A Student Model of Technical Japanese Reading Proficiency for an Intelligent Tutoring System[†]

Yun-Sun Kang Anthony A. Maciejewski
School of Electrical & Computer Engineering
Purdue University
West Lafayette, IN 47907
Phone: 765-494-9855
Fax: 765-494-6951
E-mail: maciejew@ecn.purdue.edu

Keywords: Japanese, technical literature, intelligent tutoring systems, natural language processing, artificial intelligence, parsing.

Abstract

This paper presents the development of a student model that is used in a Japanese language intelligent tutoring system to assess a pupil's proficiency at reading technical Japanese. A computer-assisted knowledge acquisition system is designed to generate a domain knowledge base for a Japanese language intelligent tutoring system. The domain knowledge represents a model of the expertise that a native English speaker must acquire in order to be proficient at reading technical Japanese. The algorithms described here are able to generate a set of grammatical transformation rules that clarify changes of syntactic structures between a Japanese text and its corresponding English translation, use them to assess the student's proficiency, and then appropriately individualize the student's instructions.

[†]This material is based upon work supported by the National Science Foundation under Grant No. INT-8818039 and in part by the NEC Corporation and a Purdue University Global Initiative Grant.

I. Introduction

Interest in Japanese language instruction has risen dramatically in recent years, particularly for those Americans engaged in technical disciplines. However, the Japanese language is generally regarded as one of the most difficult languages for English-speaking people to learn. While the number of individuals studying Japanese is increasing there remains an extremely high attrition rate, estimated by some to be as high as 80% [11]. Some of these difficulties are mitigated by using an intelligent tutoring system [9]. Intelligent tutoring systems (ITSs) are computer programs that can individualize their instruction based on inferences about a student's knowledge. One of the most important aspects in developing an ITS is the assessment of the knowledge base of the student [1]. While many knowledge acquisition tools for expert systems have been developed in previous years [2], [5], most of their techniques are not directly applicable to the knowledge acquisition process of an ITS for foreign language learning [21]. However, one tool that *has* demonstrated great promise for second language acquisition is natural language processing [12], [13], [24].

This paper presents the development of a student model, using natural language processing tools, that is used in a Japanese language intelligent tutoring system to assess a pupil's proficiency at reading technical Japanese. A computer-assisted knowledge acquisition system is also designed to generate a domain knowledge base for this ITS. The domain knowledge consists of a set of grammatical transformation rules that clarify changes of syntactic structures required for a native English speaker to comprehend Japanese. Corpora taken from a textbook for learning technical Japanese [4] are used as input data, which consist of 48 Japanese sentences and the corresponding English translation. The sentences from the source language (L1) and the target language (L2) are syntactically analyzed by a Japanese parser originally obtained through the courtesy of the Electro Technical Laboratory (ETL) of Japan [7] and an English parser that was implemented

based on Tomita's algorithm [20]. Both parsers allow the construction of a complete representation of all possible parse trees, i.e., a parse forest, for a given sentence by applying all of the available grammar rules and lexicons. The parse forest is stored in a shared packed parse forest structure [20]. Between individual nodes of the parse forest pairs, a cross-relation is derived depending on whether the parts of text corresponding to the nodes match each other. By associating this relationship with a rule that governs the modification of syntactic structures, a knowledge base of grammatical transformation rules is formed. This information is then used by an intelligent tutoring system developed previously [9] as the domain knowledge that a student must acquire in order to be proficient at reading technical Japanese.

The remainder of this paper is organized as follows: In the next section, intelligent tutoring systems are overviewed by comparing the structure of existing ITSs with the Nihongo Tutorial System [9]. Section III provides a brief introduction to the shared packed parse forest structure and the characteristics of grammar rules employed in the Japanese parser and the English parser. In Section IV, the underlying problems in the parse forest matching process are discussed. Section V describes a data structure to represent a grammatical transformation rule. This is followed by the discussion of a metric function designed to estimate the likelihood of a match in Section VI. By applying this metric function to matching the L1 and L2 parse forests, a top-down matching process is performed to generate grammatical transformation rules, the details of which are illustrated with an example in Section VII. A simple illustration of how a student model is formed by analyzing the responses of a student is presented in Section VIII. Finally, the conclusions of this work are provided in the last section.

II. Overview of Intelligent Tutoring Systems

Intelligent tutoring systems (ITSs) are computer programs that can individualize their instruction based on inferences about a student's knowledge. While existing ITSs vary in architecture, they typically consist of at least four basic components [10], [23]; the expert knowledge module, the student model module, the tutoring module, and the user interface module. The expert knowledge module provides the domain knowledge that the system intends to teach. The student model refers to the dynamic representation of a student's competence for the given domain. The tutoring module is the part of the ITS that designs and regulates instructional interactions with the student. Finally, the fourth component of intelligent tutoring systems is the user interface module, which controls interactions between the system and the student. More details on ITS structure and previously developed prototypes are available in [14], [15], [16], [18], [25].

The specific ITS being considered in this work is called the Nihongo Tutorial System and was designed to assist English-speaking scientists and engineers acquire Japanese reading proficiency in their technical area of expertise [9]. The architecture of this system closely resembles the general structure of ITSs outlined above. The domain knowledge database, which is the focus of this work, is prepared by a software module called the Parse Tree Editor that processes technical journal articles into instructional material by incorporating syntactic, semantic, phonetic, and morphological information into a representation known as an augmented parse tree. The student model is updated based on a student's interactions with the system. The instructional interactions between the system and a student are regulated by the Administrator module which matches the student's current level of Japanese proficiency and technical area of interest with the available instructional material produced by the Parse Tree Editor. The user interacts with the system through a graphical user interface to request information about the current instructional text or to

obtain examples of material on the same or related concepts.

The expert knowledge module currently used by the Nihongo Tutorial System is designed based on a rule-based representation. This type of knowledge representation is commonly used in algorithmically tractable domains such as mathematics, physics, and programming languages [1] as well as in foreign language learning [19]. The design of the expert knowledge module is closely related to other components in an ITS, especially the student model module. The student model used by the Nihongo Tutorial System falls into the general class of student models known as "overlay" models [3]. This commonly used type of model considers the student's knowledge to be a subset of the expert knowledge base. In the case of the Nihongo Tutorial System the expert knowledge contains the Japanese characters, vocabulary, and the syntactic, morphological, and phonological transformation rules required to understand the Japanese text, along with a number that represents the probability that the student understands that particular piece of knowledge. The remainder of this work will only deal with the expert knowledge base associated with the grammatical transformation rules required to understand Japanese text. The lexical and phonological portions of the rule base have been developed previously [8].

III. PRELIMINARIES

A. The Shared Packed Parse Forest Structure

The structure of phrases and sentences of a language is commonly described in a tree diagram as shown in Fig. 1. Each point in the tree is called a node; and each node represents a structural unit called a constituent. The similarities and differences between constituents are traditionally described based on the various categories to which they belong, e.g. Nouns, Verbs, Noun Phrases. An appropriate category label is, thus, attached to each of the nodes in the tree. Sentences, therefore,

have a hierarchical constituent structure in which sounds are grouped together into words, words into phrases, and phrases into sentences. Each constituent (word or phrase) in a sentence belongs to a specific syntactic category, i.e., lexical or grammatical category. Tables I through III include all of the syntactic categories used in the Japanese and English parsers. Note that the lexical categories used in the English grammar rules are distinguished from grammatical categories by using the symbol "*" in the category label. In tree diagrams, it is quite common to suppress the internal structure of a constituent, when it is not relevant to the point at hand, and to represent it by using a triangle.

For a highly ambiguous grammar, there may be numerous parse trees generated for an input sentence. Instead of storing each of the parse trees separately, the numerous parse trees are stored in the form of an efficient data structure called the shared packed parse forest originally introduced by Tomita [20]. If two or more trees have a common subtree, the subtree is represented only once in the parse forest. The parse forest is called the shared parse forest. If two or more subtrees have common leaf nodes and their top nodes are labeled with the same grammatical category, the subtrees represent local ambiguity. The total ambiguity of a sentence would grow exponentially as the number of local ambiguities increases. The top nodes of subtrees that represent local ambiguity are merged and treated by higher-level structures as if there were only one node. Such a node is called a packed node, and nodes before packing are called subnodes of the packed node.

B. Characteristics of the Japanese and English Grammar Rules

Prior to illustrating the details of the knowledge acquisition system, it is important to understand the characteristics of the Japanese and English grammar rules employed in this work. Syntax is the set of rules governing the combination of words in a sentence. Based on defined syntactic rules, a

parser builds a representation, i.e., a parse tree. Since the structure of a parse tree depends on the way in which grammar rules are defined, one would like to identify distinctions between the two grammars as well as their common aspects.

A typical example of a Japanese and an English grammar rule used in this work is shown here:

Japanese: 文 → 連用修飾句 + 文

English: SREL → *RELPRO + AUXD + VP

where each grammar rule results in a tree structure using the lexical and/or grammatical categories on the right-hand side of the arrow as children and the grammatical category on the left-hand side as the parent. The Japanese parser includes 74 grammar rules, whereas the English parser uses 415 grammar rules. Selected examples of the Japanese and the English grammar rules are presented in Table IV. The Japanese and English grammars are both written in the same formalism called a context-free grammar (CFG). This formalism specifies no context which must be satisfied before constituents can be combined. Context-free grammars are commonly augmented to describe certain language features such as subject-verb agreement, verb conjugation, gender agreement, etc. The Japanese and English grammars both use augmentation, but in the case of the Japanese grammar it is devoted largely to impose a syntactic condition on a constituent.

The Japanese grammar differs from the English grammar in that it includes inflectional morphology, i.e., the way words change in relation to grammatical contexts. Inflectional morphology is considered by many linguists to be distinct from syntax. However, it is included here because Japanese is heavily inflected compared to English, and this inflectional system must be mastered by students of the language [17]. An additional difference lies in the manner of assigning a syntactic category to a constituent. This is due to the fact that both the Japanese and the English grammar rules are specifically implemented for their own languages. For example, in the Japanese

grammar, all of the declinable[1] words, which include verbs and adjectives, are assigned the same lexical category. In addition, some constituents are labeled based on their straightforward relation to neighboring constituents, and not on their specific syntactic function within the context. For example, the grammatical category "連用修飾句", which literally means "the phrase that modifies a declinable word or phrase", can become a Subject, an Object, or a Prepositional Phrase. Consequently, the distinctions that exist between the two grammars creates a tremendous difficulty in matching parse forest pairs. To address some of these difficulties, a universal name is assigned to each syntactic category in both the Japanese and English grammars as listed in Tables I through III.

IV. The Parse Forest Matching Problem

Before going into the details of the algorithm, it is important to appreciate the underlying difficulties in the parse forest matching problem. Unlike general pattern matching problems, the name or label of a node cannot be used as a pattern for a match. In the parse forest matching problem, the nodes that have identical names are likely to be matched, however, nodes of different names can also be matched because of the possible change in syntactic structure during translation. Clearly, the top node of the L1 parse forest is directly matched with the top node of the L2 parse forest since both nodes are associated with entire sentences. Leaf nodes may match each other when the word associated with a node in L1 is directly translated into a single word in L2. It is, however, not necessary for nodes to be in the same level in a parse forest (either from the top or from the bottom) in order to be matched. Ambiguities naturally embedded in a sentence as well as existing in the grammar employed in this work are likely to generate numerous parse trees. These ambiguities contribute to increasing difficulty in matching parse forests because of the enormous

[1] Having case inflections.

number of possibilities for a match. Due to these facts it is not possible to adapt general pattern matching algorithms [6], [22] for this work. The following sections describe a more domain-specific algorithm designed to deal with all of these difficulties. Central to this algorithm is the data structure used to represent grammatical transformation rules, which is the topic of the next section.

V. The Structure of Grammatical Transformation Rules

In order to store and retrieve information effectively, a well-designed data structure for the grammatical transformation rules is needed. To illustrate the information stored in the rule base, a specific example will be considered using the following Japanese sentence and its corresponding English translation:

Japanese: “速度というのは速さと向きで表わされるものである。”

English: “Velocity is a quantity which is described by speed and direction.”.

A part of the parse forest that was constructed for this sentence by the Japanese and the English parsers is illustrated in Fig 2, where the internal structure of the constituents is suppressed for the sake of clarity. Note that a node in the parse forest is described by its syntactic category along with a unique identification number. A grammatical transformation rule results from a match of nodes in the L1 parse forest with nodes in the L2 parse forest. In this example, the node 文251 in the Japanese parse forest is matched with the node SREL113 in the English parse forest (indicated in Fig. 2 using boldface). Thus the grammatical transformation rule in this case is represented by

(文 → 連用修飾句 + 文) ⇒ (SREL → *RELPRO + AUXD + VP)

where the arrow “⇒” indicates the direction of transformation in syntactic structure. The source and target of the grammatical transformation rule consist of grammar rules associated with the nodes that matched. For this example, the source and target include only one grammar rule, however,

multiple grammar rules are also likely because of multiple nodes being possible for either side of a match. It is also possible that the source or target of a grammatical transformation rule can be null, i.e., a node may not be matched with any node. The occurrence of this null rule is mainly due to the frequency of ellipsis[2] in Japanese but also occurs due to grammatical components that only occur in one language, such as particles in Japanese and relative pronouns in English.

The first time that a grammatical transformation rule is encountered, it must be added to the rule base as illustrated in Fig. 3 where the portion of the tree represented by using a dashed line indicates the existing grammatical transformation rules. The new grammatical transformation rule generated from the match of 文254 and SREL113, along with its context, "((名詞句→文 + 名詞句) ⇒(NP→NP + SREL))", is stored in the rule base tree structure and connected to its parent "(文⇒SREL)", assuming it already exists. The parent, which consists of the names of the constituents on the left-hand side of the Japanese and English grammar rules used in the transformation, possibly has multiple children because the constituents generated from different grammar rules can have the same label, i.e., some grammar rules have an identical grammatical category on the left-hand side of their rules. If the grammatical transformation rule encountered is already in the rule base then a frequency counter is updated. After the parse forests of each data sentence have been matched completely, the frequencies of all the rules in the rule base are calculated by dividing the number of occurrences of a rule by the total number of all occurrences for all rules that have the same right-hand side.

The primary advantage of using the hierarchical tree structure for the grammatical transformation rule base is that this structure makes it possible to effectively retrieve the rule's frequency from the rule base. Due to the fact that a large number of Japanese and English grammar rules are

[2]The omission of part of a sentence, where the missing element is understood from the context.

being used, numerous grammatical transformation rules are possibly generated. Since grammatical transformation rules that result from the match of the same constituents will be saved under the same parent in the tree structure, when information is retrieved from the rule base, only the top of the hierarchical tree structure needs to be searched. If there exists a grammatical transformation rule that has the same constituents on the top of the tree structure, then its internal tree structure will be searched. In addition to improving efficiency in searching the rule base, the names of constituents that are stored in the parent provide information about the structural transformation between the two languages. Another advantage of this hierarchical tree structure lies in saving the information about the situation in which a grammatical transformation rule is generated. The context of a match is useful for prospective tutoring, for identifying the characteristics of the rules, and for validating the grammatical transformation rule.

VI. Metric for Estimating Likelihood of a Match

This section presents a metric that is designed to estimate the likelihood of a match between nodes in the L1 and L2 parse forests based on lexical and grammatical properties of the nodes. The primary source in measuring the likelihood of a match is the word-level information between the L1 and L2 sentences. The word-to-word relations are gathered by searching an on-line glossary which is available for words in the data sentences. Then the leaf nodes in the L1 and L2 parse forests, which are associated with either a word or a morpheme[3], are matched based on the word-to-word relations. When the leaf nodes correspond only to morphemes, i.e., in case of Japanese declinable words, their parent nodes need to be analyzed further. This process is conducted during the pre-processing step of the Japanese and English texts. To illustrate how the lexical information

[3]The smallest distinctive unit of grammar.

is applied to measuring the likelihood of a match, consider the possible match between the nodes 連用修飾句125 and VP127 as shown in Fig. 4 where the words identified by using an on-line glossary are circled and the relationship to the corresponding translation is represented by an arrow. When focusing on only the texts corresponding to the two nodes 連用修飾句125 and VP127, the Japanese words "速さ" and "向き" appear to be matched with the English words "speed" and "direction", respectively. Clearly, nodes that include more words in common are more likely to be matched. The converse is also true, i.e., a match between nodes becomes unlikely if words included in the node of L1 do not match the words in the node of L2 or vice versa. For example, consider the Japanese word "表わさ"and the English translation "described" which are identified as depicted in Fig. 4. By considering this word-to-word relation, it is clear that the node 連用修飾句125 should not be directly matched with the node VP127, but will require further processing.

In order to construct a metric based on the lexical matches, a match between the node A in L1 and the node B in L2 is considered. A metric for estimating the likelihood of a node match is then determined as the difference in numbers between the two different types of matches, i.e., x, the total number of words in common between A and B, and y, the number of words matched with ones included in nodes other than A or B. Clearly, any match for y is a strong indication that the nodes A and B are not likely to be matched with each other, assuming that the information obtained from the on-line dictionary is correct. Therefore the variable y is considered to be more important than x in estimating the likelihood of a match by assigning a bigger weight to the variable y. Consequently the metric used in this work is given by

$$M_l(x, y) = l_1 x - l_2 y, \qquad 0 < l_1 \ll l_2. \tag{1}$$

Due to the fact that the lexical information by itself is insufficient in determining a correct

match, grammatical information contained in a node is also considered. Typically a word is more likely to be translated into the same type of lexical category from one language to another language. Moreover, constituents that have similar syntactic properties are more likely to be matched with each other. This information is also incorporated into the metric for determining how likely it is for a node in the L1 parse forest to be matched with a node in the L2 parse forest. It is, however, not always possible to directly match the labels of nodes, i.e., the syntactic category of a constituent. This is mainly due to the fact that the grammar rules that are employed in the Japanese parser adapt a categorization for constituents that is different from the one used in the English grammar rules. As mentioned in Section III, some constituents are categorized in the Japanese grammar by focusing mainly on the relationship between modifier and modifiee. Thus the syntactic function of the constituent is not clearly defined in its grammatical category. This results in one Japanese constituent being matched with several different English syntactic categories.

In order to resolve this problem, first, further syntactic analysis is conducted on a node in the Japanese parse forest to see if the node contains a particle at the end of the text associated with the node. Japanese particles at the end of a phrase or clause along with the modifiee determine their syntactic property, regardless of the order of the phrase. Table V lists examples of particles that identify specific syntactic categories. Two or more of these universal names are assigned to the particles that possibly impose multiple syntactic structures. The syntactic structure for a node obtained from the analysis of its particle is then used to measure the likelihood of a match by checking the universal name with a constituent's universal name on the other side of a possible match. If a node does not include a particle, the constituents are checked to see if they have the same universal names. When two or more constituents are considered in either side of a match, the universal name that is common for the constituents is used. The metric based on the matches

of the universal names for constituents is determined as either one or zero.

Additional information in estimating the likelihood of a match between nodes in the L1 and L2 parse forests is obtained from the grammatical transformation rules that were generated from past input data. Each entry in the grammatical transformation rule base is associated with its frequency of occurrence. In order to retrieve this information, the following procedure is performed:

1. The top level of the hierarchical rule base is searched for a grammatical transformation rule (GTR) that has the same syntactic categories as the nodes that are being considered for a possible match.

2. If a GTR is found in the preceding step, the child nodes of this GTR are searched to see if the Japanese and English grammar rules associated with the GTR are the same as the nodes that are being considered for a possible match.

3. If the GTR is generated from the same Japanese and English grammar rules as the nodes that are being considered for a possible match, its children are checked to see if the context of the GTR is the same.

A metric based on the rule base is then defined as the largest value of the frequencies f_1, f_2, and f_3 obtained from step 1 through step 3 .

An overall metric for estimating the likelihood of a match between the nodes in the L1 and L2 parse forests is constructed by combining the information from the three different sources described above, i.e., the metric based on the lexical matches M_l, the metric based on the comparison of the universal names for constituents M_u, and the metric based on the rule base M_f. The overall metric is then defined as

$$M_{node} = \alpha M_l + \beta M_u + \gamma M_f \tag{2}$$

where α, β, and γ are coefficients for the linear combination of M_l, M_u, and M_f. The rule base is initially empty, however, the amount of information available from the rule base keeps increasing as data is processed. Thus the effect of the rule base becomes greater in estimating the likelihood for a match. To address the relationship between M_u and M_f, the coefficients in (2) are empirically determined as follows:

$$\beta = e^{-N}, \quad \gamma = (1 - e^{-N}) \tag{3}$$

where N is the number of texts processed. Due to the fact that the lexical information is much more significant than the other two sources, $\alpha \gg 1$. The next section will present the details of the top-down parse forest matching process with an example in which the relationship between children of parents that have been matched needs to be identified correctly.

VII. The Top-down Parse Forest Matching Process

A. Algorithm Description

The algorithm developed in this work performs the matching process in a top-down manner. A matching process begins from the highest level of a parse forest, i.e., the top node, and continues down to the lowest level, i.e., the leaf nodes. First, the top node of a parse forest is expanded to get its child nodes, which point to the subsequent parse forests. When the top node has only one child node, the child node continues to be expanded to prevent the matching process from being trivial. Before considering all of the possible matches between these child nodes, the word-to-word relationship obtained during the pre-processing step of the Japanese and English texts is applied to the nodes. If the word associated with any of the child nodes does not exist in the other

language, the node is matched with null. All of the possible matches are then considered between the remaining child nodes. Each possible match can be regarded as the combination of different potential grammatical transformation rules. To determine the most likely match, first, the metric for the likelihood of a node match is computed by using (2) for all combinations of the potential grammatical transformation rules included in the match. The metrics for all of these potential grammatical transformation rules in a combination are then averaged to obtain the most likely node match. This procedure is then recursively applied to each of all the potential grammatical transformation rules generated from the most likely match until no more non-terminal node remains to be matched.

This top-down method is much more efficient for parse forest pairs that are incorporated into relatively large numbers of parse trees as compared with a bottom-up approach. This is due to the fact that the top-down method makes it possible to keep the search space smaller by disambiguating a parse forest. Consider a parse forest in which the top node is packed, i.e., consists of multiple sets of child nodes, each set being associated with respective ambiguity existing in the constituent corresponding to the top node. An ambiguity that occurs due to the grammar rules and an ambiguity that exists in either the L1 or the L2 parse forest can be clearly detected. For the ambiguity that is naturally embedded in the L1 parse forest and also in the L2 parse forest, a domain expert will choose one by analyzing its context as well as by using his underlying domain knowledge. Only one set of child nodes would, therefore, be selected for further processing and the nodes included in the other sets are then completely excluded. This results in effectively removing the unnecessary part of the parse forest from a match.

B. An Example

This section presents an example of generating grammatical transformation rules, where the relationship between children of parents that have been matched needs to be identified correctly. This example will be considered using the following Japanese and English parse forests:

Japanese: 文251 $\Longrightarrow$ (連用修飾句125 文248)

English: SREL113 $\Longrightarrow$ (*RELPRO34 AUXD40 VP127) (SREL44 PP124)

where the arrow shows the direction of expansion that is from a parent node to its child nodes. Note that the multiple lists on the right-hand side of the arrow indicate that the node is of a packed type. Before considering all of the possible matches between children, the node *RELPRO34 is automatically matched by null during the pre-processing step as illustrated in Fig. 4 since the relative pronoun “which” does not exist in Japanese. All of the possible matches are formed between the remaining child nodes. Consider possible matches between the nodes (連用修飾句125 文248) and (AUXD40 VP127). None of these nodes is allowed to be matched with null because grammatical components that occur only in one language have been already matched with null and ellipsis is not considered until the metric for each possible match is computed. Thus, there exist two possible matches between the nodes (連用修飾句125 文248) and (AUXD40 VP127), i.e., 連用修飾句125 with AUXD40 and 文248 with VP127, or vice versa. Since each of the multiple lists of child nodes independently contributes to generating possibilities, another subnode of the packed node, i.e., (SREL44 PP124) forms two more possible matches, consequently, four matches are possible in this example.

For each of all the possible matches, the likelihood is estimated by taking an average of the values of likelihood for potential grammatical transformation rules. When considering possible matches between (連用修飾句125 文248) and (AUXD40 VP127), it appears that a node must

become a part of two or more grammatical transformation rules based on the word-to-word relations obtained during the pre-processing step as shown in Fig. 4. Whereas the node AUXD40 needs to match 文248, the node VP127 must match both 文248 and 連用修飾句125. In order to resolve this problem the node VP127 is expanded, i.e., subdivided as depicted in Fig. 5. All of the possible matches between (連用修飾句125 文248) and (AUXD40 VP43 PP124) are, then, considered instead. In Fig. 5 the match is being performed between the nodes (連用修飾句125 文248) in the Japanese parse forest and (AUXD40 VP43 PP124) in the English parse forest. The most likely match is selected based on the estimation listed in Table VI, i.e., (連用修飾句125)→(PP124) and (文248)→(AUXD40 VP43). The results of matches throughout the entire parse forests are used to generate grammatical transformational rules after the matching processes are completed.

VIII. The Student Model

The grammatical transformation rules determined in the previous section are used to define the domain knowledge that a student must acquire in order to become proficient at reading technical Japanese. Some examples of these rules resulting from processing the 48 Japanese sentences and their corresponding translation are illustrated in Table VII. For each student that uses the intelligent tutoring system, a database is maintained which keeps information on which of the rules the student has mastered and which ones need further review. This information is obtained both through active testing of the student as well as by passive monitoring of the student's requests for information while using the tutoring system. The Japanese sentences with which a student has difficulty are analyzed to determine which grammatical transformation rules are present and this is compared to the rules which occur in the sentences that are easily comprehended. The tutoring system then attempts to assist the student in acquiring the unfamiliar rules by presenting lessons which include

sentences that use these rules.

A difficulty of estimating the student's current knowledge state is due to the fact that factors other than syntax come into play. For example, the student may guess the correct translation from context or conversely, the student may not comprehend the meaning of a sentence due to cultural factors. The following presents a simple illustration of how the student model is formed by analyzing the responses of students who were tested for their technical Japanese reading proficiency. The data used here was obtained from a Japanese grammar test which was conducted on 10 students ranging from 2 to 4 years of classical Japanese language instruction. The test consisted of 35 sentences related to basic physics which were selected from the corpora used to generate the domain knowledge base discussed in the previous section. Each question in the test was accompanied with a word glossary which provided an English translation for all of the independent words, i.e., nouns, verbs, and adjectives. This format for the test was designed to focus on grammar proficiency by separating out any lexical factors that contribute to comprehension.

Consider the particular students labeled (A) and (B) who had difficulty in understanding the following Japanese sentence:

(I) このように、速度が一定の運動を等速度運動、または等速直線運 動という。

which can be translated into English as

Motion at constant velocity is called uniform velocity motion or uniform linear motion.

where the underlined portions are equivalent. The following are the translations provided by the students:

(A) In this manner, velocity as a constant motion is called either a constant motion velocity or a uniform linear motion.

(B) In this way, we refer to the velocity of constant motion as constant velocity or, in other words, uniform linear motion.

From analyzing the students' translations, the tutoring system was able to identify which part of the Japanese sentence the students could not understand and then to determine which of the grammatical transformation rules the students had not mastered. First, the students' translations were syntactically analyzed using an English parser, and then this resulting parse tree was compared to the parse tree generated from the correct English translation. The goal of this process is to search for any structural difference between the two parse trees. This process is illustrated in Fig. 6 for the portions of the parse trees that correspond to the underlined text in the above example. While both students' parse trees are constructed by an English grammar rule that is identical to that used in the correct translation, i.e., "NP → NP + PP", the words that belong to the nodes NP and PP in the parse tree for the students' translation are not the same as in the parse tree for the correct English translation. The tutor makes the approximate assumption that two parse trees represent the same meaning only when they have the same tree structure with the same words under each corresponding node of the parse trees. This portion of the students' translations, therefore, has a different meaning from the correct English translation, which indicates that the students are misunderstanding this portion of the Japanese sentence. The rule which students (A) and (B) have not mastered here is the transformation

(名詞句 → 文 + 名詞句) ⇒ (NP → NP + PP),

which governs the transformation from the Japanese syntactic structure "名詞句 → 文 + 名詞句" to the English syntactic structure "NP → NP + PP". This rule represents the significant transformation from pre-positional modification (文 modifies 名詞句) to post-positional modification (PP modifies NP), which mainly creates the students' misunderstanding. It is significant to note that the rule

identified here is correlated with empirical evidence noted by instructors using classical Japanese language instruction. This is due to the fact that this rule represents one of the major differences between Japanese and English syntactic structure.

To verify the estimation of student (A)'s reading proficiency obtained from analyzing his response to the previous Japanese sentence, consider student (A)'s translation for the following Japanese sentence:

(II) 速度というのは速さと向きで表わされるものである。

which can be translated into English as

Velocity is a quantity which is described by speed and direction.

From analyzing student (A)'s translation, "Velocity is speed in a certain direction.", it was found that the student also has the same difficulty in understanding this Japanese sentence as the one which he encountered in the previous sentence. As illustrated in Fig. 7, the student could not successfully translate most of the Japanese sentence. The rule which the student has not mastered in this example is the transformation,

(名詞句 → 文 + 名詞句) ⇒ (NP → NP + SREL),

which governs the transformation from the Japanese syntactic structure "名詞句 → 文 + 名詞句" to the English syntactic structure "NP → NP + SREL". Notice that this rule also represents the transformation from pre-positional modification to post-positional modification that the student had not mastered in the previous sentence.

IX. Conclusions

The goal of this work was the development of a computer-assisted knowledge acquisition system designed to generate a domain knowledge base that represents a model of the expertise that a native English speaker must acquire in order to be proficient at reading technical Japanese. This domain knowledge base is used to assess a student's competence of reading technical Japanese, and to individualize the instruction of an intelligent tutoring system that is designed to assist scientists and engineers acquire a reading knowledge of technical Japanese. To accomplish this goal an algorithm was developed to generate the rules that govern the transformations that Japanese sentences undergo when being translated into English. These rules are used as the domain knowledge base against which a student's performance is measured.

REFERENCES

[1] J. R. Anderson, "The expert module," in M. C. Polson and J. J. Richardson, editors, *Foundations of Intelligent Tutoring Systems*, pp. 21–53. Hillsdale, NJ: Lawrence Erlbaum Associates, 1988.

[2] J. H. Boose and J. M. Bradshaw, "Expertise transfer and complex problems: Using AQUINAS as a knowledge-acquisition workbench for knowledge-based systems," *Int. J. Man-Machine Studies*, vol. 26, pp. 3–28, 1987.

[3] B. Carr and I. Goldstein, *Overlays: A Theory of Modeling for Computer Aided Instruction*, Cambridge, MA: MIT, Artificial Intelligence Laboratory, 1977.

[4] E. E. Daub, R. B. Bird, and N. Inoue, *Comprehending Technical Japanese*, Tokyo, Japan: Univ. of Tokyo Press, 1975.

[5] J. Diederich, I. Ruhmann, and M. May, "KRITON: A knowledge-acquisition tool for expert systems," *Int. J. Man-Machine Studies*, vol. 26, pp. 29–40, 1987.

[6] C. M. Hoffmann and M. J. O'Donnell, "Pattern matching in trees," *J. ACM*, vol. 29, pp. 68–95, 1982.

[7] H. Isahara, "Analysis and semantic representation in CONTRAST, A context-based machine translation system," *Bulletin of the Electrotechnical Laboratory*, vol. 57, no. 2, pp. 161–173, 1993.

[8] A. A. Maciejewski and Y.-S. Kang, "A student model of katakana reading proficiency for a Japanese language intelligent tutoring system," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-24, pp. 1347–1357, 1994.

[9] A. A. Maciejewski and N. K. Leung, "The Nihongo Tutorial System: An intelligent tutoring system for technical Japanese language instruction," *CALICO Journal*, vol. 9, pp. 5–25, 1992.

[10] H. Mandl and A. Lesgold, editors, *Learning Issues for Intelligent Tutoring Systems,* New York, NY: Springer-Verlag, 1988.

[11] D. O. Mills, R. J. Samuels, and S. L. Sherwood, "Technical Japanese for scientists and engineers: Curricular options," Technical Report MITJSTP WP 88-02, MIT, Cambridge, MA, 1988.

[12] N. Nagata, "An effective application of natural language processing in second language instruction," *CALICO Journal*, vol. 13, pp. 47–67, 1995.

[13] N. Nagata, "Computer vs. workbook instruction in second language acquisition," *CALICO Journal*, vol. 14, pp. 53–75, 1996.

[14] O.-C. Park, R. S. Perez, and R. J. Seidel, "Intelligent CAI: Old wine in new bottles, or a new vintage?," in G. Kearsley, editor, *Artificial Intelligence and Instruction,* pp. 11–45. Reading, MA: Addison Wesley, 1987.

[15] M. C. Polson and J. J. Richardson, editors, *Foundations of Intelligent Tutoring Systems,* Hillsdale, NJ: Lawrence Erlbaum Associates, 1988.

[16] J. W. Rickel, "Intelligent computer-aided instruction: A survey organized around system components," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-19, pp. 40–57, 1989.

[17] A. F. Sanders and R. H. Sanders, "Syntactic parsing: A survey," *Computers and the Humanities*, vol. 23, pp. 13–30, 1989.

[18] C. B. Schwind, "An intelligent language tutoring system," *Int. J. Man-Machine Studies*, vol. 33, pp. 557–579, 1990.

[19] M. L. Swartz, "Issues for tutoring knowledge in foreign language intelligent tutoring systems," in M. L. Swartz and M. Yazdani, editors, *Intelligent Tutoring Systems for Foreign Language Learning*, pp. 219–233. Berlin: Springer-Verlag, 1992.

[20] M. Tomita, *Efficient Parsing for Natural Language*, Boston, MA: Kluwer Academic, 1986.

[21] M. B. Twidale, "Knowledge acquisition for intelligent tutoring systems," in F. L. Engel, D. G. Bouwhuis, T. Bösser, and G. d'Ydewalle, editors, *Cognitive Modelling and Interactive Environments in Language Learning*, pp. 63–71. Berlin: Springer-Verlag, 1992.

[22] R. A. Wagner and M. J. Fischer, "The string-to-string correction problem," *J. ACM*, vol. 21, pp. 168–173, 1974.

[23] E. Wenger, *Artificial Intelligence and Tutoring Systems*, Los Altos, CA: Morgan Kaufmann, 1987.

[24] J. C. Yang and K. Akahori, "Error analysis in Japanese writing and its implementation in a computer assisted language learning system," *CALICO Journal*, vol. 15, pp. 47–66, 1998.

[25] M. Yazdani, "Intelligent tutoring system, An overview," in R. W. Lawler and M. Yazdani, editors, *Artificial Intelligence and Education (Volume 1)*, pp. 183–201. Norwood, NJ: Ablex, 1987.

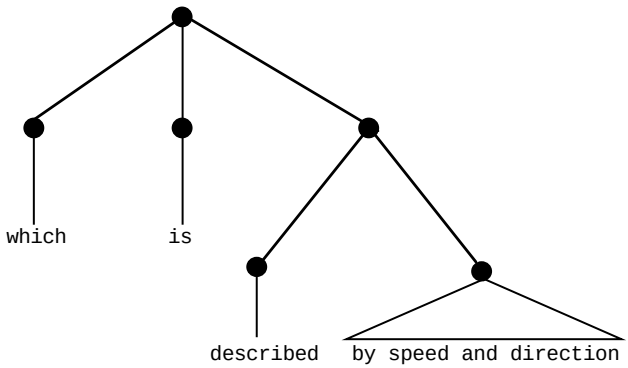
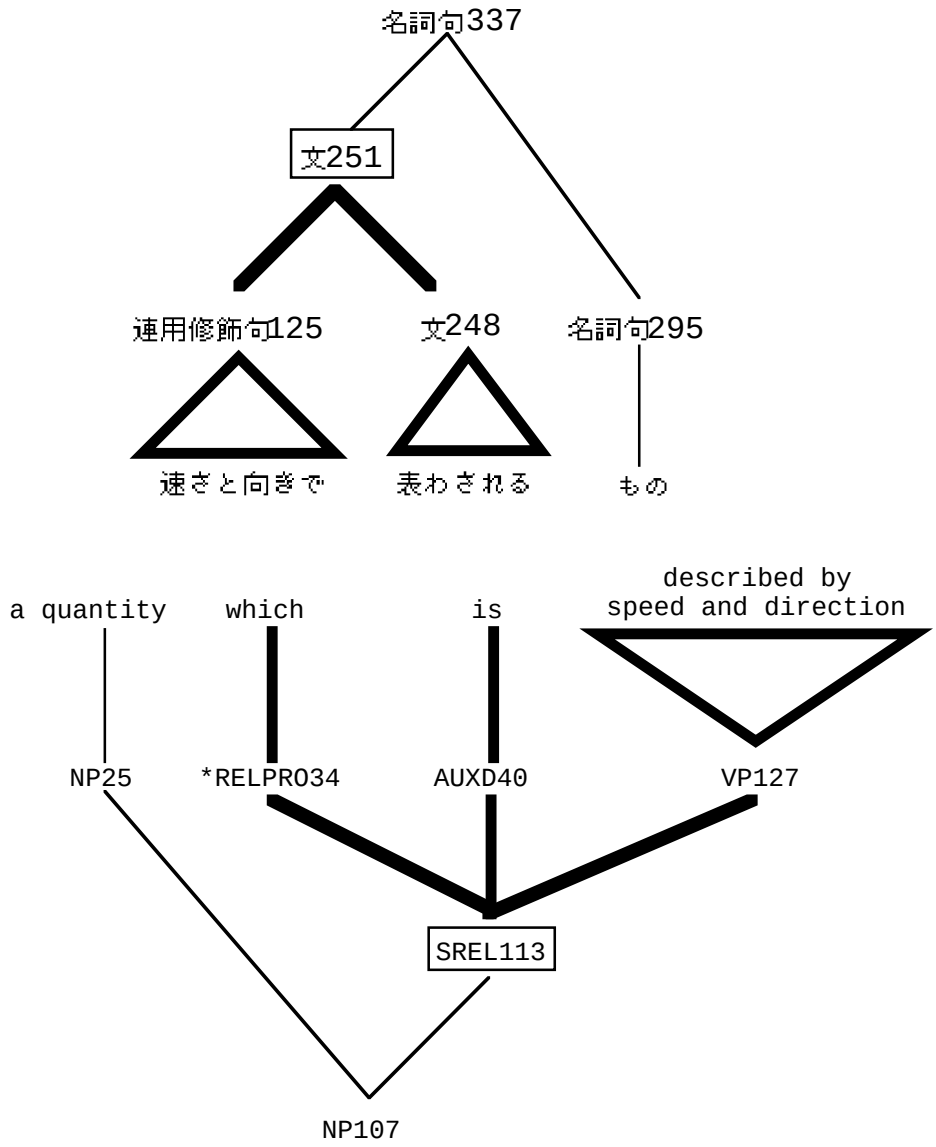


Fig. 1. Tree diagram for the structure of a phrase. Each node in the tree represents a structural unit called a constituent. The internal structure of a node is suppressed by using a triangle when it is not relevant to the point at hand.



L1: 速度というのは速さと向きで表わされるものである。

L2: Velocity is a quantity which is described by speed and direction.

Fig. 2. Comparison of two parse trees based on the relationship between the Japanese text and its corresponding English translation. The Japanese text “速さと向きで表わされるものである” is translated into “a quantity which is described by speed and direction” in English. The nodes 文251 and SREL113 (shown in bold) will be matched.

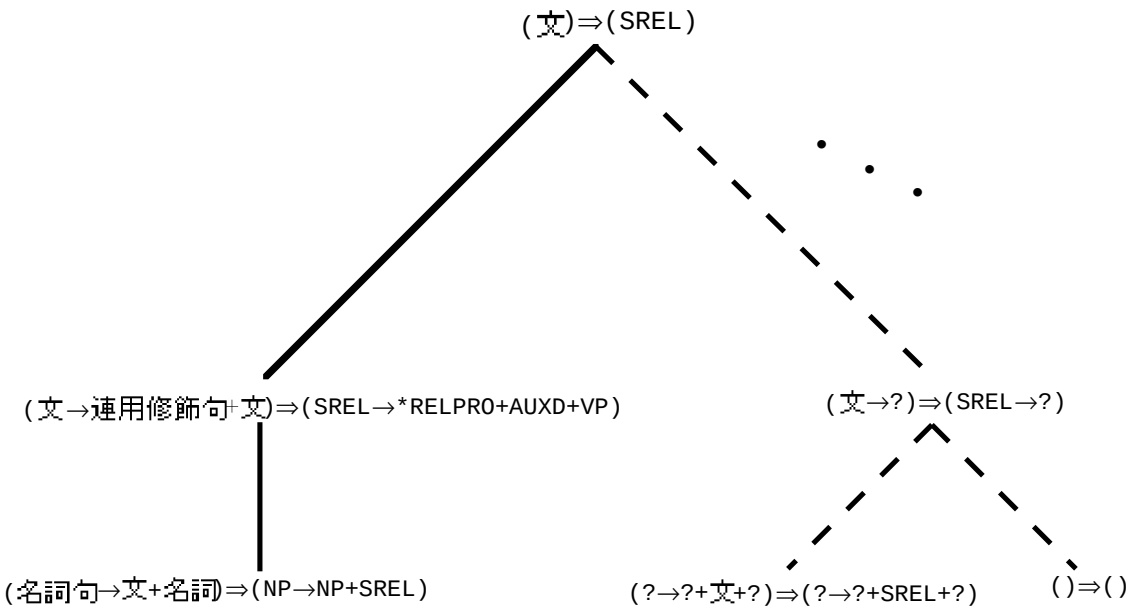


Fig. 3. The hierarchical tree structure of the grammatical transformation rule base. The top level of the tree stores only the syntactic category. The next level includes the specific Japanese and English grammar rules that were matched. The context in which a match occurs is included at the bottom level of the tree. The addition of the grammatical transformation rule acquired from the example in Fig. 2 is shown in bold. The existing grammatical transformation rule base is shown with dotted lines.

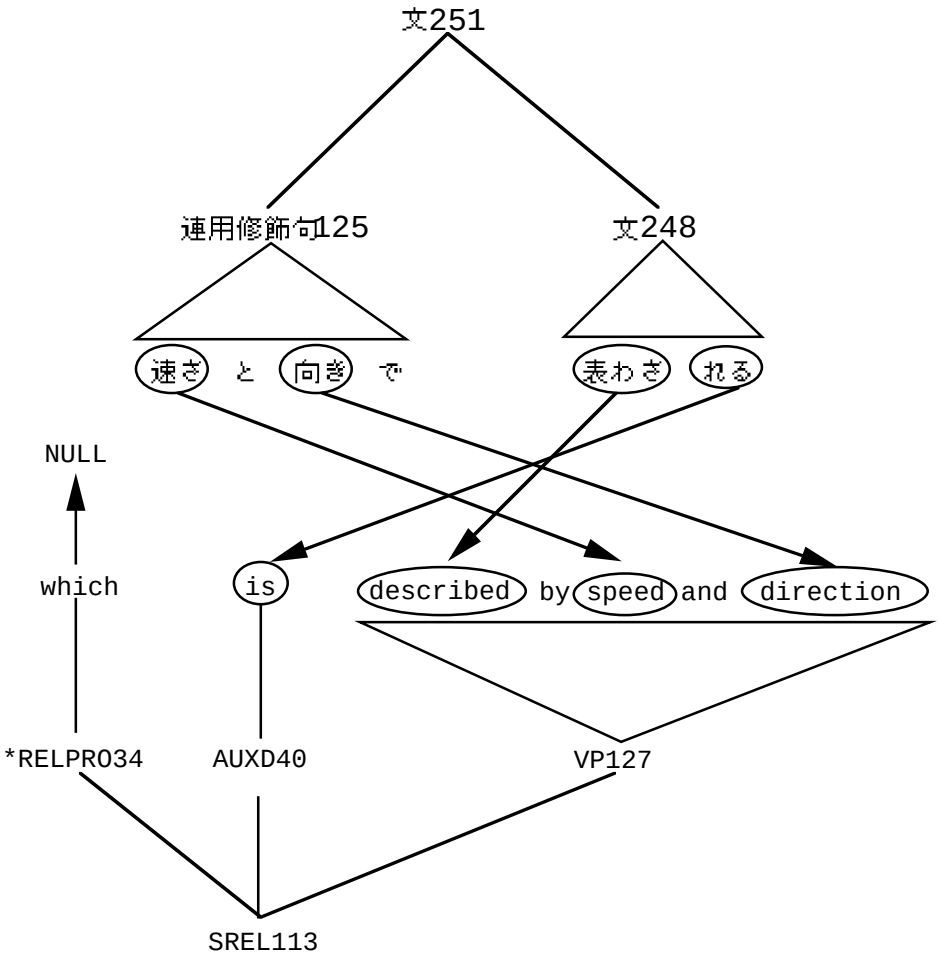


Fig. 4. An example of applying lexical information gathered from an on-line glossary during the pre-processing step of the parse forest matching algorithm. The words identified by using an on-line glossary are circled and the relationship to the corresponding translation is represented by an arrow. While the node 文248 should match with the nodes AUXD40 and VP127, the node 連用修飾句125 also should match with VP127. Thus either 文248 or VP127 must be subdivided. The match of the node *RELPRO34 with null is also identified from the on-line glossary.

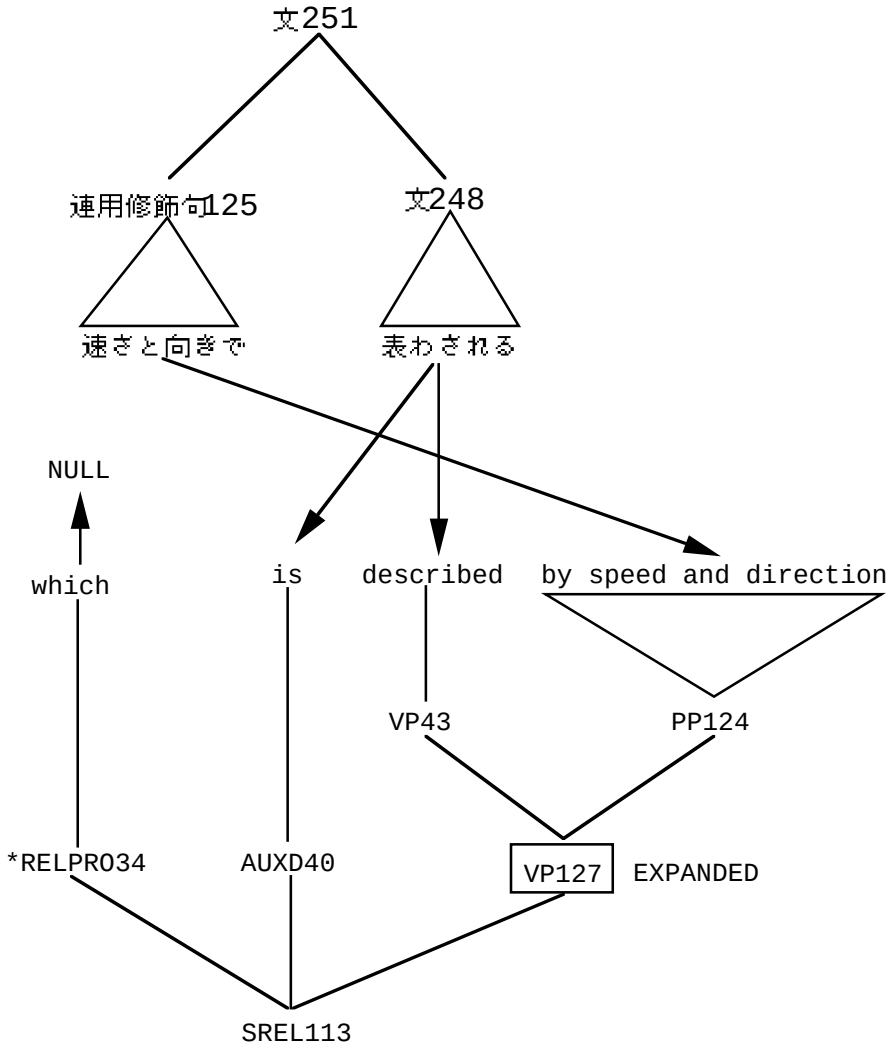


Fig. 5. An illustration of expanding the node VP127 for subdivision, thus resulting in matches between the node 文248 and the nodes AUXD40 and VP43 and between the node 連用修飾句125 and the node PP124.

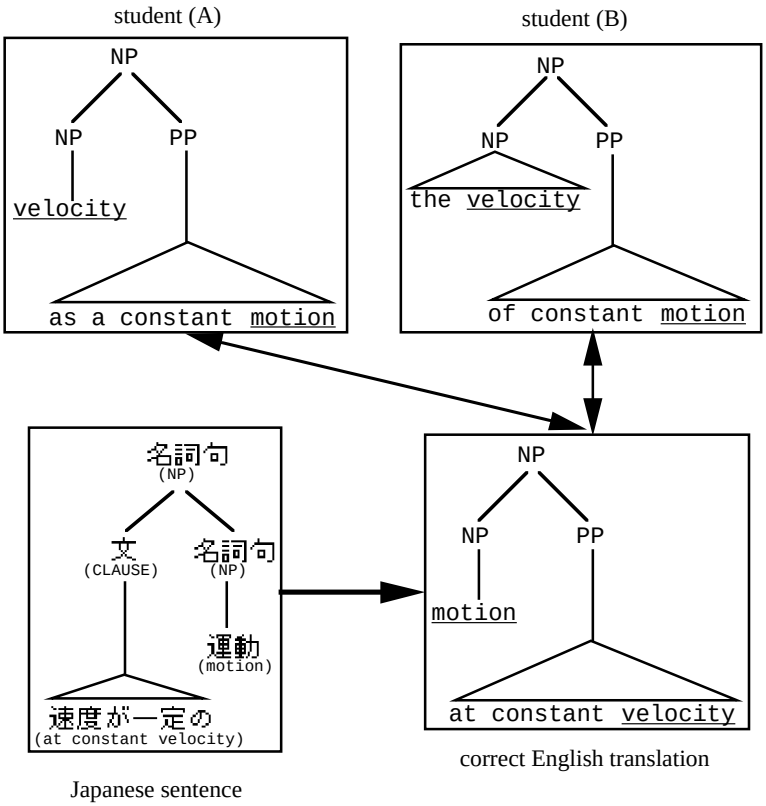


Fig. 6. An example (I) of analyzing students' translations to identify the difficulty that students have in understanding a Japanese sentence. The structural differences that exist between the students' translations and the correct English translation are illustrated in the form of a parse tree. Notice that words underlined in the parse trees should be located in the same node in order to represent an identical meaning.

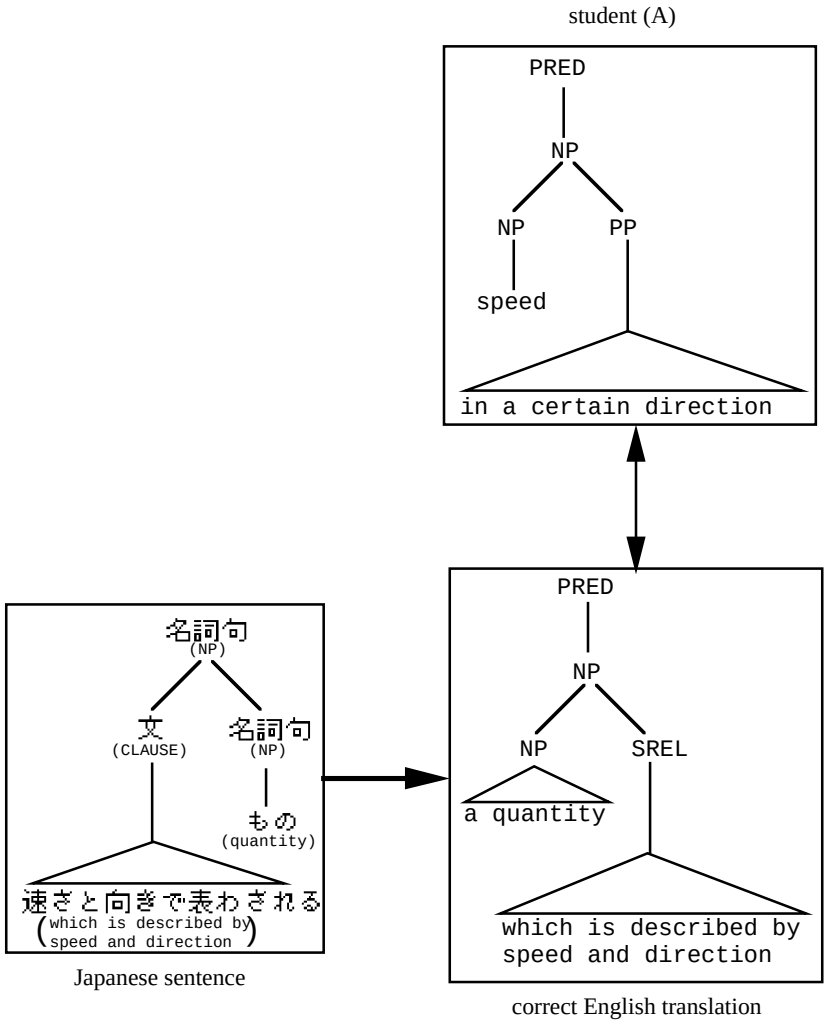


Fig. 7. An example (II) of analyzing a student's translation to identify the difficulty that this student had in understanding a Japanese sentence.

TABLE I
LIST OF UNIVERSAL NAMES FOR SYNTACTIC CATEGORIES USED IN THE JAPANESE GRAMMAR

Syntactic category	Universal name	Syntactic category	Universal name
名詞	noun	接頭語	null
代名詞	pronoun	接尾語	null
固有名詞	proper noun	接尾ー1	null
名詞句	noun phrase	接尾ー2	null
語幹	null	連用助詞	null
語尾	null	連体助詞	null
用言	verb, adjective	接続助詞	conjunction
助動詞	auxiliary verb	終助詞	null
用言句	phrase	体言接続詞	conjunction
連用修飾句	phrase	副詞	adverb
並列連体句	phrase	元号	time
連体修飾句	phrase	年	null
体言止め	clause	@月	time
だー省略	clause	@日	time
文	clause	時刻	time
TOP	top	数	number
END	final punctuation	数字	number
読点	punctuation	倍数	number
括弧	symbol	単位	number
弧括	symbol	順序	number
中黒	symbol	て	null
さ	null	で	null

TABLE II
LIST OF UNIVERSAL NAMES FOR LEXICAL CATEGORIES USED IN THE ENGLISH GRAMMAR

Lexical category	Universal name	Lexical category	Universal name
∗a	null	∗not	null
∗adj	adjective	∗num	number
∗adv	adverb	∗paraconj	conjunction
∗after	null	∗parenthesis	symbol
∗all	null	∗prep	null
∗as	null	∗pron	pronoun
∗be	auxiliary verb, verb	∗propern	proper noun
∗before	null	∗punc	punctuation
∗by	null	∗qdet	null
∗comma	punctuation	∗quant	null
∗conj	conjunction	∗reflexive	null
∗det	null	∗relpro	null
∗do	auxiliary verb, verb	∗so	null
∗enough	null	∗subconj	null
∗equation	symbol	∗than	null
∗finalpunc	final punctuation	∗there	null
∗have	auxiliary verb, verb	∗to	null
∗how	null	∗v	verb
∗little	null	∗whn	null
∗modal	auxiliary verb	∗whp	null
∗n	noun		

TABLE III
LIST OF UNIVERSAL NAMES FOR GRAMMATICAL CATEGORIES USED IN THE ENGLISH GRAMMAR.

Grammatical category	Universal name	Grammatical category	Universal name
adjcomp	phrase	qpp	phrase
adjp	phrase	scmp	clause
advp	phrase	sdec	clause
ascomp	phrase	sentence	clause
aux	auxiliary verb	simp	clause
auxd	auxiliary verb	sq	clause
bep	auxiliary verb, verb	sqa	clause
ddet	null	sqb	clause
detq	null	srel	clause
dop	auxiliary verb, verb	start	null
gerund	null	subj	subject
havep	auxiliary verb, verb	swhq	clause
infinitive	phrase	thancomp	phrase
infinitivea	phrase	thatclause	clause
infinitrel	phrase	thatclausea	clause
modalp	auxiliary verb	top	top
ncomp	phrase	vp	phrase
nomhd	noun	vpa	phrase
np	noun phrase	vpb	phrase
obj	noun phrase, object	vpc	phrase
objа	noun phrase, object	whadjp	null
objb	noun phrase, object	whdet	null
pp	phrase	whnp	null
pred	phrase		

TABLE IV
AN EXAMPLE OF THE JAPANESE AND THE ENGLISH GRAMMAR RULES USED IN THIS WORK

Japanese Grammar	English Grammar
文 → 連用修飾句 + 文	SENTENCE → SDEC
文 → 用言句	SDEC → SUBJ + VP
連用修飾句 → 用言句 + 連用助詞	SUBJ → NP
連用修飾句 → 名詞句 + 連用助詞	VP → VP + PP
用言句 → 用言句 + 助動詞	SREL → *RELPRO + AUXD + VP
用言句 → 用言	PP → *PREP + NP
用言 → 語幹 + 語尾	VP → *V
名詞句 → 文 + 名詞句	NP → NP + SREL
名詞句 → 名詞	NP → *DET + *N
	AUXD → AUX

TABLE V
LIST OF POSSIBLE SYNTACTIC STRUCTURES CHARACTERIZED BY A JAPANESE PARTICLE

Particle	Syntax (Universal name)
か	clause
から	phrase
が	subject
ずつ	phrase
だけ	phrase
てから	clause
で	phrase
でから	clause
と	conjunction, phrase
に	object, phrase
の	phrase, subject
ので	clause
は	subject, object
ば	clause
へ	phrase
ほど	phrase
まで	phrase
までに	phrase
も	subject, object
や	conjunction
を	object

TABLE VI
ESTIMATION OF LIKELIHOOD OF POSSIBLE MATCHES FOR THE EXAMPLE IN FIG. 5

Possible Match	Metric
(連用修飾句125)→(PP124) (文248)→(AUXD40 VP43)	0.8
(連用修飾句125)→(PP124) (文248)→(SREL)	0.5
(連用修飾句125)→(VP43 PP124) (文248)→(AUXD40)	-10.0
(連用修飾句125)→(AUXD40 PP124) (文248)→(VP43)	-10.0
(連用修飾句125)→(VP43) (文248)→ (AUXD40 PP124)	-30.0
(連用修飾句125)→(AUXD40) (文248)→(VP43 PP124)	-30.0
(連用修飾句125)→(AUXD40 VP43) (文248)→ (PP124)	-40.0
(連用修飾句125)→(SREL) (文248)→(PP124)	-40.0

TABLE VII
A SAMPLE OF GRAMMATICAL TRANSFORMATION RULES

Grammatical transformation rule		
Japanese grammar rule		English grammar rule
(文 → 連用修飾句 + 文)	⇒	(SDEC → SUBJ + BEP + PRED)
(文 → 連用修飾句 + 文)	⇒	(SREL → *RELPRO + AUXD + VP)
(文 → 用言句)	⇒	(AUXD → BEP)(VP → *V)
(連用修飾句 → 名詞句 + 連用助詞)	⇒	(PP → *PREP + OBJ)
(連用修飾句 → 名詞句 + 連用助詞)	⇒	(SUBJ → NP)
(名詞句 → 文 + 名詞句)	⇒	(NP → NP + PP)
(名詞句 → 文 + 名詞句)	⇒	(NP → NP + SREL)
(名詞句 → 連体修飾句 + 名詞句)	⇒	(OBJ → NP)
(用言句 → 用言)	⇒	(VP → *V)