A Probabilistic Model for Measuring Grammaticality and Similarity of Automatically Generated Paraphrases of Predicate Phrases

Atsushi Fujita   **Satoshi Sato**
Graduate School of Engineering, Nagoya University
{fujita,ssato}@nuee.nagoya-u.ac.jp

Abstract

The most critical issue in generating and recognizing paraphrases is development of wide-coverage paraphrase knowledge. Previous work on paraphrase acquisition has collected lexicalized pairs of expressions; however, the results do not ensure full coverage of the various paraphrase phenomena. This paper focuses on productive paraphrases realized by general transformation patterns, and addresses the issues in generating instances of phrasal paraphrases with those patterns. Our probabilistic model computes how two phrases are likely to be correct paraphrases. The model consists of two components: (i) a structured N-gram language model that ensures grammaticality and (ii) a distributional similarity measure for estimating semantic equivalence and substitutability.

1 Introduction

In many languages, a concept can be expressed with several different linguistic expressions. Handling such synonymous expressions in a given language, i.e., paraphrases, is one of the key issues in a broad range of natural language processing tasks. For example, the technology for identifying paraphrases would play an important role in aggregating the wealth of uninhibited opinions about products and services that are available on the Web, from both the consumers and producers viewpoint. On the other hand, whenever we draw up a document, we always seek the most appropriate expression for conveying our ideas. In such a situation, a system that generates and proposes alternative expressions would be extremely beneficial.

Most of previous work on generating and recognizing paraphrases has been dedicated to developing context-free paraphrase knowledge. It is typically represented with pairs of fragmentary expressions that satisfy the following conditions:

Condition 1. Semantically equivalent

Condition 2. Substitutable in some context

The most critical issue in developing such knowledge is ensuring the coverage of the paraphrase phenomena. To attain this coverage, we have proposed a strategy for dividing paraphrase phenomena into the following two classes (Fujita et al., 2007):

(1) **Non-productive (idiosyncratic) paraphrases**
    a. burst into tears ⇔ cried
    b. comfort ⇔ console
             (Barzilay and McKeown, 2001)

(2) **Productive paraphrases**
    a. be in our favor ⇔ be favorable to us
    b. show a sharp decrease ⇔ decrease sharply
             (Fujita et al., 2007)

Typical examples of **non-productive paraphrases** are lexical paraphrases such as those shown in (1) and idiomatic paraphrases of literal phrases (e.g., "kick the bucket" ⇔ "die"). Knowledge of this class of paraphrases should be stored statically, because they cannot be represented with abstract patterns. On the other hand, a **productive paraphrase** is one having a degree of regularity, as exhibited by the examples in (2). It is therefore reasonable to represent them with a set of general patterns such as those shown in (3). This attains a higher coverage, while keeping the knowledge manageable.

(3)   a. N_1 V N_2 ⇔ N_1's V-*ing* of N_2
    b. N_1 V N_2 ⇔ N_2 be V-*en* by N_1
             (Harris, 1957)

Various methods have been proposed to acquire paraphrase knowledge (these are reviewed in Section 2.1) where pairs of **existing expres-**

sions are collected from the given corpus, taking the above two conditions into account. On the other hand, another issue arises when **paraphrase knowledge is generated** from the patterns for productive paraphrases such as shown in (3) by instantiating variables with specific words, namely,

Condition 3. Both expressions are grammatical

This paper proposes a probabilistic model for computing how likely a given pair of expressions satisfy the aforementioned three conditions. In particular, we focus on the post-generation assessment of automatically generated productive paraphrases of predicate phrases in Japanese.

In the next section, we review previous approaches and models. The proposed probabilistic model is then presented in Section 3, where the grammaticality factor and similarity factor are derived from a conditional probability. In Section 4, the settings for and results of an empirical experiment are detailed. Finally, Section 5 summarizes this paper.

2 Previous work

2.1 Acquiring paraphrase knowledge

The task of automatically acquiring paraphrase knowledge is drawing the attention of an increasing number of researchers. They are tackling the problem of how precisely paraphrase knowledge can be acquired, although they have tended to notice that it is hard to acquire paraphrase knowledge that ensures full coverage of the various paraphrase phenomena from existing text corpora alone. To date, two streams of research have evolved: one acquires paraphrase knowledge from parallel/comparable corpora, while the other uses the regular corpus.

Several alignment techniques have been proposed to acquire paraphrase knowledge from parallel/comparable corpora, imitating the techniques devised for machine translation. Multiple translations of the same text (Barzilay and McKeown, 2001), corresponding articles from multiple news sources (Barzilay and Lee, 2003; Quirk et al., 2004; Dolan et al., 2004), and bilingual corpus (Bannard and Callison-Burch, 2005) have been utilized. Unfortunately, this approach produces only a low coverage because the size of the parallel/comparable corpora is limited.

In the second stream, i.e., paraphrase acquisition from the regular corpus, the distributional hypothesis (Harris, 1968) has been adopted. The similarity of two expressions, computed from this hypothesis, is called distributional similarity. The essence of this measure is summarized as follows:

Feature representation: to compute the similarity, given expressions are first mapped to certain feature representations. Expressions that co-occur with the given expression, such as adjacent words (Barzilay and McKeown, 2001; Lin and Pantel, 2001), and modifiers/modifiees (Yamamoto, 2002; Weeds et al., 2005), have so far been examined.

Feature weighting: to precisely compute the similarity, the weight for each feature is adjusted. Point-wise mutual information (Lin, 1998) and Relative Feature Focus (Geffet and Dagan, 2004) are well-known examples.

Feature comparison measures: to convert two feature sets into a scalar value, several measures have been proposed, such as cosine, Lin's measure (Lin, 1998), Kullback-Leibler (KL) divergence and its variants.

While most researchers extract fully-lexicalized pairs of words or word sequences only, two algorithms collect template-like knowledge using dependency parsers. DIRT (Lin and Pantel, 2001) collects pairs of paths in dependency parses that connect two nominal entities. TEASE (Szpektor et al., 2004) discovers dependency sub-parses from the Web, based on sets of representative entities for a given lexical item. The output of these systems contains the variable slots as shown in (4).

(4)   a. X wrote $Y \Leftrightarrow X$ is the author of Y
      b. X solves $Y \Leftrightarrow X$ deals with Y
                                   (Lin and Pantel, 2001)

The knowledge in (4) falls between that in (1), which is fully lexicalized, and that in (3), which is almost fully abstracted. As a way of enriching such a template-like knowledge, Pantel et al. (2007) proposed the notion of inferential selectional preference and collected expressions that would fill those slots.

As mentioned in Section 1, the aim of the studies reviewed here is to collect paraphrase knowledge. Thus, they need not to take the grammaticality of expressions into account.

2.2 Generating paraphrase instances

Representing productive paraphrases with a set of general patterns makes them maintainable and attains a higher coverage of the paraphrase phenomena. From the transformation grammar (Har-

ris, 1957), this approach has been adopted by many researchers (Mel'čuk and Polguère, 1987; Jacquemin, 1999; Fujita et al., 2007). An important issue arises when such a pattern is used to generate instances of paraphrases by replacing its variables with specific words. This involves assessing the grammaticality of two expressions in addition to their semantic equivalence and substitutability.

As a post-generation assessment of automatically generated productive paraphrases, we have applied distributional similarity measures (Fujita and Sato, 2008). Our findings from a series of empirical experiments are summarized as follows:

- Search engines are useful for retrieving the contextual features of predicate phrases despite some limitations (Kilgarriff, 2007).
- Distributional similarity measures produce a tolerable level of performance.

The grammaticality of a phrase, however, is merely assessed by issuing the phrase as a query to a commercial search engine. Although a more frequent expression is more grammatical, the length bias should also be considered in the assessment.

Quirk et al. (2004) built a paraphrase generation model from a monolingual comparable corpus based on a statistical machine translation framework, where the language model assesses the grammaticality of the translations, i.e., generated expressions. The translation model, however, is not suitable for generating productive paraphrases, because it learns word alignments at the surface level. To cover all of the productive paraphrases, we require an non-real comparable corpus in which all instances of productive paraphrases have a chance of being aligned. Furthermore, as the translation model optimizes the word alignment at the sentence level, the substitutability of the aligned word sequences cannot be explicitly guaranteed.

2.3 Existing measures for paraphrases

To date, no model has been established that takes into account all of the three aforementioned conditions. With the ultimate aim of building an ideal model, this section overviews the characteristics and drawbacks of the four existing measures.

Lin's measure

Lin (1998) proposed a symmetrical measure:

$$Par_{Lin}(s \Leftrightarrow t) = \frac{\sum_{f \in F_s \cap F_t} (w(s, f) + w(t, f))}{\sum_{f \in F_s} w(s, f) + \sum_{f \in F_t} w(t, f)},$$

where F_s and F_t denote sets of features with positive weights for words s and t, respectively.

Although this measure has been widely cited and has so far exhibited good performance, its symmetry seems unnatural. Moreover, it may not work well for dealing with general predicate phrases because it is hard to enumerate all phrases to determine the weights of features $w(\cdot, f)$. We thus simply adopted the co-occurrence frequency of the phrase and the feature as in (Fujita and Sato, 2008).

Skew divergence

The skew divergence, a variant of KL divergence, was proposed in (Lee, 1999) based on an insight: the substitutability of one word for another need not be symmetrical. The divergence is given by the following formula:

$$d_{skew}(t, s) = D(P_s \| \alpha P_t + (1 - \alpha) P_s),$$

where P_s and P_t are the probability distributions of features for the given original and substituted words s and t, respectively. $0 \le \alpha \le 1$ is a parameter for approximating KL divergence D. The score can be recast into a similarity score via, for example, the following function (Fujita and Sato, 2008):

$$Par_{skew}(s \Rightarrow t) = \exp(-d_{skew}(t, s)).$$

This measure offers an advantage: the weight for each feature is determined theoretically. However, the optimization of α is difficult because it varies according to the task and even the data size (confidence of probability distributions).

Translation-based conditional probability

Bannard and Callison-Burch (2005) proposed a probabilistic model for acquiring phrasal paraphrases[1]. The likelihood of t as a paraphrase of the given phrase s is defined as follows:

$$P(t|s) = \sum_{f \in tr(s) \cap tr(t)} P(t|f) P(f|s),$$

where $tr(e)$ stands for a set of foreign language phrases that are aligned with e in the given parallel corpus. Parameters $P(t|f)$ and $P(f|s)$ are also estimated using the given parallel corpus. A large-scale parallel corpus may enable us to precisely acquire a large amount of paraphrase knowledge. It

---

[1] In their definition, the term "phrase" is a sequence of words, while in this paper it designates the subtrees governed by predicates (Fujita et al., 2007).

is not feasible, however, to build (or obtain) a parallel corpus in which all the instances of productive paraphrases are translated to the same expression in the other side of language.

3 Proposed probabilistic model

3.1 Formulation with conditional probability

Recall that our aim is to establish a measure that computes the likelihood of a given pair of automatically generated predicate phrases satisfying the following three conditions:

Condition 1. Semantically equivalent
Condition 2. Substitutable in some context
Condition 3. Both expressions are grammatical

Based on the characteristics of the existing measures reviewed in Section 2.3, we propose a probabilistic model. Let s and t be the source and target predicate phrase, respectively. Assuming that s is grammatical, the degree to which the above conditions are satisfied is formalized as a conditional probability $P(t|s)$, as in (Bannard and Callison-Burch, 2005). Then, assuming that s and t are paradigmatic (i.e., paraphrases) and thus do not co-occur, the proposed model is derived as follows:

$$
\begin{aligned}
P(t|s) &= \sum_{f \in F} P(t|f)P(f|s) \\
&= \sum_{f \in F} \frac{P(f|t)P(t)}{P(f)} P(f|s) \\
&= P(t) \sum_{f \in F} \frac{P(f|t)P(f|s)}{P(f)},
\end{aligned}
$$

where F denotes a set of features. The first factor $P(t)$ is called the **grammaticality factor** because it quantifies the degree to which condition 3 is satisfied, except that we assume that the given s is grammatical. The second factor $\sum_{f \in F} \frac{P(f|t)P(f|s)}{P(f)}$ ($Sim(s,t)$, hereafter), on the other hand, is called the **similarity factor** because it approximates the degree to which conditions 1 and 2 are satisfied by summing up the overlap of the features of two expressions s and t.

The characteristics and advantages of the proposed model are summarized as follows:

1) Asymmetric.
2) Grammaticality is assessed by $P(t)$.
3) No heuristic is introduced. As the skew divergence, the weight of the features can be simply estimated as conditional probabilities $P(f|t)$ and $P(f|s)$ and marginal probability $P(f)$.

4) There is no need to enumerate all the phrases. s and t are merely the given conditions.

The following subsections describe each factor.

3.2 Grammaticality factor

The factor $P(t)$ quantifies how the phrase t is grammatical using statistical language model.

Unlike English, in Japanese, predicates such as verbs and adjectives do not necessarily determine the order of their arguments, although they have some preference. For example, both of the two sentences in (5) are grammatical.

(5)   a. *kare-wa* **pasuta-o hashi-de**   *taberu.*
       he-TOP   pasta-ACC chopsticks-IMP to eat
       He eats pasta with chopsticks.

      b. *kare-wa* **hashi-de**   **pasuta-o** *taberu.*
       he-TOP   chopsticks-IMP pasta-ACC to eat
       He eats pasta with chopsticks.

This motivates us to use structured N-gram language models (Habash, 2004). Given a phrase t, its grammaticality $P(t)$ is formulated as follows, assuming a $(N-1)$-th order Markov process for generating its dependency structure $T(t)$:

$$
P(t) = \left[\prod_{i=1\ldots|T(t)|} P_d\big(c_i | d_i^1, d_i^2, \ldots, d_i^{N-1}\big) \right]^{1/|T(t)|},
$$

where $|T(t)|$ stands for the number of nodes in $T(t)$. To ignore the length bias of the target phrase, a normalization factor $1/|T(t)|$ is introduced. d_i^j denotes the direct ancestor node of the i-th node c_i, where j is the distance from c_i; for example, d_i^1 and d_i^2 are the parent and grandparent nodes of c_i, respectively.

Then, a concrete definition of the nodes in the dependency structure is given. Widely-used Japanese dependency parsers such as CaboCha[2] and KNP[3] consider a sequence of words as a node called a "*bunsetsu*" that consists of at least one content word followed by a sequence of function words if any. The hyphenated word sequences in (6) exemplify those nodes.

(6)   *kitto*   *kare-ha*   *kyou-no*
      surely   he-TOP   today-GEN
      kaigi-ni-ha   *ko-nai-daro-u.*
      meeting-DAT-TOP   to come-NEG-must
      He will surely not come to today's meeting.

As *bunsetsu* can be quite long, involving more than ten words, regarding it as a node makes the model complex. Therefore, we compare the

---
[2] http://chasen.org/~taku/software/cabocha/
[3] http://nlp.kuee.kyoto-u.ac.jp/nl-resource/knp.html

N: noun
V: verb
Adv: adverb
AdvN: adverbial noun
Pro: pronoun
cp: case particle
tp: topic-marking particle
ap: adnominal particle
aux: auxiliary verb
punc: punctuation

Japanese base-chunk
(*bunsetsu*)

Figure 1: MDS of sentence (6).

C: Content part
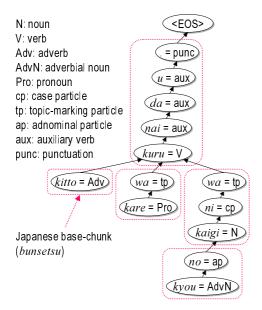F: Function part

Japanese base-chunk
(*bunsetsu*)

Figure 2: CFDS of sentence (6).

following two versions of dependency structures whose nodes are smaller than *bunsetsu*.

MDS: Morpheme-based dependency structure (Takahashi et al., 2001) regards a morpheme as a node. MDS of sentence (6) is shown in Figure 1.

CFDS: The node of a content-function-based dependency structure is either a sequence of content words or of function words. CFDS of sentence (6) is shown Figure 2.

Structured N-gram language models were created from 15 years of Mainichi newspaper articles[4] using a dependency parser Cabocha, with N being varied from 1 to 3. Then, the 3-gram conditional probability $P_d(c_i|d_i^1, d_i^2)$ is given by the linear interpolation of those three models as follows:

$$
\begin{aligned}
P_d(c_i|d_i^1, d_i^2) &= \lambda_3 P_{ML}(c_i|d_i^1, d_i^2) \\
&\quad +\lambda_2 P_{ML}(c_i|d_i^1) \\
&\quad +\lambda_1 P_{ML}(c_i), \\
\text{s.t.} \quad \sum_j \lambda_j &= 1,
\end{aligned}
$$

where mixture weights λ_j are selected via an EM algorithm using development data[5] that has not been used for estimating P_{ML}.

3.3 Similarity factor

The similarity factor $Sim(s, t)$ quantifies how two phrases s and t are similar by comparing two sets of contextual features $f \in F$ for s and t.
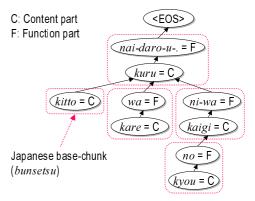
We employ the following two types of feature sets, which we have examined in our previous work (Fujita and Sato, 2008), where a feature f consists of an expression e and a relation r:

BOW: A pair of phrases is likely to be semantically similar, if the distributions of the words surrounding the phrases is similar. The relation set R_{BOW} contains only "co-occur_in_the_same_sentence".

MOD: A pair of phrases is likely to be substitutable with each other, provided they share a number of instances of modifiers and modifiees: the set of the relation R_{MOD} consists of two relations "modifier" and "modifiee".

Conditional probability distributions $P(f|s)$ and $P(f|t)$ are estimated using a Web search engine as in (Fujita and Sato, 2008). Given a phrase p, snippets of Web pages are firstly obtained via Yahoo API[6] by issuing p as a query. The maximum number of snippets is set to 1,000. Then, the features of the phrase are retrieved from those snippets using a morphological analyzer ChaSen[7] and CaboCha. Finally, the conditional probability distribution $P(f|p)$ is estimated as follows:

$$
\begin{aligned}
P(f|p) &= P(\langle r, e \rangle | p) \\
&= \frac{freq_{sni}(p, r, e)}{\sum_{r' \in R} \sum_{e'} freq_{sni}(p, r', e')},
\end{aligned}
$$

where $freq_{sni}(p, r, e)$ stands for the frequency of the expression e appealing with the phrase p in relation r within the snippets for p.

The weight for features $P(f)$ is estimated using a static corpus based on the following equation:

$$
\begin{aligned}
P(f) &= P(\langle r, e \rangle) \\
&= \frac{freq_{cp}(r, e)}{\sum_{r' \in R} \sum_{e'} freq_{cp}(r', e')},
\end{aligned}
$$

---

[4]Mainichi 1991-2005 (1.5GB, 21M sentences).
[5]Yomiuri 2005 (350MB, 4.7M sentences) and Asahi 2005 (180MB, 2.7M sentences).

[6]http://developer.yahoo.co.jp/search/
[7]http://chasen.naist.jp/hiki/ChaSen/

where $freq_{cp}(r, e)$ indicates the frequency of the expression e appearing with something in relation r within the given corpus. Two different sorts of corpora are separately used to build two variations of $P(f)$. The one is Mainichi, which is used for building structured N-gram language models in Section 3.2, while the other is a huge corpus consisting of 470M sentences collected from the Web (Kawahara and Kurohashi, 2006).

4 Experiments

4.1 Data

We conducted an empirical experiment to evaluate the proposed model using the test suite developed in (Fujita and Sato, 2008). The test suite consists of 176,541 pairs of paraphrase candidates that are automatically generated using a pattern-based paraphrase generation system (Fujita et al., 2007) for 4,002 relatively high-frequency phrases sampled from a newspaper corpus[8].

To evaluate the system from a generation viewpoint, i.e., how well a system can rank a correct candidate first, we extracted paraphrase candidates for 200 randomly sampled source phrases from the test suite. Table 1 shows the statistics of the test data. The "All-Yield" column shows that the number of candidates for a source phrase varies considerably, which implies that the data contains cases that have various difficulties. While the average number of candidates for each source phrase was 48.3 (the maximum was 186), it was dramatically reduced through extracting features for each source and candidate paraphrase from Web snippets: to 5.2 with BOW and to 4.8 with MOD. This suggests that a large number of spurious phrases were generated but discarded by going to the Web, and the task was significantly simplified.

4.2 Questions

Through this experiment, we evaluated several versions of the proposed model to answer the following questions:

Q1. Is the proposed model superior to existing measures in practice? Par_{Lin} and Par_{skew} are regarded as being the baseline.

Q2. Which language model performs better at estimating $P(t)$? MDS and CFDS are compared.

Q3. Which corpus performs better at estimating $P(f)$? The advantage of Kawahara's huge

---

[8]The grammaticality of the source phrases are guaranteed.

Table 1: Statistics of test data ("Ph.": # of phrases).

Phrase type	Source Ph.	All Ph.	All Yield	BOW Ph.	BOW Yield	MOD Ph.	MOD Yield
N:C:V	18	57	3.2	54	3.0	54	3.0
N_1:N_2:C:V	57	4,596	80.6	594	10.4	551	9.7
N:C:V_1:V_2	54	4,767	88.3	255	4.7	232	4.3
N:C:Adv:V	16	51	3.2	39	2.4	38	2.4
Adj:N:C:V	2	8	4.0	5	2.5	5	2.5
N:C:Adj	53	173	3.3	86	1.6	83	1.6
Total	200	9,652	48.3	1,033	5.2	963	4.8

corpus (WebCP) over Mainichi is evaluated.

Q4. Which set of features performs better? In addition to BOW and MOD, the harmonic mean of the scores derived from BOW and MOD is examined (referred to as HAR).

Q5. Can the quality of $P(f|s)$ and $P(f|t)$ be improved by using a larger number of snippets? As the maximum number of snippets (N_S), we compared 500 and 1,000.

4.3 Results

Two assessors were asked to judge paraphrase candidates that are ranked first by either of the above models if each candidate satisfies each of the three conditions. The results for all the above options are summarized in Table 2, where the strict precision is calculated based on those cases that gain two positive judgements, while the lenient precision is for at least one positive judgement.

A1: Our greatest concern is the actual performance of our probabilistic model. However, no variation of the proposed model could outperform the existing models (Par_{Lin} and Par_{skew}) that only assess similarity. Furthermore, McNemer's test with $p < 0.05$ revealed that the precisions of all the models, except the combination of CFDS for $P(t)$ and Mainichi for $P(f)$, were significantly worse than those of the best models.

To clarify the cause of these disappointing results, we investigated the performance of each factor. Table 3 shows how well the grammaticality factors select a grammatical phrase, while Table 4 illustrates how well the similarity factors rank a correct paraphrase first. As shown in these tables, neither factor performed the task well, although combinations produced a slight improvement in performance. A detailed discussion is given below in A2 for the grammaticality factors, and in A3-A5 for the similarity factors.

A2: Comparisons between MDS and CFDS revealed that CFDS always produced better results than MDS not only when used for measuring grammaticality (Table 3), but also when used as a

Table 2: Precision for 200 test cases.

$N_S = 500$	Strict			Lenient		
Model	BOW	MOD	HAR	BOW	MOD	HAR
Par_{Lin}	78 (39%)	**88** (44%)	87 (44%)	116 (58%)	**128** (64%)	127 (64%)
Par_{skew}	81 (41%)	**88** (44%)	**88** (44%)	120 (60%)	127 (64%)	**128** (64%)
MDS, Mainichi	72 (36%)	73 (37%)	76 (38%)	109 (55%)	112 (56%)	114 (57%)
MDS, WebCP	71 (36%)	73 (37%)	72 (36%)	108 (54%)	110 (55%)	113 (57%)
CFDS, Mainichi	79 (40%)	78 (39%)	**83** (42%)	120 (60%)	119 (60%)	**123** (62%)
CFDS, WebCP	79 (40%)	77 (39%)	80 (40%)	118 (59%)	116 (58%)	118 (59%)

$N_S = 1,000$	Strict			Lenient		
Model	BOW	MOD	HAR	BOW	MOD	HAR
Par_{Lin}	79 (40%)	88 (44%)	88 (44%)	116 (58%)	128 (64%)	**129** (65%)
Par_{skew}	84 (42%)	**89** (45%)	**89** (45%)	121 (61%)	128 (64%)	128 (64%)
MDS, Mainichi	72 (36%)	75 (38%)	76 (38%)	109 (55%)	114 (57%)	114 (57%)
MDS, WebCP	71 (36%)	74 (37%)	72 (36%)	109 (55%)	111 (56%)	113 (57%)
CFDS, Mainichi	79 (40%)	82 (41%)	**83** (42%)	121 (61%)	121 (61%)	**122** (61%)
CFDS, WebCP	79 (40%)	78 (39%)	79 (40%)	119 (60%)	116 (58%)	119 (60%)

Table 3: Precision of measuring grammaticality.

Model	Strict	Lenient
MDS	104 (52%)	141 (71%)
CFDS	**108** (54%)	**142** (71%)

Table 4: Precision of similarity factors.

N_S	Corpus	Strict			Lenient		
		BOW	MOD	HAR	BOW	MOD	HAR
500	Mainichi	60 (30%)	68 (34%)	**74** (37%)	98 (49%)	109 (55%)	114 (57%)
500	WebCP	57 (28%)	61 (31%)	**74** (37%)	94 (47%)	99 (50%)	**120** (60%)
1,000	Mainichi	57 (28%)	70 (35%)	**74** (37%)	92 (46%)	113 (57%)	116 (58%)
1,000	WebCP	57 (28%)	60 (30%)	72 (36%)	93 (47%)	96 (48%)	116 (58%)

component of the entire model (Table 2). This result is quite natural because MDS cannot verify the collocation between content words in those cases where a number of function words appear between them. On the other hand, CFDS with $N = 3$ could verify this as a result of treating the sequence of function words as a single node.

As mentioned in A1, however, a more sophisticated language model must enhance the proposed model. One way of obtaining a suitable granularity of nodes is to introduce latent classes, such as the Semi-Markov class model (Okanohara and Tsujii, 2007). The existence of many orthographic variants of both the content and function words may prevent us from accurately estimating the grammaticality. We plan to normalize these variations by using several existing resources such as the Japanese functional expression dictionary (Matsuyoshi, 2008).

A3: Contrary to our expectations, the huge Web corpus did not offer any advantage over the newspaper corpus: Mainichi always produced better results than WebCP when it was combined with the grammaticality factor or when MOD was used.

We can speculate that morphological and dependency parsers produce errors when features are extracted, because they are tuned to newspaper articles. Likewise, $P(f|s)$ and $P(f|t)$ may involve noise even though they are estimated using relatively clean parts of Web text that are retrieved by querying phrase candidates.

A4: For Par_{Lin} and Par_{skew}, different sets of features led to consistent results with our previous experiments in (Fujita and Sato, 2008), i.e., BOW $<$ MOD $\simeq$ HAR. On the other hand, for the proposed models, MOD and HAR led to only small or sometimes negative effects. When the similarity factor was used alone, however, these features beat BOW. Furthermore, the impact of combining BOW and MOD into HAR was significant.

Given this tendency, it is expected that the grammaticality factor might be excessively emphasized. Our probability model was derived straightforwardly from the conditional probability $P(t|s)$; however, the combination of the two factors should be tuned according to their implementation.

A5: Finally, the influence of the number of Web snippets was analyzed; no significant difference was observed.

This is because we could retrieve more than 500 snippets for only 172 pairs of expressions among our test samples. As it is time-consuming to obtain a large number of Web snippets, the trade-off between the number of Web snippets and the performance should be investigated further, although the quality of the Web snippets and what appears at the top of the search results will vary according to several factors other than linguistic ones.

5 Conclusion

A pair of expressions qualifies as paraphrases iff they are semantically equivalent, substitutable in some context, and grammatical. In cases where paraphrase knowledge is represented with abstract patterns to attain a high coverage of the paraphrase phenomena, we should assess not only the first and second conditions, but also the third condition.

In this paper, we proposed a probabilistic model for computing how two phrases are likely to be paraphrases. The proposed model consists of two components: (i) a structured N-gram language model that ensures grammaticality and (ii) a distributional similarity measure for estimating semantic equivalence and substitutability between two phrases. Through an experiment, we empirically evaluated the performance of the proposed model and analyzed the characteristics.

Future work includes building a more sophisticated structured language model to improve the performance of the proposed model and conducting an experiment on template-like paraphrase knowledge for other than productive paraphrases.

References

Bannard, Colin and Chris Callison-Burch. 2005. Paraphrasing with bilingual parallel corpora. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics* (*ACL*), pages 597–604.

Barzilay, Regina and Kathleen R. McKeown. 2001. Extracting paraphrases from a parallel corpus. In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics* (*ACL*), pages 50–57.

Barzilay, Regina and Lillian Lee. 2003. Learning to paraphrase: an unsupervised approach using multiple-sequence alignment. In *Proceedings of the 2003 Human Language Technology Conference and the North American Chapter of the Association for Computational Linguistics* (*HLT-NAACL*), pages 16–23.

Dolan, Bill, Chris Quirk, and Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: exploiting massively parallel news sources. In *Proceedings of the 20th International Conference on Computational Linguistics* (*COLING*), pages 350–356.

Fujita, Atsushi, Shuhei Kato, Naoki Kato, and Satoshi Sato. 2007. A compositional approach toward dynamic phrasal thesaurus. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing* (*WTEP*), pages 151–158.

Fujita, Atsushi and Satoshi Sato. 2008. Computing paraphrasability of syntactic variants using Web snippets. In *Proceedings of the 3rd International Joint Conference on Natural Language Processing* (*IJCNLP*), pages 537–544.

Geffet, Maayan and Ido Dagan. 2004. Feature vector quality and distributional similarity. In *Proceedings of the 20th International Conference on Computational Linguistics* (*COLING*), pages 247–253.

Habash, Nizar. 2004. The use of a structural N-gram language model in generation-heavy hybrid machine translation. In *Proceedings of the 3rd International Natural Language Generation Conference* (*INLG*), pages 61–69.

Harris, Zellig. 1957. Co-occurrence and transformation in linguistic structure. *Language*, 33(3):283–340.

Harris, Zellig. 1968. *Mathematical structures of language.* John Wiley & Sons.

Jacquemin, Christian. 1999. Syntagmatic and paradigmatic representations of term variation. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics* (*ACL*), pages 341–348.

Kawahara, Daisuke and Sadao Kurohashi. 2006. Case frame compilation from the Web using high-performance computing. In *Proceedings of the 5th International Conference on Language Resources and Evaluation* (*LREC*).

Kilgarriff, Adam. 2007. Googleology is bad science. *Computational Linguistics*, 33(1):147–151.

Lee, Lillian. 1999. Measures of distributional similarity. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics* (*ACL*), pages 25–32.

Lin, Dekang. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics* (*COLING-ACL*), pages 768–774.

Lin, Dekang and Patrick Pantel. 2001. Discovery of inference rules for question answering. *Natural Language Engineering*, 7(4):343–360.

Matsuyoshi, Suguru. 2008. *Hierarchically organized dictionary of Japanese functional expressions: design, compilation and application.* Ph.D. thesis, Graduate School of Informatics, Kyoto University.

Mel'čuk, Igor and Alain Polguère. 1987. A formal lexicon in meaning-text theory (or how to do lexica with words). *Computational Linguistics*, 13(3-4):261–275.

Okanohara, Daisuke and Jun'ichi Tsujii. 2007. A discriminative language model with pseudo-negative samples. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics* (*ACL*), pages 73–80.

Pantel, Patrick, Rahul Bhagat, Bonaventura Coppola, Timothy Chklovski, and Eduard Hovy. 2007. ISP: Learning inferential selectional preferences. In *Proceedings of Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics* (*NAACL-HLT*), pages 564–571.

Quirk, Chris, Chris Brockett, and William Dolan. 2004. Monolingual machine translation for paraphrase generation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing* (*EMNLP*), pages 142–149.

Szpektor, Idan, Hristo Tanev, Ido Dagan, and Bonaventura Coppola. 2004. Scaling Web-based acquisition of entailment relations. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing* (*EMNLP*), pages 41–48.

Takahashi, Tetsuro, Tomoya Iwakura, Ryu Iida, Atsushi Fujita, and Kentaro Inui. 2001. KURA: a transfer-based lexico-structural paraphrasing engine. In *Proceedings of the 6th Natural Language Processing Pacific Rim Symposium* (*NLPRS*) *Workshop on Automatic Paraphrasing: Theories and Applications*, pages 37–46.

Weeds, Julie, David Weir, and Bill Keller. 2005. The distributional similarity of sub-parses. In *Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment*, pages 7–12.

Yamamoto, Kazuhide. 2002. Acquisition of lexical paraphrases from texts. In *Proceedings of the 2nd International Workshop on Computational Terminology* (*CompuTerm*), pages 22–28.