the results of learning and instruction and can be used to make decisions about individuals (e.g., placement test, achievement test, and diagnostic test). Among all kinds of tests, grammar test is commonly used in every education setting and is always included in well-established standardized tests like TOEFL (Test of English as Foreign Language).

According to Larsen-Freeman (1997), grammar comprises three dimensions: form, meaning and use (See Figure 1) and hence the goal of grammar testing is to test learners to use grammar accurately, meaningfully, and appropriately. Consider the present progressive tense in English. The present progressive form is composed of the verb “be” and the gerund. The grammatical meaning of the present progressive can be (1) representing the behavior on the instant: “I am writing a letter now.”; (2) describing the current situation: “I am looking for a job at the moment.”. When having to express one of these two meanings, learners use the present progressive tense. Therefore, a comprehensible grammar question needs to examine learners’ grammatical knowledge from all three aspects (morphosyntax, semantics and pragmatics).



(Larsen-Freeman, 1997)

Figure 1: Three Dimensions of Grammar

Close (1982) defines grammar as knowledge of sentence level form, resulting in multiple-choice items we are all familiar with. As the most common way of testing grammar is the multiple-choice test (Kathleen and Kenji, 1996), issues about validity and reliability on this test have been further investigated. Some researchers have reported that it is reliable and valid to evaluate learners’ grammatical competence through multiple-choice format (Rau, 1999; Purpura, 2004). Research on how to create an effective multiple-choice test item has also drawn lots of attention (Wang & Liou, 1991; Chang, 2000).

Multiple-choice test format on grammaticality consists of two kinds: one is traditional multiple-choice and the other is error detection. Multiple-choice question has merits in that (1) it is easy for test takers to select answers, (2) it is objective and easy to score, and (3) lots of grammar points can be covered quickly. Standardized tests such as TOEFL and GRE (Graduate Record Examinations) are examples of multiple-choice tests.

Figure 2 shows a typical example of traditional multiple-choice item. As for Figure 3, it shows a sample of error detection question. Traditional multiple-choice is made up of three components, where we define the sentence with a gap as the stem, the correct choice to the gap as the key and the other erroneous choices as the

distractors. (In Figure 2, the blanked sentence acts as the stem and the key “*the largest*” is accompanied by three distractors of “*the large*”, “*the larger*”, and “*most largest*”.) As to error correction, it consists of a partially underlined sentence (stem), one correct choice for the correction (key) and three other distractors to distract test takers. In Figure 3, the stem is “*Although maple trees are among the most colorful varieties in the fall, they lose its leaves sooner than oak trees.*” and “*its*” is the key with distractors “*among*”, “*in the fall*”, and “*sooner than*”.

<p>The blue whale is _____ known animal, reaching a length of more than one hundred feet.</p> <p>(A) the large</p> <p>(B) the larger</p> <p>(C) the largest</p> <p>(D) most largest</p>
---

Figure 2: An example of traditional multiple-choice test.

<p>Although maple trees are <u>among</u> the most colorful varieties <u>in the fall</u>,</p> <p style="text-align: center;">(A) <span style="float: right;">(B)</span></p> <p>they lose <u>its</u> leaves <u>sooner than</u> oak trees.</p> <p style="text-align: center;">(C) <span style="float: right;">(D)</span></p>
---

Figure 3: An example of error detection test.

Grammar tests are widely used to measure learners’ grammatical knowledge; however, it is costly to manually design these questions. In recent years, some attempts (Coniam, 1997; Mitkov and Ha, 2003; Liu et al., 2005) have been made on the automatic generation of language testing. Nevertheless, no attempt has been made to generate English grammar tests in a fully automatic way. Additionally, previous research merely focuses on generating questions of traditional multiple-choice task, none has been made for the automatic generation of correction type questions.

In this paper, we present a novel approach to automatically generate grammar tests of traditional multiple-choice and error correction types. First, by analyzing the characteristics of grammar questions from well-known standardized tests like TOEFL, we write a number of patterns for the development of structure tests. For example, a verb-related pattern requires an infinitive in the complement (e.g., the verb “*seem*”). For each pattern, distractors are created for the completion of each grammar question. As in the example of “*seem to evolve*”, wrong alternatives are constructed by changing the verb “*evolve*” into different forms: “*to evolving*”, “*evolve*”, and “*evolving*”. Then, we collect authentic sentences from the Web as the source of the tests. Finally, by applying different generation strategies, grammar tests in two test

formats are produced. A complete grammar question in traditional multiple-choice format is generated as shown in Figure 4. Intuitively, based on certain surface patterns (See Figure 5), computer is able to compose the grammar question presented in Figure 4. We have implemented a prototype system *FAST* and initial experiment results have shown that about 70 grammatical patterns can be successfully converted into grammar tests by using sentences accumulated from the Web.

Representative democracy seemed \_\_\_\_\_ simultaneously during the eighteenth and nineteenth centuries in Britain, Europe, and the United States.

(A) to evolve  
(B) to evolving  
(C) evolving  
(D) evolve

Figure 4: An example of generated question.

```
* X/INFINITIVE * PP.  
_____  
: _____ : PP  
(A) X/INFINITIVE  
(B) X/VBG  
(C) X/VBG  
(D) X/VR
```

Figure 5: An example of surface pattern.

We review relevant previous work in Section 2. In Section 3, we introduce the algorithm for the automatic question generation. As for Section 4, it includes the evaluation results. Finally, we make a brief conclusion (Section 5).

### Related Work

Since the mid-80s, item generation for test development has been an area of active research. Currently, the practice of creating assessment items algorithmically has been attracting more and more attention as computer technology advances. In our work, we address an aspect of CAIG (computer-assisted item generation) centers on the automatic construction of grammar tests. We also implement a pattern-based system of automated grammar question generation.

CAIG has played a crucial role in language learning. Wolfe (1976) developed AUTOQUEST to automatically generate questions for independent study of written texts. Wilson (1997) employed a corpus-based concept to automate the generation of CALL exercises. Later in 2001, Shei introduced a system (FollowYou!) for the automatic language lesson generation. In 2003, Huang et al. exploited on the learning

for English dictation and created relevant teaching materials and drills with the assistance of computer.

For language assessment, CAIG has served to generate questions (in multiple-choice format) by applying natural language processing (NLP) techniques. Mitkov and Ha (2003) established a system which generates reading comprehension tests in a semi-automatic way by employing a NLP-based approach to extract key concepts from instructional documents, convert sentences with key terms into questions based on certain transformation rules, and obtain semantically related alternatives from WordNet.

As to vocabulary assessment, Coniam in 1997 described a process to compose vocabulary test items relying on corpus word frequency data. Gao (2000) presented the system AWETS that semi-automatically produce vocabulary tests based on word frequency and part-of-speech tagging information. Recently, Wang et al. (2003), with the analysis of selectional preference and the use of machine learning, introduced an approach to compose vocabulary tests with identical features of JCEE (Joint College Entrance Exam). Hoshino and Nakagawa (2005) established a real-time system of automatically generating vocabulary questions using machine learning techniques. In the same year, Brown et al. introduced a method to automatically generate 6 types of vocabulary questions by employing data from WordNet.

More recently, Liu et al. (2005) reported an approach for the construction of listening test. By considering phonetic features of different words and calculating the similarity between sounds, they successfully composed tests for listening assessment.

In addition to the measurement of reading ability, listening proficiency, and vocabulary knowledge, cloze test is another conventional language assessment. Liu et al. (2005) applied word sense disambiguation-based method and collocation-based approach for the automatic composing of English cloze items. Sumita et al. (2005) proposed a method that constructs cloze questions using a corpus, a thesaurus, and Web-based data.

In contrast to previous work emphasizing the automatic generation of reading comprehension, vocabulary, and cloze questions, we present a system that allows grammar test writers to represent common patterns of test items and distractors. With these patterns, the system automatically gathers authentic sentences and generates grammar test items.

### **The *FAST* System**

Current research does not explore the automatic question generation for grammar assessment. To cope with this issue, we present a promising approach to systematically convert syntactic features into test patterns, automatically collect Web data as candidate question stems, and specifically apply deliberate test strategy to two

different tasks. We sketch the flow of the question generation of *FAST* system in Figure 6.

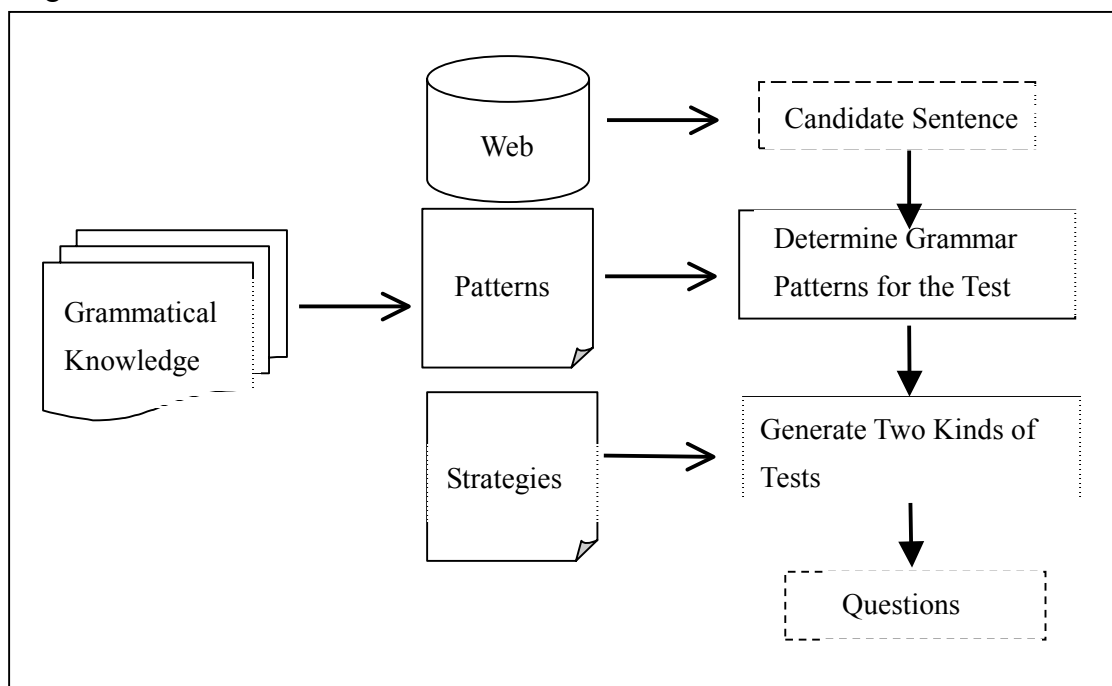


Figure 6: The flow of the generation procedure.

Grammar tests usually follow a set of patterns covering syntactic categories of the English grammar. These patterns (including structural patterns and distractor generation patterns) are easily conceptualized and written down. We compose a structural pattern by analyzing syntactic characteristics of English sentences. For instance, if the construct of the test is the use of past tense, we can construct a structural pattern as *{\*PPSS VBD\*}* by examining the sentence “*I lived in a small town in Ohio in my early childhood.*” The pattern is written in terms of part of speech tags produce by a POS tagger.

Based on each constructed structural pattern, we design distractors separately. Distractors are usually some words in the grammatical pattern with some modification: changing part of speech, adding, or deletion of words. The symbol *\$0* designates the key word in the grammatical pattern, while *\$9* and *\$1* are the word preceding and following the key word respectively. By way of example, distractors of the preliminary pattern are: *{\$0 VBG}* (meaning the key word with part of speech changed to “*vbg*”), *{\$0 VB}*, and *{\$0 life}* (meaning the word “*life*”).

To sum up, a complete test pattern is made up of a grammatical pattern and its distractor generation pattern. It should be noted that each test pattern needs to be able to generate questions with only one best key.

Having designed a number of test patterns, we employ the concept of “Web as

corpus” and hence gather data from the Web. We extract sentences  $S$  therein and are self-contained after removing HTML tags. We then tag and chunk these sentences (e.g. “*In biology, binary fission refers to the process whereby a prokaryote reproduces by cell division.*”). Part-Of-Speech (POS) tagging and phrase chunking accompanying the process of lemmatization provide adequate linguistic knowledge for constructing grammar tests (See Figure 7 for the example of the foregoing sentence using a tagger on the Brown corpus and a chunker trained on CoNLL2002 data). Having this knowledge, we apply appropriate test patterns  $P$  to a group of candidate sentences  $D$ , resulting in generating test items. The following is the detailed construction procedure:

**Input:**  $P$  = common patterns for grammar test items, URL = a Web site for gathering sentence

**Output:**  $Q$ , a set of grammar test items

1. Crawling the site URL for webpages
2. Clean up HTML tags. Get sentences  $S = \{S_1, S_2, \dots, S_n\}$  that are self-sufficient.
3. POS tagging and chunking sentences  $S$ .
4. Matching  $P = \{p_1, p_2, \dots, p_n\}$  against  $S$  to get a set of candidate sentence  $D = \{d_1, d_2, \dots, d_n\}$ .
5. Convert each sentence  $d$  in  $D$  into a grammar test item  $g$ .

Lemmatization: in biology, binary fission refer to the process whereby a prokaryote reproduce by cell division.

POS: in/in biology/nn ./, binary/jj fission/nn refer/vbz to/in the/at process/nn whereby/wrb a/at prokaryote/nns reproduce/nns by/in cell/nn division/nn ./.

Chunk: in /B-PP biology /B-NP ./O binary/B-ADJP fission/B-NP refer/B-VP to/B-PP the/B-NP process/I-NP whereby/B-ADVP a/B-NP prokaryote /I-NP reproduce/B-VP by/B-PP cell/B-ADJP division/B-NP ./O

Figure 7: Lemmatization, POS tagging and chunking of a sentence.

The formation strategies of multiple-choice and error identification question are different. In the following, we will separately state the generation process of these two questions.

The generation strategy of traditional multiple-choice question involves: (1) blanking the construct based on the structural pattern, (2) producing three incorrect choices according to the correspondent distractor generation pattern, and (3) randomly assigning options (e.g. A, B, C, D) to each alternative. We take the aforementioned

sentence as an example to test the construct of the agreement between noun and verb (See Figure 8).

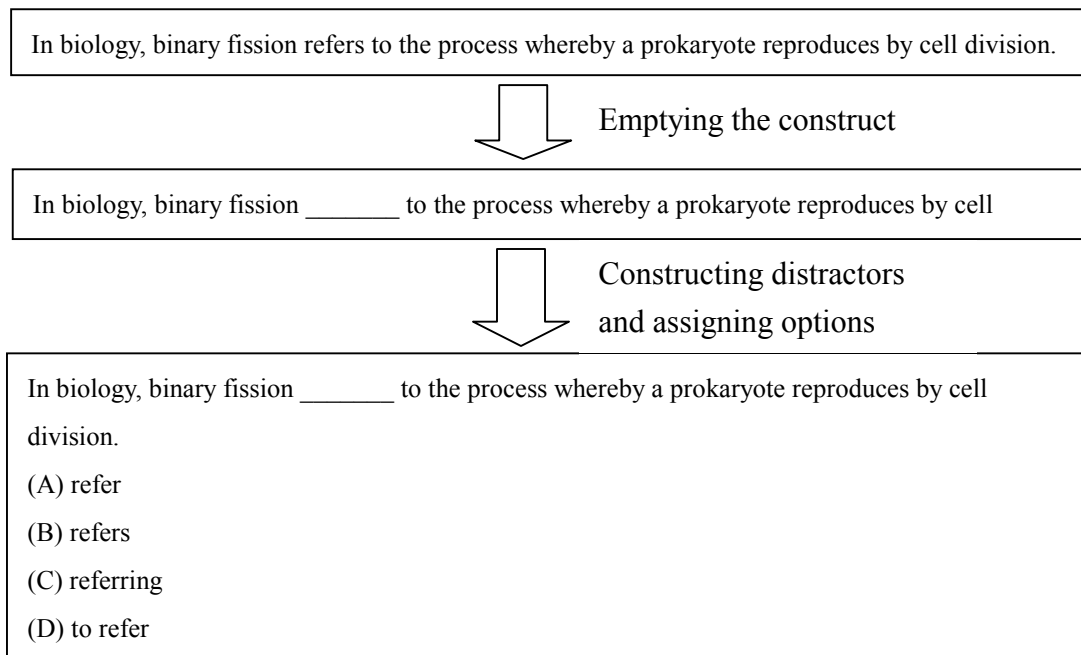


Figure 8: The generation strategy of the multiple-choice question.

As to error correction question, the test strategy includes: (1) locating the target point, (2) replacing the construct by selecting wrong items produced based on the distractor generation pattern, (3) grouping words of same chunk type to phrase chunk (e.g., “the/B-NP nickname/I-NP” becomes “the nickname/NP”) and randomly choosing three phrase chunk to act as distractors, and (4) assigning options on the basis of position. We still take the foregoing sentence as an illustration (See Figure 9).

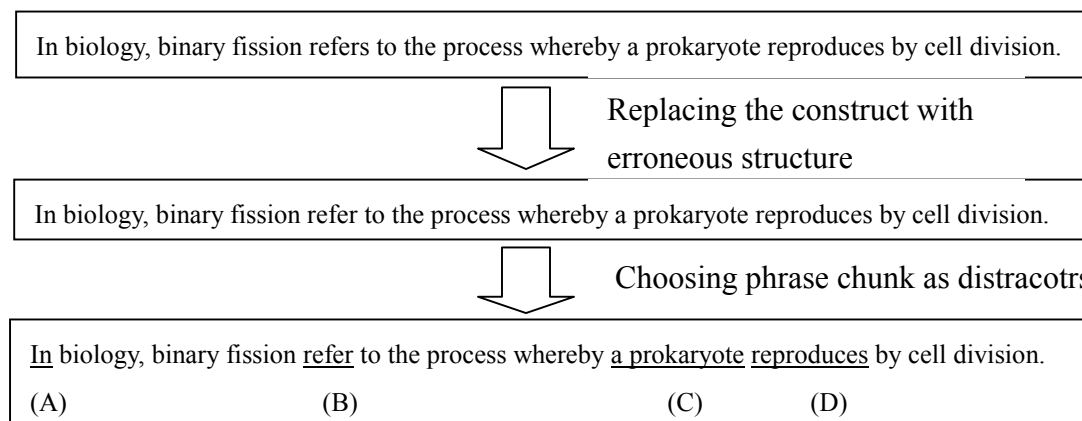


Figure 9: The generation strategy of the error detection question.

### Experiment Results and Evaluation

The introduced approach is aimed at automatically constructing structural quizzes of two formats, traditional multiple-choice and error identification. As such, this method will be validated over the feasibility of the generation with the help of the computer. Furthermore, we compare the efficiency of computer-generated questions in reference

to human-designed tests.

We adapted 68 organized grammatical rules (both are used by two test formats) as test patterns from the book “How to Prepare for the TOEFL” written by Sharpe. We then gathered a number of articles from two websites, Wikipedia and VOA (Voice of America). After concerning about the readability issue (Dale-Chall, 1984) and the self-contained characteristic of grammar questions, we extracted the first sentence of each article and accumulated these sentences based on the readability distribution of simulated TOEFL tests (See Figure 10 and Figure 11). At last, we collected 3,900 and 2,780 for multiple-choice and error detection questions, respectively.

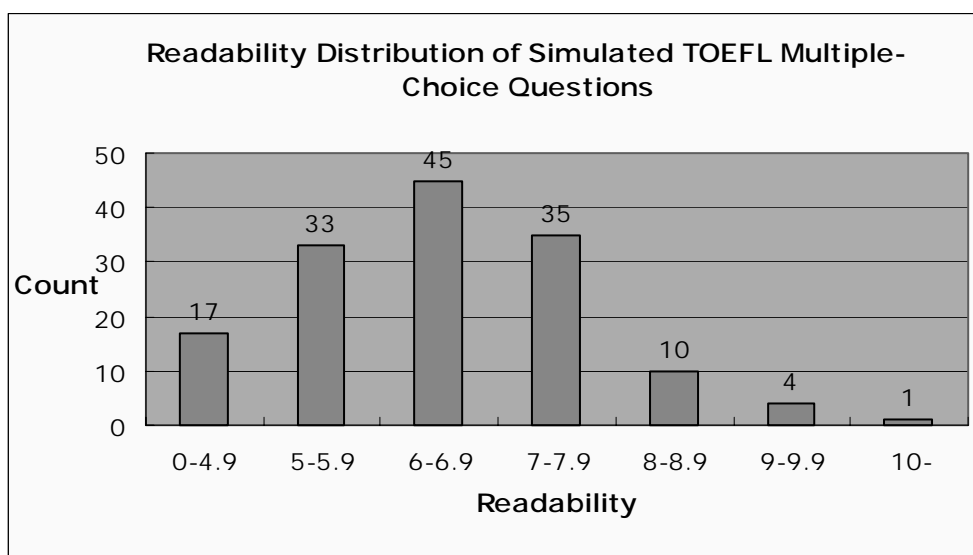


Figure 10: Readability distribution of simulated TOEFL multiple-choice questions.

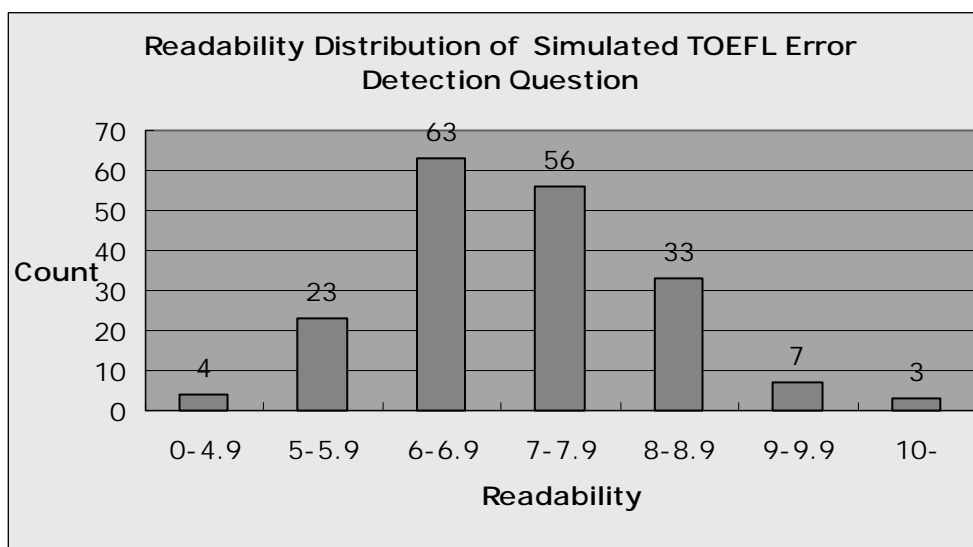


Figure 11: Readability distribution of simulated TOEFL error detection questions.

With the assistance of the computer, 25,906 traditional multiple-choice questions are constructed while 24,221 error correction questions are produced. The time needed to produce traditional multiple-choice test items is about 6 hours. As for the

production of error correction, the time needed for the computer is about 4 hours. This results in an average of approximately 51 seconds to compose an error correction question and 38 seconds to compose a traditional multiple-choice question by the computer.

In order to examine the quality of computer-generated question, we plan to demonstrate the following experiment. Questions generated by computer and human will be first evaluated by 7 people from TESOL. Questions with satisfactory quality will be selected to comprise a test administered to at least 30 people who have taken TOEFL in one year and 30 who have not. By calculating correlation with TOEFL and doing item analysis (item facility, item discrimination, and distractor efficiency), we will accurately assess the quality of algorithmically - composed questions. Besides, through analyzing test results from TOEFL unrelated group, we will understand whether test patterns constructed by Sharpe merely restrict to TOEFL test or are general so that can also be used in other examines.

### **Conclusion**

In this paper, we shed new light on automatically generating English structural tests. Our proposed method consists of several steps. First, we convert grammatical knowledge into test patterns. We then automatically collect authentic sentences from the Web as candidate question stems. Finally, different strategies are applied to two formats. At runtime, a given sentence conformed to structural patterns is generated into grammatical tests of two question types. Initial experimental results prove the facility and effectiveness of the introduced method and indicate that this novel approach paves a new way of CAIG.

### **Reference**

- Larsen-Freeman, Diane. Grammar and its teaching: challenging the myths. (1997)
- Kathleen, Kenji. (1996). Testing Grammar. *The Internet TESL Journal*, Vol. 2, No. 6
- Rau, D.H. (1999). Validity and Reliability of Grammar Proficiency Test and Evaluation of the Effectiveness of Freshman English Composition Class. *Providence Journal of Humanities*, 12, 155-184
- Purpura, J. E. (2004). *Assessing Grammar*. First Edition. 202-209 New York: Cambridge University Press.
- Wang, Shiuh & Hsien-chin Liou. (1991). An Item Analysis of an English Proficiency Test at Tsing Hua University: Grammar. Papers from the 7<sup>th</sup> on English Teaching and Learning in the Republic of China. 459-186. Taipei: Crane.
- Chang, Y.L. (2000). Analysis of a Doctoral Students' English Writing Proficiency Test at Chiao Tung University. Papers from the 9<sup>th</sup> International Symposium on English Teaching. 178-187. Taipei: Crane.
- Coniam, David. (1997) *A Preliminary Inquiry Into Using Corpus Word Frequency*

Data in the Automatic Generation of English Cloze Tests. *CALICO Journal*, No 2-4, pp. 15- 33.

Ruslan Mitkov, Le An Ha, Computer-Aided Generation of Multiple-Choice Tests. (2003)

Liu, Chao-Lin, Wang, Chun-Hung, Gao, Zhao-Ming, and Huang, Shang-Ming. Applications of Lexical Information for Algorithmically Composing Multiple-Choice Cloze Items, In Proceedings of the Second Workshop on Building Educational Applications Using NLP, pp. 1-8, Ann Arbor, Michigan, 2005.

Wolfe, J.H. (1976). Automatic Question Generation From Text-An Aid To Independent Study. *ACM SIGCUE Bulletin*, 2(1), 104-112.

Wilson, Eve. (1997). The Automatic Generation of CALL Exercises form General Corpora. In Wichmann et al. (eds.) *Teaching and Language Corpora*, pp. 116 – 130. Longman.

Chi-Chiang Shei. (2001). Follow You ! : An automatic Language Lesson Generation System. *Computer Assisted Language Learning*. Vol. 14, No. 2, pp. 129-144

Gao, Zhao-Ming. (2000). AWETS: An Automatic Web-Based English Testing System. In *Proceedings of the 8th Conference on Computers in Education/International Conference on Computer-Assisted Instruction ICCE/ICCAI, 2000, Vol. 1, pp. 28-634.*

Hoshino, Nakagawa. (2005). A real-time multiple-choice question generation for language testing-a preliminary study-. In Proceedings of the Second Workshop on Building Educational Applications Using NLP, pp. 1-8, Ann Arbor, Michigan, 2005.

Eiichiro SUMITA, Fumiaki SUGAYA, Seiichi Yamamoto. Measuring Non-native Speakers' Proficiency of English by Using a Test with Automatically-Generated Fill-in-the-Blank Questions. In Proceedings of the Second Workshop on Building Educational Applications Using NLP, pp. 61-68, Ann Arbor, Michigan, 2005.

Pamela J. Sharpe, "How to Prepare for the TOEFL" 11th ed., Barron's Educational Series, Inc. (2004)

王俊弘，劉昭麟，高照明。利用自然語言處理技術自動產生英文克漏詞試題之研究。(2004)

王俊弘，劉昭麟，高照明。電腦輔助英文字彙出題系統之研究( Toward Computer Assisted Item Generation for English Vocabulary Tests ) (2002)

黃上銘，劉昭麟，高照明。適性化線上英語聽寫測驗系統之研究( Toward Computer Assisted Learning for English Dictation ) (2003)