

An Unsupervised Learning Method for Associative Relationships between Verb Phrases

Kentaro Torisawa

School of Information Science,
Japan Advanced Institute of Science and Technology
1-1 Asahidai, Tatsunokuchi-machi, Nomi-gun, Ishikawa, 923-1292 JAPAN
Email: torisawa@jaist.ac.jp

Abstract

This paper describes an unsupervised learning method for associative relationships between verb phrases, which is important in developing reliable Q&A systems. Consider the situation that a user gives a query “How much petrol was *imported* by Japan from Saudi Arabia?” to a Q&A system, but the text given to the system includes only the description “X tonnes of petrol was *conveyed* to Japan from Saudi Arabia.” We think that the description is a good clue to find the answer for our query, “X tonnes.” But there is no large-scale database that provides the associative relationship between “imported” and “conveyed.” Our aim is to develop an unsupervised learning method that can obtain such an associative relationship, which we call *scenario consistency*. The method we are currently working on uses an expectation-maximization (EM) based word-clustering algorithm, and we have evaluated the effectiveness of this method using Japanese verb phrases.

1 Introduction

In natural language, there are various ways to express almost the same semantic content. In addition, one expression is *associated* with the other. These properties cause problems in NLP applications such as Q&A (See Harabagiu et al., 2000 for one of the most successful Q&A systems.). For instance, consider the following sentences.

S1 Japan *imported* X tonnes of petrol from Saudi Arabia.

S2 X tonnes of petrol was *conveyed* to Japan from Saudi Arabia.

We can say that sentences S1 and S2 are somehow associated with each other. Q&A systems, for instance, should regard S2 as a good clue to find the answer “X tonnes” for the query

“How much petrol did Japan *import* from Saudi Arabia?” The problem is that there are no sufficiently large knowledge-bases that can provide semantic relationships between expressions including *verbs* such as S1 and S2. Although there has been an attempt to provide semantic relationships between expressions with *verbs* by using *definitions of verbs* in dictionaries for human (Kaji et al., 2002), we think that such a framework has a limitation in its coverage.

The aim of this work is to develop a system that can automatically obtain associative relationships between sentences or verb phrases (VPs), as exemplified by S1 and S2, from corpora. We reformulate associative relationships with respect to the *scenario* of events, though we do not give *precise definitions* of scenarios at the current stage of our work. We have developed an unsupervised statistical method to learn the associative relationships.

We say that S1 and S2 are *scenario consistent* since they are likely to be part of the same *common-sense scenario of events*, say, an importing-something-from-foreign-countries scenario. We propose a statistical measure that can capture such intuition concerning the relationship to a certain degree. The statistical measure is estimated by looking at large-scale corpora in an unsupervised manner, i.e., the estimation process does not demand any prior semantic knowledge. As a work on a similar task, Lin et al., proposed a statistical method to extract inference rules for Q&A systems from a text corpus (Lin and Pantel, 2001). Here, the inference rules specify a relationship between expressions which is similar to our scenario consistency. The difference is that their method focuses on the extractions of *symbolic* rules for inference processes, while we treat a whole inference process. Our scenario consistency is represented in terms of a statistical measure, and it tells us not only what kind of inference is possible, as symbolic rules tells, but also when

a certain inference should be done. In other words, our statistical measure can be seen as a device representing preferences on applications of *inference rules* in the Lin’s framework.

Briefly, our method captures the scenario consistency in terms of co-occurrence probabilities of nouns and verbs in VPs. Although this approach might seem strange at first glance, we explain how the co-occurrence probabilities and *common-sense scenarios* can be related in the next section. As a statistical tool, we have developed a method to calculate the likelihood of co-occurrences by extending an existing Expectation-Maximization (EM) based word-clustering algorithm (Rooth et al., 1999; Hofmann and Puzicha, 1998). The currently available corpus is too small to directly estimate the co-occurrence probabilities for the words in VPs, so smoothing of the probabilistic distributions by means of word clustering is unavoidable. This method is described in Section 3. Section 4 gives our results from experiments using Japanese verb phrases.

2 Scenario Consistency and Co-occurrence Probabilities

This section presents our working hypothesis on scenario consistency, and provides a scoring function that gives us good candidates for scenario-consistent VPs based on the working hypothesis. To explain the scoring function, we need a probabilistic model of word co-occurrences. This model will also be described in this section.

2.1 Working Hypothesis

We say that two expressions are *scenario consistent* when they are likely to be part of the same common-sense scenario. For the sake of simplicity, we restrict the expressions to those consisting of two nouns and a verb. Such expressions may be sentences or verb phrases, but we call them *verb phrases (VPs)* uniformly throughout the rest of this paper. The procedure presented in this paper ranks a set of VPs for a given VP. The top-ranked VPs are expected to be *scenario consistent* with the given VP. For instance, consider the following examples, which are English translations of the outcomes obtained from our experimental system.

input drink beer at a restaurant.

output 1 the restaurant sells beer.

output 2 the restaurant serves beer.

output 3 enjoy beer at the restaurant.

Not many people will deny that the outputs can be part of typical scenarios of drinking beer at a restaurant. We regard all of these outputs as *scenario consistent* with the input. Generally, we say two expressions are *scenario consistent* when both represent events that belong to the same *common-sense scenario of events*. In this example, one who drinks beer at a restaurant, *usually* buys it from the restaurant, and conversely that restaurant sells beer. In other words, scenario consistency should capture the *usual* course of events, which should be simultaneously characterized as *common-sensical*. The notion of common-sense scenarios is similar to the notion of *scripts* (Schank, 1982), but we do not assume that the scenarios are described in a formal language as the scripts are.

The aim of this work is to provide an automatic method to capture the scenario consistency, although we cannot define scenario consistency precisely and we have to rely on human intuition to determine what kind of scenarios are common-sense scenarios at this stage of our work. We only make a certain working hypothesis on scenario consistency, and justify our method based on this hypothesis. The validity of the working hypothesis is supported empirically by our experiments.

To formulate our hypothesis, we need to introduce certain terminology. A verb-phrase template (VPT) is a verb phrase with empty slots that noun phrases can occupy. From a VPT and some nouns, we can *generate* normal verb phrases. By using VPTs, our working hypothesis is stated as follows.

- If there are VPTs that are likely to include the nouns in a given VP, then the VPTs and the nouns are likely to constitute VPs in a common-sense scenario that the given VP belongs to. In other words, VPs consisting of the VPTs and nouns are good candidates for the VPs that are scenario consistent with the given VP.

This hypothesis is based on the following observations.

- A few words can be good indices for a common-sense scenario; i.e., people can identify a common-sense scenario given only a few words. And words in a given VP can be good indices for a common-sense

scenario if the VP sufficiently identifies the scenario.

- Expressions in a common-sense scenario are likely to have lexical coherence; i.e., if a set of expressions describe the events in a common-sense scenario, they tend to share common words. In addition, words that can be good indices for the scenario tend to be shared by many expressions in the scenario. If a given VP identifies the scenario sufficiently, the words in the VP are simultaneously shared by the expressions that become part of the scenario.

Recall the input and outputs in the beer drinking example. We could say that they are part of a common-sense scenario. In addition, people can sufficiently identify the scenario given the words “beer,” “restaurant” and “drink.” All the expressions in the examples share the words “beer” and “restaurant.” Thus, the drinking beer example includes expressions that follows the above observations.

An important point is that our working hypothesis enables us to view the notion of scenario consistency in terms of the likelihood of word sharing by VPs. In other words, the hypothesis opens the way to formulate the likelihood that two VPs are scenario consistent in terms of the co-occurrence probabilities among the words. We will formulate a scoring function that can tell us the likelihood that two VPs are scenario consistent, in terms of the co-occurrence probabilities of the verbs and nouns used. In addition, we introduce a class-based probabilistic model to avoid data sparseness in the estimation of co-occurrence probabilities.

Finally, we will make a remark on the limitation of our current framework. Note that outputs of our procedure are restricted to the VPs that share *two* nouns with a given VP, as in the drinking beer example. In other words, only the VPs sharing *two* nouns with a given VP are regarded as candidates of scenario consistent VPs. However, there can be scenario consistent VPs that share *less* than two nouns with a given VP. In addition, such VPs may contain additional nouns that do not appear in the given VP. For instance, one may say that “pay money for beer” should be scenario consistent with “drink beer at a restaurant,” although the former VP shares only “beer” with the latter VP and it contains additional noun “money.” If we can develop an automatic method to recognize such a type of scenario consistent VPs, it will enable

us to automatically obtain common-sense scenarios that are more similar to Schank’s scripts. But such a method is more difficult to realize than our current method and it is our future work. Note that while our current method basically determines only one verb (and argument positions which two nouns fill in), such a method must choose one verb and additional nouns (and their arrangement). In the pay-money-for-beer example, the method has to choose not only the verb “pay” but also the noun “money.” We found that choice of such additional nouns is a particularly difficult task at the current stage. It is a major obstacle for realizing the automatic method.

2.2 Probabilistic Model

Before going on to our scoring function for scenario consistency, we need to describe a probabilistic model for the co-occurrence of words in VPs. The scoring function is defined in terms of the model parameters. The model is *class based*, and we introduce the notion of semantic word classes in the model. The probabilistic distributions based on the given model can be estimated through an unsupervised learning method called the EM algorithm, which is described in the next section.

We assume that all the VPs can be denoted by a tuple $\langle v, rel_1, rel_2, n_1, n_2 \rangle$. Here v is a verb, and n_1 and n_2 are nouns. rel_1 and rel_2 are *co-occurrence relations*, which can be a *syntactic subject*, an *object*, or (as in the rest of this paper) a *postposition* since verb arguments in Japanese are usually marked by postpositions. In short, the tuple denotes the following VP in the case of Japanese, where P denotes a postposition.

n_1	rel_1	n_2	rel_2	v
noun	P	noun	P	verb

The occurrence probability of a VP $\langle v, rel_1, rel_2, n_1, n_2 \rangle$ can then be denoted by $P(\langle v, rel_1, rel_2, n_1, n_2 \rangle)$. The problem is that reliable values of $P(\langle v, rel_1, rel_2, n_1, n_2 \rangle)$ cannot be estimated by simply counting words in a text corpus since any currently available corpus is too small. To obtain reliable probability values, we therefore introduced smoothing of the probabilistic distributions by means of word clustering. The smoothed distributions are represented by the following formula.

$$P(\langle v, rel_1, rel_2, n_1, n_2 \rangle) =_{def} \sum_{a,b \in A} P(\langle v, rel_1, rel_2 \rangle | a, b) P(n_1 | a) P(n_2 | b) P(a, b)$$

This formula means that the occurrence probabilities are determined from the other forms of probabilities, which are shown in the right side

of the formula. We call such probabilities *parameters*. In the parameters, a and b denote semantic classes of noun n_1 and n_2 , respectively. We assume that A is a set of k symbols. We call a member of A a *class symbol* and A is denoted by $\{a_1, a_2, \dots, a_k\}$ if necessary. Roughly speaking, in this formula the occurrences of the nouns n_1 and n_2 are *replaced* by occurrences of the semantic classes a and b that the nouns belong to. The number of semantic classes is set to a value smaller than the total number of nouns. This enables us to obtain smoothing effect.

More precisely, we can regard the parameter $P(n|a)$ as a representation of a word class when class symbol a is fixed. We can easily derive the probability $P(a|n)$ from $P(n|a)$. $P(a|n)$ can be interpreted as the probability that n 's appearance is used as a word in the class a . Note that we can compute all values of $P(a|n)$ for each class denoted by a , and that this distribution can be considered the distribution of distinct usages of the word n . From here, we identify the distribution $\{P(n|a)|n \text{ is a noun}\}$ by using the class a , and call a collection of all the classes in A a *word classification*.

A tuple $\langle v, rel_1, rel_2 \rangle$ denotes a *template* of the verb phrase that is headed by the verb v and has the two argument positions denoted by $\langle v, rel_1 \rangle$ and $\langle v, rel_2 \rangle$. We call the tuples in the form of $\langle v, rel_1, rel_2 \rangle$ *verb-phrase templates (VPT)*. The parameters in the form $P(\langle v, rel_1, rel_2 \rangle | a, b)$ then denote the probability that, given two classes a and b , the argument positions $\langle v, rel_1 \rangle$ and $\langle v, rel_2 \rangle$ are filled by two nouns belonging to classes a and b , respectively. On the other hand, the other type of parameter, $P(a, b)$, expresses the probabilities that classes a and b co-occur in the same verb phrases.

2.3 Scoring Function for Scenario-consistent Verb Phrases

Thus, we presented a class-based probabilistic model for the occurrence probabilities of VPs. Our scoring function for scenario consistency is defined in terms of the model parameters. This function is defined as follows.

$$S(\langle v, rel_1, rel_2, n_1, n_2 \rangle, \langle v', rel'_1, rel'_1, n_1, n_2 \rangle) \\ =_{def} \sum_{a, b \in A} \{P(\langle v, rel_1, rel_2 \rangle | a, b) P(n_1 | a) P(n_2 | b) \\ P(a, b | \langle v', rel'_1, rel'_2 \rangle)\}$$

We assume that a given VP is represented by the tuple $\langle v', rel'_1, rel'_1, n_1, n_2 \rangle$ and the score S is used to assess whether the other VP $\langle v, rel_1, rel_2, n_1, n_2 \rangle$ is scenario consistent with $\langle v', rel'_1, rel'_1, n_1, n_2 \rangle$.

Note that the scoring function is quite similar to the occurrence probability of the VP $\langle v, rel_1, rel_2, n_1, n_2 \rangle$, consisting of a VPT and nouns in a given VP. This means that the scoring function tends to give a high ranking to a VPT that frequently co-occurs with the nouns in a given VP. The difference between the scoring function and the occurrence probability $P(\langle v, rel_1, rel_2, n_1, n_2 \rangle)$ is that the parameter $P(a, b)$ is replaced with a new term $P(a, b | \langle v', rel'_1, rel'_2 \rangle)$, which represents the probability that the VPT in the given VP is filled by nouns belonging to classes a and b . Note that this term can be estimated from the parameters $P(a, b)$ and $P(\langle v', rel'_1, rel'_2 \rangle | a, b)$. Intuitively, through this modification, the scoring function can reflect the *biases* posed by the VPT $\langle v', rel'_1, rel'_2 \rangle$ in the given VP. Consider the following four sentences.

1. I drove a car.
2. I bought a car.
3. I got into a car.
4. I purchased a car.

We can say that the car is treated as a *vehicle* in sentence 1, while sentence 2 focuses on the aspect of the car as *merchandise*. In our view, the scoring function for the scenario consistency should regard sentence 3 as a better candidate to be a scenario-consistent VP for sentence 1 than sentence 4, while sentence 2 should receive stronger preference regarding sentence 4 than regarding sentence 3. Our function S can reflect such preferences, at least occasionally. Assume that *person*, *merchandise*, and *vehicle* are class symbols corresponding to the word classes of persons, merchandise, and vehicles, respectively. In an ideal situation, we can expect the following inequalities to hold.

$$P(\textit{person}, \textit{vehicle} | \langle \textit{drove}, \textit{subj}, \textit{obj} \rangle) \gg \\ P(\textit{person}, \textit{merchandise} | \langle \textit{drove}, \textit{subj}, \textit{obj} \rangle) \\ P(\textit{person}, \textit{vehicle} | \langle \textit{bought}, \textit{subj}, \textit{obj} \rangle) \ll \\ P(\textit{person}, \textit{merchandise} | \langle \textit{bought}, \textit{subj}, \textit{obj} \rangle) \\ P(\langle \textit{got-into}, \textit{subj}, \textit{obj} \rangle | \textit{person}, \textit{vehicle}) \gg \\ P(\langle \textit{purchased}, \textit{subj}, \textit{obj} \rangle | \textit{person}, \textit{vehicle}) \\ P(\langle \textit{got-into}, \textit{subj}, \textit{obj} \rangle | \textit{person}, \textit{merchandise}) \ll \\ P(\langle \textit{purchased}, \textit{subj}, \textit{obj} \rangle | \textit{person}, \textit{merchandise})$$

These inequalities are considered in S since the terms in the inequalities are multiplied in the score. This means that the ranking for sentence 1 is likely to indicate a strong preference regarding sentence 3, while sentence 4 tends to be highly ranked regarding sentence 2.

$$\begin{aligned}
& \frac{P_j(a|\langle v, rel, n \rangle)}{\sum_{a' \in A} \frac{P_j(a)P_j(\langle v, rel \rangle|a)P_j(n|a)}{P_j(a')P_j(\langle v, rel \rangle|a')P_j(n|a')}} , & \frac{P_j(a, b|\langle v, rel_1, rel_2, n_1, n_2 \rangle)}{\sum_{a', b' \in A} \frac{P_j(\langle v, rel_1, rel_2 \rangle|a, b)P_j(n_1|a)P_j(n_2|b)P_j(a, b)}{P_j(\langle v, rel_1, rel_2 \rangle|a', b')P_j(n_1|a')P_j(n_2|b')P_j(a', b')}} \\
& P_{j+1}(a) = \frac{1}{Z_{class}} \sum_{\langle v, rel, n \rangle \in set(L_1)} P_j(a|\langle v, rel, n \rangle) f(\langle v, rel, n \rangle, L_1), \\
& P_{j+1}(a, b) = \frac{1}{Z_{joint}} \sum_{\langle v, rel_1, rel_2, n_1, n_2 \rangle \in set(L_2)} P_j(a, b|\langle v, rel_1, rel_2, n_1, n_2 \rangle) f(\langle v, rel_1, rel_2, n_1, n_2 \rangle, L_2), \\
& P_{j+1}(n|a) = \frac{1}{Z_N} \left\{ \sum_{\langle v, rel, n \rangle \in set(L_1)} (P_j(a|\langle v, rel, n \rangle) f(\langle v, rel, n \rangle, L_1)) \right. \\
& \quad + \sum_{\langle v, rel_1, rel_2, n, n' \rangle \in set(L_2), b \in A} (P_j(a, b|\langle v, rel_1, rel_2, n, n' \rangle) f(\langle v, rel_1, rel_2, n, n' \rangle, L_2)) \\
& \quad \left. + \sum_{\langle v, rel_1, rel_2, n', n \rangle \in set(L_2), b \in A} (P_j(b, a|\langle v, rel_1, rel_2, n', n \rangle) f(\langle v, rel_1, rel_2, n', n \rangle, L_2)) \right\}, \\
& P_{j+1}(\langle v, rel \rangle|a) = \frac{1}{Z_{VP1}} \sum_{\langle v, rel, n \rangle \in set(L_1)} (P_j(a|\langle v, rel, n \rangle) f(\langle v, rel, n \rangle, L_1)), \\
& P_{j+1}(\langle v, rel_1, rel_2 \rangle|a, b) = \\
& \quad \frac{1}{Z_{VP2}} \sum_{\langle v, rel_1, rel_2, n_1, n_2 \rangle \in set(L_2)} (P_j(a, b|\langle v, rel_1, rel_2, n_1, n_2 \rangle) f(\langle v, rel_1, rel_2, n_1, n_2 \rangle, L_2))
\end{aligned}$$

The factors Z are normalizing factors such that the sum of each type of probability equals 1. $f(item, L)$ denotes the frequency that $item$ is observed in list L , and $set(L)$ is the set of all the elements in list L .

Figure 1: Update functions for a triple model.

3 Parameter Estimation with the EM algorithm

To obtain the parameter values needed to compute our scoring function for scenario consistency, we developed an EM-based word-clustering method that is described in this section. Note that the method is unsupervised in the sense that the semantic classes of words are obtained without any manually provided clues.

We have defined a probabilistic model that dominates the co-occurrence between two nouns and VPTs. Basically, the EM-based method is derived from the defined model. However, in practice, we need another probabilistic model, which was used in the original EM-based word-clustering method (Rooth et al., 1999; Hofmann and Puzicha, 1998). The original method is robust in that it can obtain probabilistic distributions from verb phrase *fragments*. In Japanese, verb arguments are frequently omitted, and the frequencies of well-formed VPs are relatively small. Thus, to enlarge the training samples, we used the original probabilistic model which is given by the following formula.

$$\begin{aligned}
& P(\langle v, rel, n \rangle) \\
& =_{def} \sum_{a \in A} P(\langle v, rel \rangle|a)P(n|a)P(a)
\end{aligned}$$

Here, a denotes a semantic class of a noun n and $\langle v, rel \rangle$ specifies an argument position consisting of a verb v and a co-occurrence relation rel .

The training samples for our EM-based estimation procedure must therefore include two types of data, each corresponding to

one of the probabilistic models we used. We denote the first type of data as $L_1 = \langle \langle v_0, rel_0, n_0 \rangle, \langle v_1, rel_1, n_1 \rangle, \dots, \langle v_m, rel_m, n_m \rangle \rangle$ and the second as $L_2 = \langle \langle v_0, rel_0^1, rel_0^2, n_0^1, n_0^2 \rangle, \dots, \langle v_k, rel_k^1, rel_k^2, n_k^1, n_k^2 \rangle \rangle$. While L_1 is a list of VPs with only one noun and one verb, L_2 is a list of VPs consisting of two nouns and one verb. We call the items in L_1 *co-occurrence pairs* and the items in L_2 *co-occurrence triples*.

From lists L_1 and L_2 , the EM algorithm estimates the probabilities in an iterative manner. We denote the parameters computed at the j -th iteration step as $P_j(\cdot)$. The computation at the $(j+1)$ -th iteration proceeds as follows. First, we compute $P_j(a|\langle v, rel \rangle, n)$ and $P_j(a, b|\langle v, rel_1, rel_2, n_1, n_2 \rangle)$ using the formulae in Fig. 1. The new parameters denoted by $P_{j+1}(\cdot)$ are then calculated using the update functions listed in the same figure. These functions were derived according to standard derivation steps within the EM framework. Note that the iteration steps are repeated until a given number of the steps are executed. The parameters computed at the final step are regarded as the estimated parameters.

Finally, we will make some remarks concerning the extension of this framework. First, the EM algorithm has to determine the initial values of the parameters which are denoted as $P_0(\cdot)$. Although the initial parameters were often determined randomly in previous works, we developed a method to determine the initial values by means of hard clustering (Brown

- CLASS 1
 - sake (drink with alcohol) 0.725
 - ocha (green tea) 0.631
 - biiru (beer) 0.541
 - nihonshu (Japanese sake) 0.525
- CLASS 2
 - kukkii (cookie) 0.692
 - udon (noodle) 0.644
 - raamen (noodle) 0.623
 - karee (curry) 0.619
- CLASS 3
 - furansu (France) 0.624
 - doitsu (Germany) 0.601
 - eikoku (England) 0.575
 - igirisu (England) 0.246
- CLASS 4
 - tai (Thailand) 0.722
 - mareisia (Malaysia) 0.721
 - indonesia (Indonesia) 0.529
 - firipin (Philippine) 0.439

Figure 2: Examples of word classes.

	<i>Top 10</i>	<i>Top 5</i>
Avg	473/909 (52.0%)	239/415 (57.6%)
Agreed	300/909 (33.0%)	165/415 (39.8%)

Table 1: Accuracy of scenario consistency.

et al., 1992). Second, we had to introduce an approximation technique to execute the above procedure since its time/space complexity exceeded the practical capability of our workstations. (See Torisawa, 2001 for more details.) Third, we observed that the above probabilistic model suffers from data sparseness since the number of parameters becomes quite large when we try to obtain probabilistic distributions for fine word classifications. Simultaneously, we found that a fine grained word classification is crucial for our task. To solve this dilemma, we developed yet another probabilistic model that combines two of the above models. In the new model, the co-occurrence probabilities of a VPT and two nouns are dominated by *two* distinct word classifications, one corresponding to a fine-grained word classification and one to a coarse-grained classification. This modification suppresses the effect of data sparseness to a certain degree without sacrificing the fine-grained word classification. We used this extended model in the experiments we discuss below.

4 Experiments

We obtained co-occurrence triples/pairs from Japanese newspaper articles (nine years of the Nikkei Shinbun and five years of the Mainichi Shinbun) by using an existing parser

(Kanayama et al., 2000). The total frequency of the co-occurrence pairs was 1.08×10^8 , while that of the co-occurrence triples was 1.61×10^7 . They included the most frequent 18,360 words, along with 25,473 argument positions and 19,704 VPTs.

We obtained the probabilistic distribution based on a probabilistic model consisting of *two* word classifications with 500 classes and 2,500 classes. Some of the classes we obtained, taken from the classification with 2500 classes in our distribution, are presented in Fig. 2 along with the parameter $P(a|w)$ where w is a word and a is a class symbol. The words are the top four in terms of $P(a|w)$ in each class.

Next, we randomly selected 100 unseen VPs from Japanese newspaper articles. They are restricted to the ones including only the words considered in our parameter estimation. Then, we applied our scoring functions to the VPs to rank candidates of scenario consistent VPs. We asked three human subjects to assign the following labels to the top 10 in the ranking for each of the given VPs.

Same The candidate VP expresses the same semantic content as that of the given VP. The difference from the given VP is limited to orthographical differences of the verbs¹ and differences of the postpositions.

Consistent The candidate VP has a verb different from that of the given VP, and it is scenario consistent to the given VP.

Others

Fig. 3 shows an example of the labeling given by one of the subjects. The accuracy of our experimental system is presented in Table 1. We regarded the VPs with the labels Same and Consistent as scenario-consistent VPs. *Avg* refers to a simple average of the ratio of scenario-consistent VPs over the subjects. This ratio ignores how the labeling by each subject coincided with others'. *Agreed* shows the ratio of the VPs which all the subjects judged scenario consistent with the input VPs. *Top 10* and *Top 5* shows the results of the labeling to the top 5 candidates and the top 10 candidates for each of the given VPs. Note that our experimental system produced the ranking that might contain the input VP itself. After excluding the input VPs from the ranking results, the total

¹Japanese has Chinese characters and two different types of alphabets. Sometime, the same words is written differently by using distinct types of characters.

Input 1	chichi ga kekkaku de taoreru. (A father is down with tuberculosis)	
1st	the same as the input VP.	
2nd	chichi ga kekkaku ni naru. (The father becomes tuberculosis.)	C
3rd	chichi wo kekkaku de ushinau. (Lose the father by tuberculosis.)	C
4th	chichi ga kekkaku ni taoreru. (The father is down with tuberculosis)	S
5th	chichi ga kekkaku kara ukeru. (not understandable)	O
6th	chichi ga kekkaku de teikyou-suru. (not understandable)	O
7th	chichi ga kekkaku de shibou-suru (The father dies because of tuberculosis.)	C
Input 2	endaka de kyousouryoku ga teika-suru. (Competitive power (CP) is reduced because of Expensive Yen (EY).)	
1st	the same as the input VP.	
2nd	endaka de kyousouryoku wo ushinau. (lose CP because of EY.)	C
3rd	endaka ga kyousouryoku wo yowameru. (EY weakens CP.)	C
4th	endaka de kyousouryoku ga ochiru. (CP goes down because of EY.)	C

C stands for Consistent, S for Same, and O for Others.

Figure 3: Examples of ranking of VPs.

number of the top 10 candidates became 909, while the number of the top 5 candidates was 415. We used these numbers as the total numbers of produced candidates.

The difference between *Avg* and *Agreed* shows the difficulty of the task. Note that the human subject who regarded the largest number of VPs as scenario consistent assigned Same or Consistent labels to 62.7% of all the VPs in the top 5 candidates. As for the top 10 candidates, the most generous human subject gave Same or Consistent to 55.7% of the VPs. These mean that even the human subjects were inconsistent for more than 20% of the candidates. Considering such difficulty, we think that 40% accuracy of *Agreed* for the Top 5 rankings is reasonable. Note that Same labels are assigned to only 22% of the *correct* candidates.

5 Future Work and Concluding Remarks

We have proposed a notion of *scenario consistency*, which is a type of associative relationship between VPs or sentences, and have presented a statistical method that can produce VPs that are *scenario consistent* to a given VP, with a certain accuracy. For instance, our experimental system could produce “A restaurant serves beer.” as a VP that is scenario consistent with

“drink beer at a restaurant”. We expect scenario consistency to be important in finding reliable answers in Q&A systems, and believe that our method will prove useful for developing such systems. Our method is unsupervised in the sense that it does not require any prior semantic/pragmatic knowledge, and our future work will include the application of our method to a larger text corpus.

References

- Peter F. Brown, Vincent J. Della Pietra, Peter V. deSouza, Jennifer C. Lai, and Robert L. Mercer. 1992. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):31–40.
- Sanda M. Harabagiu, Marius A. Pasca, and Steven J. Maiorano. 2000. Experiments with open-domain textual question answering. In *Proceedings of COLING 2000*, pages 292–298.
- Thomas Hofmann and Jan Puzicha. 1998. Unsupervised learning from dyadic data. Tr-98-042, International Computer Science Institute, Berkley, CA.
- Nobuhiro Kaji, Daisuke Kawahara, Sadao Kurohashi, and Satoshi Sato. 2002. Kokugo jiten to kopasu wo mochiita yougen no iikae kisoku no gakusyuu (learning of paraphrasing rules for predicates by using dictionaries and corpora). In *Proceedings of The Eighth Annual Meeting of The Association for Natural Language Processing*, pages 331–334. in Japanese.
- Hiroshi Kanayama, Kentaro Torisawa, Yutaka Mitsuishi, and Jun’ichi Tsujii. 2000. A hybrid Japanese parser with hand-crafted grammar and statistics. In *Proceedings of COLING 2000*, pages 411–417.
- Dekang Lin and Patrick Pantel. 2001. Discovery of inference rules for question answering. *Journal of Natural Language Engineering*. in print.
- Mats Rooth, Stefan Riezler, Detlef Prescher, Glenn Carroll, and Franz Beil. 1999. Inducing a semantically annotated lexicon via em-based clustering. In *Proceedings of 37th Annual Meeting of the ACL*, pages 104–111.
- Roger Schank. 1982. *Dynamic Memory*. Cambridge University Press.
- Kentaro Torisawa. 2001. A nearly unsupervised learning method for automatic paraphrasing of Japanese noun phrases. In *Proceedings of Workshop on Automatic Paraphrasing: Theories and Applications*, pages 63–72, Tokyo, Japan.