Journal of English Language
Teaching and Learning
No. 9,2012

Design and Implementation of an Intelligent Part of Speech Generator

Vahid Reza Mirzaeian

Assistant Professor, Arak Industrial University

Abstract

The aim of this paper is to report on an attempt to design and implement an intelligent system capable of generating the correct part of speech for a given sentence while the sentence is totally new to the system and not stored in any database available to the system. It follows the same steps a normal individual does to provide the correct parts of speech using a natural language processor. It uses both morphological and syntactic analysis of the input to arrive at correct part of speech. It, then, checks to see if the correct part of speech is provided. If not, it displays the correct part of speech with a short note referring to the specific rule responsible for the selection of correct part of speech. This tool can be used to help learners master English parts of speech system.

Keywords: part of speech, word class, lexical class, lexical category, ICALL, morphology, syntax, NLP.

– تاریخ وصول: 1391/4/20، تأیید نهایی: 1391/7/15

Introduction

Developing an ICALL (Intelligent CALL) system follows a completely different direction. We are nearing the end of an era in which language training software is confined to let us say a list of vocabulary items to be taught. Due to advances in computing industry, both teacher and students expect more form the language teaching software. They prefer wide range of coverage instead of providing a simple message, saying the answer is correct or incorrect to a limited number of sentences. This is also in line with advances in learner type saying training should be geared to the student's need, desire and his mental capacity. For instance, a typical vocabulary software nowadays should give personalized settings designed for variety of users. The user should have freedom to choose words he likes to learn and also the level of difficulty should be based on student's mental capacity.

One of the promising areas in this field is the advances made in Natural Language Processing (NLP). NLP components should be incorporated into language teaching software. So it is the inclusion of this NLP technology (intelligent factor) which changes the CALL application into an ICALL one. Such applications that are called intelligent have, although relatively rare, already entered the scene of computer assisted language learning (Wood 2008, Heift and Shulze 2007, Amaral and Meurers 2008). However, the recent ICALL workshop during CALICO 2008 conference indicates that there is a lot to be done in this area.

As demands for more versatile language teaching applications increase, so does the job of software developers producing them. Developing NLP engines capable of parsing numerous unpredicted natural sentences is highly difficult. To improve the quality of task-based synchronous computer-mediated communication (CMC) (Dickinson et al. 2008) designed a parser-based system that provided feedback on particle usage for first-year L2 Korean learners while they chatted in CMC. Both to facilitate the use of a new orthography by beginning learners and to make processing feasible for the ICALL, they guided the content of the activity by using picture-based information-gap tasks and a game record. A word bank was also used to control the range of allowable learner input. In another similar project, (Chang et al. 2008) developed an online collocation aid for EFL writers in Taiwan, aiming at detecting and correcting learners' miscollocations attributable to L1 interference. Their system used natural language processing techniques to segment sentences in order to extract V-N collocations in given texts, and to derive a list of candidate English verbs that shared the same Chinese translations via consulting electronic bilingual dictionaries. The system with the aid of a reference corpus was able to pinpoint the miscollocations and provide the learner with adequate collocations that the learner intended to write but misused.

Providing intelligent feedback on learners' errors has attracted considerable interest from both language teachers in general and second language acquisition (SLA) researchers in particular. Computer-assisted language learning also provides ample opportunities to test the feasibility and usefulness of feedbacks given to students. However, the students are limited in the range of possible responses the may offer to a language teaching software. Generally speaking, the more freedom is given to the student, the more complex the software will turn out to be. A key solution to this problem is the inclusion of a natural language processing (NLP) component (Heift and Shulze 2007).

Due to their flexibility in parsing any input given by the students, ICALL systems are very promising in dealing with a wide range of input not stored exactly in any database available to them. The NLP component can provide detailed linguistic analysis of the student input and suggest possible corrections. Faced with a non-native input full of mistakes in all the areas of language, these systems are able to analyze the input and provide insight on the nature of the error to both teachers and students (Heift and Nicholson 2001). This paper reports on an Intelligent CALL system capable of providing the correct part of speech by the use of an NLP engine.

Related work

"Natural language parsers take written language as their input and produce a formal representation of the syntactic and sometimes semantic structure of this input." (Schulze 2001) Lack of progress in some areas of CALL such as ICALL can be attributed to the lack of linguistic modelling and insufficient deployment of natural language processing techniques (Levy 1997). This is true since there are few CALL applications incorporating NLP techniques (Schulze 2001).

In a conference held in 1997 whose proceedings appeared in (Nerbonne, Jager and van Essen 1998), (Carson-Berndson 1998), (Witt and Young 1998) and (Skrelin and Volskaja 1998) reported on three successful applications in the field of pronunciation (phonology). GLOSSER and COPERNICUS were two projects upon which a number of papers were based in the conference demonstrating the use of language processing tools in the field of CALL. Other CALL projects in the conference were RECALL, a "knowledge-based error correction application" (Murphy, Kruger and Griesz 1998) and the development of a tool for learning Basque as a foreign language (Diaz de Ilaranza, Maritxalar and Oronze 1998). The latter project relied on a spell-checker, morphological analyzer, syntactic parser and a lexical database for Basque and the authors reported on the development of an interlanguage model.

In 1999, a conference was held at UMIST, Manchester, investigating the role of NLP in CALL. The proceedings were published in 1999 as a special issue of ReCALL. (Tschichold 1999) discussed strategies to improve the success rate of grammar checkers; (Menzel and Schroder 1999) described error diagnosis in a multi-level representation; (Visser 1999) introduced CALLex, a program for learning lexical functions; (Ilarraza et al. 1999) described aspects of IDAZkide, a learning environment for Spanish learners of Basque. (Foucou and Kubler 1999) presented a web-based environment for teaching technical English to students of computing. Exercises were based on a corpus compiled from a number of help pages, frequently asked questions (FAQ), etc.

(Heift and Nicholson 2000), implemented a parser-based CALL system for German using Java applets interacting with a web server which preprocessed Java/HTML input. The system, using HPSG (Head-driven Phrase Structure Grammar) as its underlying grammatical formalism, differed from the traditional approach by permitting the grammar to freely generate as many parses as it could and using separate pedagogic principles to select the appropriate interpretation and response. The system tightly integrated the student model into the process of selecting the appropriate interpretation and generating a response tailored to the students' level of expertise.

A Greek system using a similar architecture and language processing approach also relied on buggy rules -- rules that describe grammatically ill-formed input. These rules were triggered when the rules for well-formed input failed. (Heift and Nicholson 2000)

(Chen, Tokuda and Xiao 2002) developed an effective learner model for a template-structured ICALL system for Japanese-English Writing skills which was reinforced by an efficient matching algorithm and a POST parser. Later, as a step towards implementing a human language teacher, they developed a template-based on-line ICALL system capable of automatically diagnosing learners' free-format translated inputs and returning error contingent feedback.

(Dansuwan et al. 2001) also designed a Thai learning system to help learners acquire Thai word order system. The system facilitated the lessons on the web using HTML and Perl programming which interfaced with Natural Language Processing (NLP) by means of Prolog. The system introduced an easily understandable presentation of a sentence structure by indicating syntactic trees and case grammar principles. The results of the evaluation showed that the system had an effective user interface and handled learner's input efficiently.

(Shaalan 2005) described the development of an intelligent computer-assisted language learning system for learning Arabic. His system could be used both by students of Arabic at primary schools and those learning it as a second or foreign language. It used NLP techniques to parse Arabic language. To make benefit from ICALL, (Dickinson et al. 2008) designed a parser-based system that provided feedback on particle usage of first-year L2 Koeran learners. The material also included activities using picture-based information-gap tasks and game records to control the range of allowable learner input by using a word bank.

This list of recent examples of CALL applications that make use of parsing techniques is by no means exhaustive. These examples not only illustrate that research in parser-based CALL is vibrant, but also that parsing techniques have an important contribution to make in the further development of CALL.

System Architecture

As Figure 1 illustrates, the user enters a sentence containing a blank space into the system. Next, the student is prompted to provide a vocabulary item for the blank space. They system checks the item and if it is acceptable for the blank space, it will provide the same item with a message saying the given word is correct. However, if the given word is not a correct word form (adjective, adverb, verb or noun), the system provides the correct word form with a rule describing why the correct word form has been selected.

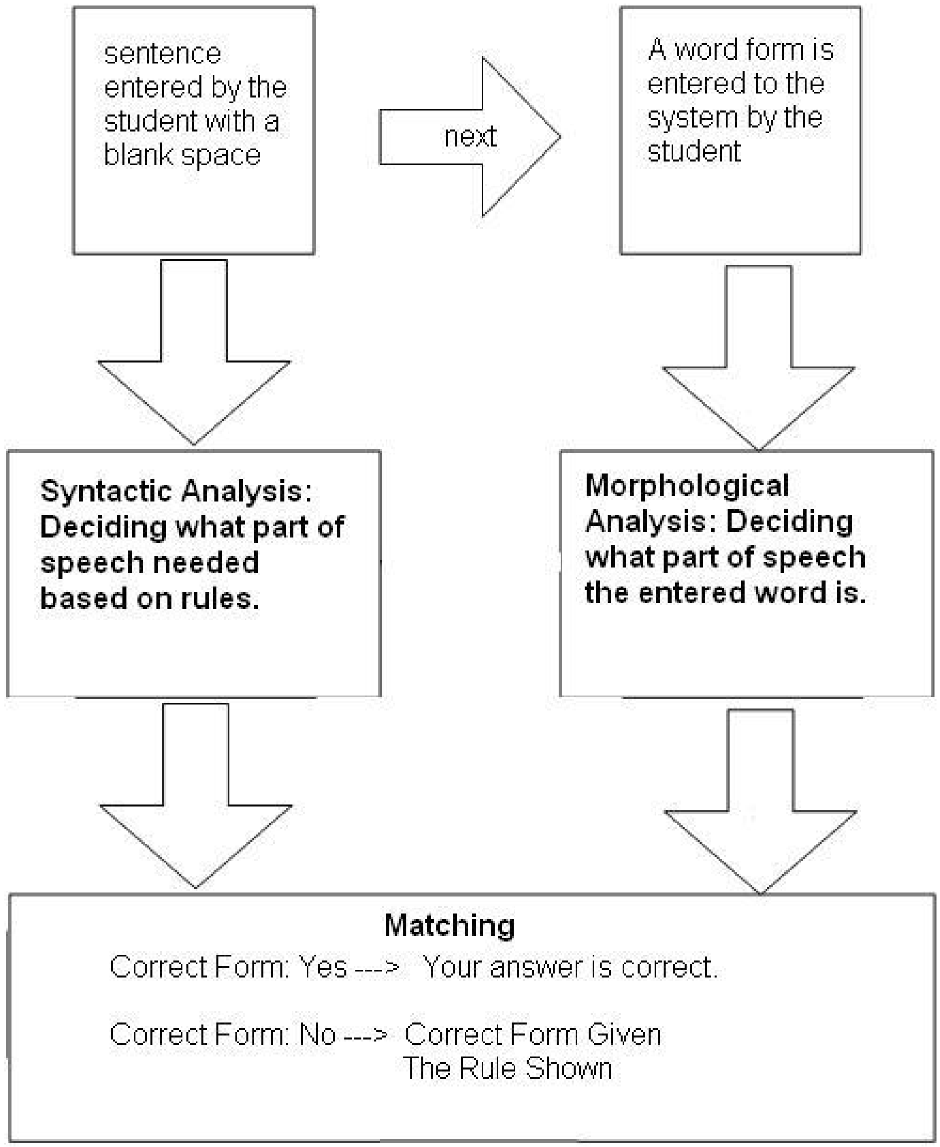

Figure 1: Our Model

The System's Grammatical Formalism

The choice of grammatical framework in ICALL has important implications for both research and development . ICALL applications have used a large variety of grammatical frameworks in different projects over the last 30 years (Heift and Shulze 2007). The term grammatical framework has been used a generic term covering different formal grammars. Such concentration on formal grammar is necessary since only a mathematical approach can facilitate an implementation of this grammar in ICALL systems (Schulze 1998). One of the major works devoted to this issue is a paper by (Matthews 1993) in which he lists eight grammar frameworks used until then in ICALL. He realizes the importance of the grammatical framework for an ICALL system because of a crucial decision for the facilitation of CALL and SLA research. He argues for 'choosing a formalism that potentially meshes with SLA' (Matthews 1993). He sketches three criteria (pp. 9–13) that ought to inform the decision of selecting a grammatical framework for an ICALL system:

- **computational effectiveness**: describing the criterion of successful computational implementation of the grammatical framework, which needs to be associated with a formalism and an efficient parsing algorithm;

- **linguistic perspicuity**: referring to the necessary descriptive power of a grammatical framework, a descriptive power which needs to go beyond the adequate description of a few selected linguistic phenomena;

- **acquisitional perspicuity**: capturing another role of grammar frameworks namely their contribution to the explanation of the acquisition and the development of L2.

Since our system had to have both a morphological analyzer to determine and provide the correct part of speech and a syntactanalyzer to determine which part of speech is needed for a given sentence, experiments with different grammatical formalisms were performed (Mirzaeian 2008). At the end, a combination of CM (Categorial Morphology) for morphological analysis and HPSG, for syntactic analysis was adopted.

Part of Speech (Word class)

According to the Encyclopedia of Language and Linguistics, the word-class system (or part of speech) of a language is a classification of the individual elements that make it up, viz., words, in terms of properties that determine their use in syntax and their contribution to meaning. (Anward 2006) More specifically, it refers to a linguistic category of words, defined by syntactic and morphological behavior of any lexical item. In addition, there are open word classes constantly acquiring new members through different linguistic processes, and closed word classes, which as the name suggests, occasionally, if ever, acquire new members. The first group includes mainly nouns, adjectives, adverbs and verbs, and the latter includes everything except the four types mentioned.

Although traditional grammarians have divided English words into main eight categories namely, nouns, pronouns, adjectives, verbs, adverbs, prepositions, conjunctions and interjections, based on function, more specific categories and subcategories have been introduced by modern linguists. In language teaching, part of speech exercises, the main concern of this article, are exercises in which students are asked to provide correct part of speech of a given word mainly limited to the open word class i.e. nouns, adjectives, adverbs and verbs.

In books published for Persian students of English, exercises of this sort are usually provided in three forms as shown in Figure 2. (Mirzaeian 2005, Mirzaeian 2007, Mirzaeian 2009)

Type 1: (from Mirzaeian, 2005)

Fill in the blanks with the correct word forms given:

computation, computational, computerize, computer

A. The .... requirements necessary to produce the payroll for a large company take a very long time.

B. People in banking industry try to .... all aspects of banking

C. It is a fact that humans cannot perform mathematical operations as fast as ....

D. John was graduated from the department of ....

Type 2: *(from Mirzaeian, 2007)*

Word Formation:

Verb	Noun	Adjective	Adverb
Account	accountant/accounting/account	Accountable	Accountably
Allocate	Allocation	Allocated	-----
Apply	Application	applicable	Applicably
Provide	Provision	Provided	Provisionally

Now use the correct form of the words given above to complete the following sentences.

1. Reza's brother is a(n) .... in our company who keeps and examines business records.

2. The bank sends me a monthly record of my ....

3. This year the government has .... a good part of the budget to the improvement of local industries in the country.

4. It seemed quite unwise to .... so much money to buying machinery which produced old-fashioned goods.

5. The .... of the government's budget to the development of industries is something that most citizens appreciate.

6. Scientific discoveries are often .... to industrial processes.

7. The .... of atomic energy to manufacturing has resulted in fast producing of complicated tools.

8. The new formula of psychological behaviour is not .... to our classroom setting.

9. The manager agreed to give the workers a raise .... that they promise to work harder.

10. The success of a company depends on the correct use of both financial and human ....

Type 3: *(from Mirzaeian, 2009)*

Use the correct form of the word *'inform'* in the blank spaces.

a. I read a very .... article about operating systems.

b. The article had lots of interesting ....

c. The news websites always .... me about world events.

d. The author of the article wrote .... about operating systems.

Figure 2: Various Part of Speech Exercises

As we can see, in the first type, different word forms are given and the students are asked to select the correct form for the blank space. In the second type, the students are asked to choose the correct form and put it in the blank space. What makes this exercise different from the previous type is the number of similar items. For instance, when we decide to use a noun, we have to choose between similar nouns. This combines the part of speech knowledge with vocabulary knowledge. Our system cannot handle this second type since it does not have a semantic analyzer to distinguish between different nouns for a given sentence. In the third type, students are asked to provide the correct form on their own and put it in the blank space.

It is clear that the first type is the simplest one since all students have to do is to select the correct form (with the exception of changing singular forms into plural and using the correct form of the verb). In the second type, in addition to the above point, meaning is also involved since the correct word in addition to its meaning has to be provided. In third type, students should provide the correct word form on their own. It means that they should already know that the noun for the verb inform is information.

Our System

Two approaches can be taken to design a system capable of providing the correct word form for the above exercises.

Using a Database

In this case, all the POS exercises of a given book are stored in the database together with the correct answer. The system has to check the database, match the sentence and provide the correct word form blindly. The advantage of this system is that it can be easily designed and implemented. However, the main disadvantage is that it is incapable of providing correct answer to sentences not stored in the database. As a result, it can only be used with reference to a particular textbook.

Using a natural language processing engine

In this case, we are using an NLP engine which tries to either provide the correct answer or to check the correctness of the provided answer although it does not have access to a sentence database. It means that the sentence with the blank space and the correct word form are new to the system. It tries to solve the problem using an NLP engine. This is the aim of this paper. It shows how we are going to accomplish this task. Our aim here is to create a system capable of solving all the above types of exercises intelligently using the immense power of a natural language processing engine.

How does an individual solve POS exercises?

To enable a computer system to solve these types of exercises, one has to observe closely how a human being solves these exercises using his/her intelligence. First, let's have a look at first type as shown in Figure 2. At the beginning, a student has to identify that in Figure 2, the word computation is a noun, and the word computational is an adjective. This is called morphological analysis.

Morphological Analysis

To computationally implement morphological analysis, we have two choices. The first strategy to adopt is to store all word forms in a database. For instance we can store the word *inform* as noun, *information* as a verb etc. We should also store singular and plural forms as well as different verb forms. This makes our database huge since every word form has to enter the database. The second strategy is to store the words and affixes separately. This is the approach we have adopted.

In ourapproach, our lexicon (the dictionary file to be specific) contains a set of words and affixes, with the process of combining the two mediated by a set of morphotactic rules based on categorial morphology which operate in much the same way as the classical 2-level rules of (Koskiennemi 1985). All the necessary information is included in the definition of the words and affixes. For instance, the word *inform*normally subcategorize for different kinds of affixes such as *–ation*, -*ative –atively* etc. These affixes have to appear in a given order. The stem has a subcategorization requirement which contains the feature structure of the affix the stem needs in order to form sublexical signs of the next higher order. This sublexical sign in turn subcategorizes for an affix of a certain kind. This process continues until the sign requires no more affixes and thus is morphologically saturated.

Our system normally relies on the interaction of two distinct sets of information: firstly, a dictionary storing information about lexical signs, and secondly, a set of rules and principles governing the possible combinations of lexical signs available. It will be more efficient to store morphemes in the dictionary and extend the set of governing rules and principles in such a way that they cover not only the possible combinations of lexical signs within sentences, but also the combination of smaller linguistic units to create lexical signs. Although this approach requires an extended set of grammatical rules because morphological rules have to be added to the syntactic rules, it has advantages in at least two respects: the dictionary will be more concise and consequently more easily maintainable, and it can be shown the grammatical information lexical signs have is rule-governed to a large extent. It has to be noted that the formation of morphological structures can be described with a few categorial rules which the system uses. We store the word *inform* as an entry in our system (Figure 3). We also store each *suffix* separately (Figure 4). Using morphological rules, (Figure 5) the system can combine *inform*with *-ation* to make the noun *information*. The system can also generate the plural nouns from singular nouns as well as different verb forms if the need arises. Figure 6 shows the entry for the plural maker *'s'* in English. For a list of suffixes gathered for this study, please refer to appendix 1.

```
"inform" $$ X lextype verb(d) delayed vtype(X, valency(2, [agent, v])) :-

verb(X).
```

Figure 3: The entry for the word 'inform'

```
"ation" $$ X :-

affix@X<-> *1,

    X <> [noun, suffix, common_noun],

affixes@X<-> [AGR],

subject@X<> nom,
```

```
dir@AGR<>xafter,

affix@AGR<-> *agr,

[syntax, lex_type]@AGR <-> [syntax, lex_type]@X.
```

Figure 4: The entry for suffix '-ation'

```
needsFirstAffix(X) :-

affixes@X<-> [A1],

    affix@A1 <-> *1,

    [lex_type, branch, syntax, uses]@X

<-> [lex_type, branch, syntax, uses]@A1.
```

Figure 5: A typical morphological rule

```
"s" $$ X :-

affix@X<-> *1,

    X <> [noun, suffix, plural].
```

Figure 6: The entry for plural maker '-s'

Part of Speech Selection

In order to see how students go about providing the correct part of speech, we started observing some teachers when teaching students how to provide the correct part of speech. Our findings were summarized into two sections namely 'Helps for Word Studies' shown in Appendix 2 and 'Patterns Found in Word Studies' shown in Appendix 3. Based on these observations, we summarized the information and converted them to specific rules so that they could be computationally implemented.

Let us look at an example. To identify the correct part of speech for the sentence, "*The .... requirements necessary to produce the payroll for a large company take a very long time*," we should use the word or words preceding the blank space to determine which POS to use. To provide the correct part of speech, one has to use the rule: "Det + (Adj) + Noun" to get to the correct answer.

We have studied all the POS exercises in the three books mentioned in Figures 2 and we have devised 50 rules covering 90 percent of all POS exercises. The rule mentioned in the previous paragraph covered 35 percent of the cases in our data.

Implementation of rules within the system

We use Steedman's slash notation (Steedman 1985) for the description of the syntactic rules governing the grammatical behavior in our current system. For the treatment of syntax here, we need Steedman's forward composition rule.(Pollard 1988, McGee Wood 1993)) The association rules (Figure 7) are necessary in order to conclude syntactic processes, to find the correct category (part of speech) for a given sentence:

1	X / YY→X
2	XY\X→Y
3	X / Y Y/Z→X/Z
4	Y\ZX\Y→X\Z

Figure 7: Association rules to find the correct part of speech

A few general remarks regarding the notation employed in Figure 7 are necessary. The correct part of speech is defined by as:

X is the word we need if:

1. X is a member of the set of primitive categories ; or

2. X is of the form V/W, where V and W are the categories; or

3. X is of the form V\W, where V and W are the categories.

(Hoeksma 1985)

The use of the slash and backslash has to be defined explicitly here because they are used in different ways in the literature. According to (Hoeksma 1985), V/W denotes that V is missing a W to its right, i.e. V will be a proper V once it has found a W. However, in the literature, V\W denotes a V that is missing a W to its left, i.e., it is the same again, V Will be a proper V once it has found a W – just that this W is now expected on the left. This notation has the clear advantage that denominator and numerator do not change positions in these two operations as they do in Hoeksma's notation. As a result, this will be the understanding of left and right cancellation:

RightCanc(V/W W) → V

LeftCanc(W V\W → V

Based on the above description, our system checks the word before the blank space and tries to figure out what part of speech it needs. It the information is found in the rules is enough for the system to decide, the correct part of speech is selected, if not, it moves to the next word before the blank or after the blank until the correct part of speech is selected.

Based on what was said, our system is now capable of providing the correct part of speech for the exercises described in Figure 2.

Handling exceptions

Exceptions are one of the problematic areas in computational linguistics. Among our data we came across the following sentence, "I will do the job ………that you help me." We are required to choose the correct word form for *provide* in the blank space.

As we can see, the correct form needed here is *provided* but it does not follow any of our rules. This is because *provided that* as a whole is a subordinator which can be used instead of *if*. These exceptions should be added to the system. Inclusion of these exceptions is time consuming, but the robustness of any natural language application depends on the amount of time and energy spent on the development of that system.

The system's interface

Our system written in Prolog does not enjoy a user-friendly interface. So we had to develop an interface which could make the system useful for both students and teachers. As (Schulze 2008) puts it, ‘interface’ is an established technical term in domains such as computer science (e.g., the interface of the computer to a monitor or projector), software engineering (e.g., the application programming interfaces of an operating system which support requests by other computer programs), and human computer interaction (the interfaces between user and machine). However, here it only refers to the actual bridge between the user and the system to test whether or not the current system could be easily used by the learners and/or teachers. Although, our system can be executed via a command-line interface from within Prolog window, this interface was created using Java to show that it could be executed from a program that uses a graphical user interface (Figure 8). Java and Prolog can also be installed on a web server so that they could be accessed via the web.

Figure 8: The graphical user interface

As you can see, the user enters the sentence and a word form. By clicking on the **Check** button, the system first analyzes the word morphologically, checks the sentence based on the rules to determine the part of speech need. Next, it matches this syntactic information with the morphological information to see if the correct word form is entered. If the answer is correct, a window (Figure 9) is displayed.

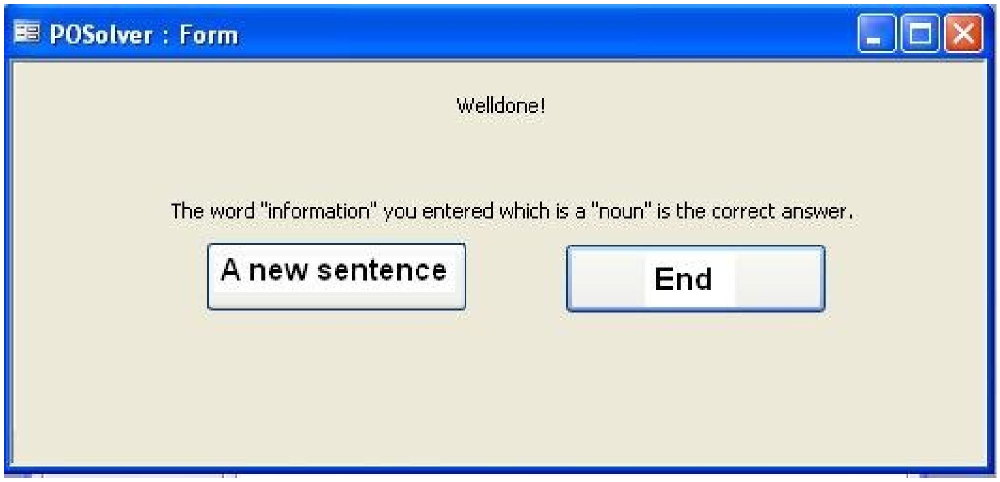

Figure 9: System accepting the correct POS

If the answer is incorrect, the correct answer and the applicable rule is shown (Figure 10).

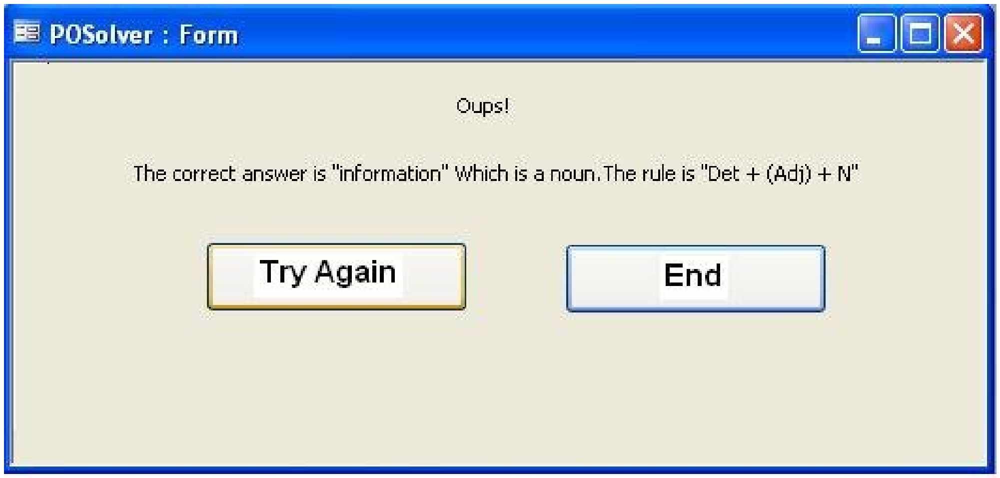

Figure 10: System displaying correct POS and the rule

Conclusion

Natural Language Processing is an invaluable tool currently used in the field of computer science especially artificial intelligence. However, it can be quite useful in CALL applications to extend the range of inputs given by the students. A parser capable of processing words morphologically and sentences syntactically was employed and it was adopted to check the correctness of the input. It could also provide the correct word form. It was good if detailed explanation of the applied rule for a given sentence comprehensible to the students would be generated. Then, the system could be tested with various types of students to measure its effectiveness.

It also has to be noted that the system described here can only handle exercise types one and three. The system cannot handle part two since it has to decide which noun to be used among the three nouns given. This leads to semantics which is very difficult to implement computationally. Experiments are being performed but the findings are not mature enough to be reported. Inclusion of NLP components with wide range of coverage is time-consuming and costly. It is hoped that with the fast developments in the field of computers in general and NLP in particular, we witness a growing use of this useful tool in the language teaching software development.

Appendix 1:*Common suffixes found in our data*

Noun Making	**Adjective Making**	**Verb Making**	**Adverb Making**
-age	-able	-ate	-fold
-al	-ible	-en	-ily
-ance	-ate	-ify	-ly
-ence	-ish	-fy	
-ary	-y	-ize	
-dom	-al	-ise	
-ee	-ary	x	
-hood	-like	x	
-ure	-less	x	
-ment	-ful	x	
-ation	-ive		
-ion	-ent		
-ition	-ant		
-tion	-ative		
-xion	-ive		
-sion	-an		
-er	-ian		
-or	-en		
-ness	-ful		
-ity	-ic		
-ism	-ical		
-y	-ious		
-ive	-ous		
-ist	-ese		
-ness	-ward		
-ship			

(the only exception in our data is the prefix *en-* as in enlarge changing an adjective to a verb)

Appendix 2: Help for Word Studies

General Procedure:

1. Read the whole sentence
2. Do not look up every word you do not know.
3. Decide what the word before and after the blank are. Write N, V, etc above them.
4. If you have difficulties, look at the patterns. Most of the patterns you need are listed.
5. Notice that the N, V, Adj and Adv forms of a word do not always have the same meaning.

Kinds of Nouns

Non-count: satisfaction

$$Count\begin{Bmatrix} \sin gular : life \\ Plural : lives \end{Bmatrix}$$

Use of Articles

$$a(n) + Ncount \sin g : life$$

$$the + \begin{Bmatrix} Ncount \sin g : life \\ Ncountplural : lives \\ Nnon-count : satisfaction \end{Bmatrix}$$

$$(none)\begin{Bmatrix} NcountPlural : lives \\ Nnon-count : satisfaction \end{Bmatrix}$$

Prepositions

1. Prepositions are followed by an N with its describing words

For peace

For his short stories

2. Prepositions are also used

a) as adverbs alone: *He came here before*.

b) to start a clause (clause=N+V): *He came here before I did*.

3. Common prepositions to look for:

About, above, according to, across, after, against, along, around, at, before, behind, below, beside, between, by, down, during, except, for, from, from under, in, in spite of, into, off, of, on, out of, outside, over, since, through, to, under, up, upon, with.

Adjectives

1. In the pattern $\left\{\begin{matrix} a \\ the \end{matrix}\right\}$+Adj+N sometimes, the Adj is in the comparative or superlative:

TheAdjN

The happiest boy

The most beautiful city

2. Remember that past participle and present participle forms of verbs are used as adjectives:

Developing countries

A settled area

Not

Not is not included in the patterns. It can be used in almost any sentence.

Question Word Order

There are no examples of question word order in the patterns. But usually the patterns can be used for questions anyway.

What are the major differences?

The +Adj+N(pattern 4(b) or 5(b))

Sometimes it is easier to change the question into a statement.

What are you majoring in? You are majoring in what?

N + V+ Prep Phrase

Omission of words

Sometimes words are omitted in English sentences:

It's possible , but (it is) not probable.

(Pattern 5g) (BE) + Adj

Symbols used in patterns

( ) = optional (may be omitted)

{} = choose one item at a time

$$N = \left\{ \begin{array}{l} noun \\ pronoun \\ noun + describingWordsBeforeAndAfterit \\ objectOfVerb(sometimesIsAGroupOfWords \\ withaS + V) \end{array} \right\}$$

Appendix 3: 'Patterns found in Word Studies'

1. Subjects and Predicates (predicate=V + modifiers, object)

a. N+Vi

man cannot live

b. N + Vt + N (or other object)

I did not recognize him.

c. N + be + V-ed (passive)

Nation has not been recognized.

d. N + V + to V

doctors are trying to conquer

e. N + V + N + to V

mother has hung out the clothes to dry.

f. N + V + N + Adj

You find part desirable.

2. Adverbs which divide verbs

a. Modal + Adv + V

will probably be

b. have/had + adv + V-ed

has carefully explained

c. be + Adv + V-ing

is always emphasizing

3. Adverbs of Manner

a. V + Adv /m

spoke emphatically

b. V + N + Adv/m

spends time productively

c. N + Adv/m + V

I easily finished.

d. Adv/m + N + V

naturally I miss

4. Kinds of Noun Phrases

$$(\Pr ep) + \left(\begin{Bmatrix} a(n) \\ the \\ --- \end{Bmatrix} \right) + \begin{Bmatrix} a)N \\ emphasis \\ b)Adj + N \\ man's Life \\ c)Adj + Adj + N \\ different Racial Groups \\ d)Adv + Adj + N \\ usually Calm Mann \\ e)N + N \\ Population Explosion \end{Bmatrix}$$

5. Following Be

$$BE + \left(\left\{ \begin{array}{l} a(n) \\ the \\ --- \end{array} \right\} \right) + \left\{ \begin{array}{l} a) N \\ Certa\text{int}\, y \\ b) Adj + N \\ PeacefulSolution \\ c) Adj + Adj + N \\ ManyNewSettlements \\ d) Adv + Adj + N \\ VeryDevotedSon \\ e) Adj + N + N \\ ShortBusRide \\ f) N + Adv / pl - t \\ sameOriginally \\ g) Adj \\ \text{Re}\, cognizable \\ h) Adj + Adv / m - t \\ undevelopedIndustrially \\ i) Adj + toV \\ difficultToDistinguish \\ j) Adv + Adj \\ recognizablySimilar \end{array} \right\}$$

References

Amaral, L. & D. Meurers (2008) From recording linguistic competence to supporting inferences about language acquisition in context. *Computer Assisted Language Learning,* 21**,** 323-338.

Anward, J. 2006. Word Classes/Parts of Speech: Overview. In *Encyclopedia of Language & Linguistics,* ed. B. Keith, 628-632. Oxford: Elsevier.

Carson-Berndson, J. 1998. Computational Autosegmental Phonology in Pronunciation Teaching. In *Language Teaching and Language Technology,* eds. J. Nerbonne, S. Jager & A. van Essen, 11-20. Lisse: Swets & Zeitlinger.

Chang, Y.-C., J. S. Chang, H.-J. Chen & H.-C. Liou (2008) An automatic collocation writing assistant for Taiwanese EFL learners: A case of corpus-based NLP technology. *Computer Assisted Language Learning,* 21**,** 283 - 299.

Chen, L., N. Tokuda & D. Xiao (2002) A POST Parser-Based Learner Model for Template-Based ICALL for Japanese-English Writing Skills. *Computer Assisted Language Learning,* 15**,** 357-372.

Dansuwan, S., K. Nishina, K. Akahori & Y. Shimizu (2001) Development and Evaluation of a Thai Learning System on the Web Using Natural Language Processing. *CALICO,* 19**,** 67-88.

Diaz de Ilaranza, A., M. Maritxalar & M. Oronze. 1998. Reusability of Language Technology in Support of Corpus Studies in an ICALL Environment. In *Language Teaching and Language*

Technology, eds. J. Nerbonne, S. Jager & A. van Essen, 149-166. Lisse: Swets & Zeitlinger.

Dickinson, M., S. Eom, Y. Kang, C. M. Lee & R. Sachs (2008) A balancing act: how can intelligent computer-generated feedback be provided in learner-to-learner interactions? *Computer Assisted Language Learning,* 21**,** 369-382.

Foucou, P.-Y. & N. Kubler (1999) A Web-Based Language Learning Environment: General Architecture. *ReCALL,* 11**,** 31-39.

Heift, T. & D. Nicholson (2000) Theoretical and practical considerations for Web-based Intelligent Language Tutoring Systems. *Intelligent Tutoring Systems, Proceedings,* 1839**,** 354-362.

--- (2001) Web Delivery of Adaptive and Interactive Language Tutoring. *International Journal of Artificial Intelligence in Education,* 12**,** 310-325.

Heift, T. & M. Shulze. 2007. *Errors and Intelligence in Computer-Assisted Language Learning:Parsers and Pedagogues*. Routledge.

Hoeksma, J. 1985. *Categorial Morphology*.New York: Garland Publishing.

Ilarraza, A. D. d., A. Maritxalar, M. Maritxalar & M. Oronze (1999) IDAZKIDE: an Intelligent CALL Environment for Second Language Acquisition. *ReCALL,* 11**,** 12-19.

Koskiennemi, K. 1985. A General Two-Level Computational Model for Word-Form Recognition and Production. In *COLING-84*, 178-181.

Levy, M. 1997. *Computer-Assisted Language Learning. Context and Conceptualization*.Clarendon: Oxford.

Matthews, C. (1993) Grammar Frameworks in Intelligent CALL. *CALICO,* 11**,** 5-27.

McGee Wood, M. 1993. *Categorial Grammar*.London: Routledge.

Menzel, W. & I. Schroder (1999) Error Diagnosis for Language Learning Systems. *ReCALL,* 11**,** 20-30.

Mirzaeian, V. R. 2005. *Beneral English for University Students*.Arak: Fath Danesh Publications.

---. 2007. *English for the Students of IT*.Tehran: Samt.

---. 2008. *An Introduction to Prolog and Persian Language Processing*.Tehran: Iran University of Science and Technology.

---. 2009. *English for the students of Management*.Tehran: Iran University of Science and Technology Press.

Murphy, M., A. Kruger & Griesz. 1998. RECALL -- Towards a Knowledge-Based Approach to CALL. In *Language Teaching and Language Technology,* eds. J. Nerbonne, S. Jager & A. van Essen, 62-73. Lisse: Swets & Zeitlinger.

Nerbonne, J., S. Jager & A. van Essen. 1998. Introduction. In *Language Teaching and Language Technology,* eds. J. Nerbonne, S. Jager & A. van Essen. Lisse: : Swets & Zeitlinger.

Pollard, C. J. 1988. Categorial Grammar and Phrase Structure Grammar: An Excursion on the Syntax-Semantics Frontier. In *Categorial Grammars and Natural Language Strcutures,* eds.

R. T. Oehrle, E. Bach & D. Wheeler, 391-416. Dordrecht/Boston/Lancaster/Tokyo: D. Reidel Publishing Company.

Schulze, M. (1998) Checking Grammar - Teaching Grammar. *Computer Assisted Language Learning,* 11**,** 215-227.

---. 2001. Textana - Grammar and Grammar Checking in Parser-Based CALL. In *Center for Computational Linguistics*, 323. Manchester: UMIST.

--- (2008) Interfaces in Intelligent CALL. *Computer Assisted Language Learning,* 21**,** 301 - 303.

Shaalan, K. F. (2005) An Intelligent Computer Assisted Language Learning System for Arabic Learners. *Computer Assisted Language Learning,* 18**,** 81 - 109.

Skrelin, P. & N. Volskaja. 1998. The Application of new Technologies in the Development of Education Programs. In *Language Teaching and Language Technology,* eds. J. Nerbonne, S. Jager & A. van Essen. Lisse: Swets & Zeitlinger.

Steedman, M. (1985) Dependency and Coordination in the Grammar of Dutch and Enlgish. *Language,* 61**,** 523-568.

Tschichold, C. (1999) Intelligent Grammar Checking for CALL. *ReCALL,* 11**,** 5-11.

Visser, H. (1999) CALLex: a CALL Game to Study Lexical Relationships Based on a Semantic Database. *ReCALL,* 11**,** 50-56.

Witt, S. & S. Young. 1998. Computer-Assisted Pronunciation Teaching Based on Automatic Speech Recognition. In

Language Teaching and Language Technology, eds. J. Nerbonne, S. Jager & A. van Essen, 25-35. Lisse: Swets & Zeitlinger.

Wood, P. (2008) Developing ICALL tools using GATE. *Computer Assisted Language Learning,* 21**,** 383-392.