# Terminal Device Oriented Comparable Corpora and its Alignment

# -- Towards Extracting Paraphrasing Patterns --

**Hiroshi Nakagawa*, Hidetaka Masuda†, Dai Sato†**

* Information Technology Center, The University of Tokyo
Hongou, Bunkyo, Tokyo, 113-0033, Japan
nakagawa@dl.itc.u-tokyo.ac.jp

†Tokyo Denki University
Kanda-Nishiki-cho., Chiyoda, Tokyo,101-8457, Japan
masuda@im.dendai.ac.jp, sdai@cdl.im.dendai.ac.jp

**Abstract**

Many terminal devices for mobile environment such as mobile phones have small and low resolution screens compared to the big and high resolution screen of personal computers. In this circumstance, Web pages for ordinary personal computer and mobile phones written in the same language are developed separately even though they describe the same topic or contents. In this research, we collected Web news articles aimed at displaying on personal computer screens and news articles aimed at mobile terminals for more than two years. Then we aligned these two kinds of news articles first in article level and then in sentence level. As the result, we got more than 88,000 pairs of aligned sentences. Next, we extract paraphrases of the final part of sentences from this aligned corpus. Actual results are the sentence final nouns of mobile article sentences and their counterpart expressions of Web article sentences. We extract character strings for paraphrases based on branching factor, frequency and length of string. The precision is 90% for highest ranked candidate and 80% for each top four candidates of 10 most frequently used nouns.

## 1. Introduction

Recently many terminal devices for mobile environment, like mobile phones and PDAs, became used as well as personal computers in our everyday life. These kinds of devices have small and low resolution screens. On the other hand, a personal computer has a big and high resolution screen. In this circumstance, two types of Web pages written in the same language are developed separately even though they describe the same topic or contents: one for ordinary personal computer and the other for mobile terminals.

Let's look at these types of Web pages, especially their text parts, from the viewpoint of computational linguistics. They are seen as comparable corpora because the topics are same but the texts are not. They are different because they are aimed at displaying on different terminal devices respectively. In other words, they are comparable corpora in terms of terminal devices.

In computational linguistics, the research topics for parallel and/or comparable corpora include alignment and extraction of translations. Thus alignment is thought to be an important research topic for the collection of those two types of Web pages. Paraphrase extraction is also promising research topic because these Web pages are written in the same language. These topics are what we pursue in this paper.

Japanese newspaper company Mainichi publishes two kinds of articles about the same news on the Web. One is for ordinary personal computers and the other is for mobile terminals. The latter articles are, in Japan, distributed via "i-MODE" distribution system run by NTT DoCoMo company. Henceforth, we call the Internet newspaper articles aimed at personal computers "Web articles" and those distributed on i-MODE system aimed at mobile phones "mobile articles." In this paper, we first compare Web articles and mobile articles. Then we describe the alignment of these two kinds of news articles. Finally, we propose the way to extract paraphrases from this aligned corpus. Those paraphrases would be useful, for instance, to compress the Web articles into shorter mobile articles.

## 2. Web articles and mobile articles

### 2.1. Characteristics

More than a hundred Web newspaper articles written in Japanese are distributed on the Web every day from Mainichi newspaper company ( http:www.mainich.co.jp). Their lengths are a few hundreds characters up to five hundreds and the average length is about 250 characters. A Web article consists of several key words, a title and a body of text.

About 70 mobile newspaper articles also written in Japanese are distributed by Mainichi newspaper co. via i-MODE system. The average length of one article is generally around 50 characters for old types of mobile phones, however some are at most 100 characters for the new types of mobile phones that display almost 100 characters. A mobile article consists only of the body text.

Since they are on the Web only for a few days, we have routinely downloaded them day-to-day basis. Actually we gathered 48,075 pairs of Web articles and mobile articles of Mainichi newspaper from April 26th 2001 to March 30th 2003. Since one mobile article often consists of more than one sentences, the total number of mobile sentences is 88,333. In the following sections, we describe alignment of these two types of articles.

### 2.2. Sentence final parts of mobile articles

Mobile articles are short and compact. We find this compactness especially in final part of mobile article's sentences. Ordinary sentences, which are obviously used in Web articles, almost always end with a verb or an auxiliary verb because Japanese is a head final language. On the contrary, sentences of mobile articles end variety of patterns as shown in Table 1, where the ratio is the total of each case against the above described 88333 mobile sentences.

| POS | Ratio(%) |
|---|---|
| SA-hen noun | 38.8 |
| Other noun | 18.0 |
| post positional particle | 16.4 |
| verb | 18.3 |
| auxiliary verb | 7.6 |
| others | 0.9 |

Table 1: Patterns of sentence final parts of mobile articles

In this table, SA-hen noun is a kind of noun expressing action etc. Its English counterpart is a noun appearing in the pattern of light verb + noun, i.e. "tennis" in "do tennis." In addition, within 30 most frequent expressions appearing at the last part of sentences, 15 of them are SA-hen noun. From the viewpoint of paraphrasing, this is a compression of a phrase which contains SA-hen noun + light verb. Considering these factors, we focus on this type of expressions as our target of paraphrase extraction in this paper.

# 3. Alignment

## 3.1. Article to article alignment

As stated previously, the number of Web articles is larger than the number of mobile articles, every mobile article has its counterpart in Web article. In this circumstance, the first thing to do is to find the Web article which corresponds to each mobile article. For this we use the similarity score: *SimArticle(W, M)* where *W* means a Web article and *M* means a mobile article.

$$SimArticle(W,M) = a \times K + b \times T + NN \quad (1)$$

where *K* is the number of *W*'s key words which also appear in *M*, T is the number of nouns in the title of *W* which also appear in *M*, and *NN* is the number of nouns that appear in both of *W*'s body and *M*, respectively. The parameters *a* and *b* are weights of the first and second factors respectively and both are chosen to be 3.0 experimentally. Sentence pairs whose *SimArticles* are more than 35 are correct pairs as far as investigating randomly selected 605 mobile articles by hand and resulting in 481 correct pairs. Thus we apply this threshold of 35 to all of Web articles and mobile articles described in section 2.

## 3.2. Sentence to sentence alignment

Next, we extract sentence pairs from these paired articles. Since newspaper articles always put the most important information in the first few sentences, we only focus on the first paragraph of Web articles. Practically, sentences of Web article matching the sentence of mobile article are identified by the following method where Ws means a sentence in the Web article's first paragraph, Ms means a sentence of mobile article, and Ws(Ms) is a Web article sentence aligned to Ms.

foreach (Ms)
{Ws(Ms) = a sentence in {Ws} having highest similarity with Ms} (2)

where the similarity is defined as a number of shared nouns by both of Ws and Ms.

We extract 88,333 aligned pairs of sentences by this method. We choose 500 pairs randomly from these pairs and check by hand. Then 92.8% of them were correctly aligned. This figure might not be sufficiently high for alignment task itself. The main objective of our research, however, is extraction of paraphrases by means of some statistical method. Therefore we decided not to pay more effort for alignment but to proceed to the next task namely extraction of paraphrases using these pairs of sentences.

# 4. Paraphrase Extraction

## 4.1. Background

Recently, paraphrase extraction became one of the main research topics in computational linguistics. Enormous amount of research results have been published through many workshops as well as conferences (Sato and Nakagawa,2001), (Inui and Hermjakob, 2003) and so on. Paraphrase candidates extraction from an entire corpus is the first tough task to paraphrase acquisition. Several sophisticated methods are proposed such as (Yamamoto, 2002). However this difficulty can be avoided by using parallel corpus (Brazilay and McKeown, 2001). We are also using a corpus of aligned sentences described in Section 3 except that our corpus consists of Web articles and mobile articles. Using our aligned corpus, we can avoid the problem about how to find paraphrase candidates from the entire corpus.

Paraphrases would be used for many purposes including text simplification (Inui, et al, 2003). Our target paraphrases are expressions of the same meaning in Web sentences and mobile sentences. The latter is a more compact and simplified form of the former. What we want to extract is paraphrases by which we simplify Web sentences into sentences that can be used as mobile sentences. By using the corpus of aligned sentences of Web and mobile, we can extract paraphrases that fit well our purpose.

## 4.2. Extraction framework

Paraphrases we focus on are the last clause of each of Web sentences and the last noun phrases of mobile sentences because (1) a last part of sentence is usually verb in Japanese, and (2) a last part of sentence of compact text like mobile articles is often a noun which expresses an action etc.. For instance, Japanese verb phrase "waka-tta"('proved out to be') is sometimes paraphrased with a noun "hanmei" which can be

translated into "be known", "be discovered", "be identified" etc.

The extraction framework we used is:

**Step 1:** Gather mobile article sentences: Ms having the same expression like "hanmei" at the last part of sentence.
**Step 2:** Gather Web article sentences: Ws(Ms) paired with each of Ms gathered at Step 1.
**Step 3:** Extract strings: Str's which are the candidates of paraphrases from the end of each Ws(Ms) gathered at Step 2 backwardly and in character-by-character manner.
**Step 4:** Sort the candidates extracted at Step 3 in descending order of appropriateness as a paraphrase.

Figure 1. The framework of paraphrase extraction

Since we use the aligned sentences described in Section 3.2, the set of sentences:{Ws(Ms)} which is the result at Step 2 contain paraphrases of Ms. Therefore it is extremely easier to find paraphrases in this case than the cases using non-aligned corpus.

Even though we use aligned sentences to extract paraphrases, we still have many possible types of paraphrases like a noun, a noun phrase, verb, verb phrases, and so on. Thus we make the problem one step easier by focusing on the last parts of sentences as stated at Step 3. By this narrowing down, we identify where paraphrases exist. The remaining problem is the way to identify character strings that are proper paraphrases from the last part of sentence. This problem is dealt with at Step 3 and 4. We will describe these steps in the remaining part of this paper.

### 4.3. Character based extraction with branching factor and frequency

As already said, we focus only on paraphrase between SA-hen noun appearing at the end of a mobile sentence and their counterpart expressions which are located in the last part of Web sentence aligned with the mobile sentence. In our extraction system, we extract a SA-hen noun which locates at the last part of mobile sentence Ms at first. Henceforth we call this SA-hen noun SAN(Ms). Secondly, we extract character strings from the last part of the Web sentences {Ws(Ms)}.

The problem is how many characters from the end of sentence Ws(Ms) we should extract. To solve this problem, we pay attention to the following three factors.

**Factor 1: Frequency:** As all of the Ws(Ms)'s contain paraphrases of SAN(Ms), we expect that many of them share the same expression which has the same meaning SAN(Ms) represents. Therefore the character string which has a high frequency within {Ws(Ms)} probably is a paraphrase of SAN.

**Factor 2: Branching factor:** Here, we define forward and backward branching factor of a character string Cs in a set of sentences. Forward branching factor: FB(Cs) is defined as the number of kinds of character which are right adjacent to Cs in a set of sentences. Let Cs be "n." Then FB("n") is big because we may have many kinds of characters after "n" of the first character of words, like "na", "ne", etc. After "na", "m" of "name", "t" of "nation" or "nature", etc. may come, and still FB("na") is high. But after "natu", very few kinds of character can come like "r" of "nature." Thus FB decreases as we proceed left within a word. Obviously, once a word ends, forward branching factor suddenly increases. Thus we would extract linguistically meaningful expression by cutting out character strings at the point where a FB increases.

The backward branching factor: BB is defined as the number of kinds of character which are left adjacent to Cs in a set of sentences. We expect that if we scan character string backwards from the end of sentence, the same situation as described in a forward branching case is expected to happen. We depict the situation with more concrete example.

Consider, for instance, the Japanese sentence
"Han nin ga tai ho sa re ta"(The suspect was arrested) (3)
In (3) a sequence of character separated by a space such as "re" and "ta" indicates one Japanese character.

The candidates of Japanese character strings taken from the end of sentence are: "ta", "re ta", "sa re ta", "ho sa re ta", "tai ho sa re ta", and so forth. Here "ta" is an independent morpheme which indicates past tense. Therefore very many kinds of character come to the left of "ta" like "re ta" in (3) or "si ta"(did), "ki ta"(came), "mi ta"(saw) and so on. Actually stems of all verbs can come. That means BB("ta") is extremely high. BB("re ta") is rather low because there can be several possible strings expressing linguistic functions like "sa re ta"(be -ed), "ra re ta" (be -ed), "ku re ta"(beneficiary be given), and so on. On the contrary, since "sa re ta" is fixed expression for passive voice, very many kinds of character can come just left of "sa re ta" like "ho sa re ta" of "tai ho sa re ta." In fact every SA-hen noun can come to the left of "sa re ta." Then BB("sa re ta") increases. By this nature of BB, we can extract the good candidates of fixed expression by picking up a character string from the end of sentence whose BB increases.

Taking into account these two factors, we propose the following method to extract candidate character string for paraphrase. Precisely speaking, we extract a character string which has high frequency and BB increases.

**Factor 3: length:** Paraphrase extraction method using frequency and BB is powerful but would extract noisy strings. One extreme case is short strings like "si ta"(did). Since every SA-hen noun can come just left of "si ta", BB("si ta") is high and possibly increasing, and the frequency of "si ta" is obviously high, we might extract "si ta" in some {Ws(Ms)}. Of course it is not desirable because apparently "si ta" is not a paraphrase of any SA-hen noun. The other extreme case is a long expression like "saku zitu tou kyou de tai ho sa re ta"(got arrested in Tokyo yesterday). It is not a paraphrase of SA-hen noun "tai ho"(arrest) because it expresses too detailed information than "tai ho" expresses. Thus we have to exclude too long and too short strings.

It is natural to expect that the longer a character string is, the less frequent it is. From this observation, we take the length of string or log of length of string as a length factor. Actually log(length – 1)is used to exclude less than two character expressions because 1) SA-hen nouns almost always consists of two Chinese characters and 2) Paraphrases in longer Web sentences for SA-hen nouns in mobile sentences are expected to be longer than two characters. Thus we expect the combination of frequency

and length factor accomplishes the task to identify paraphrase candidates.

In addition, it is not necessary to search very long character strings because the target is the paraphrase of one SA-hen noun. Thus we only focus on character strings which is less than 30 characters.

Based on these considerations, we propose a paraphrase extraction system which corresponds to Step 3 and 4 of Figure 1 as shown below.

**Step A:** Scan every sentence of {Ws(Ms)| Ms's have the same SAN(Ms)} backward from the end of sentence to extract character strings of any length less than 30 characters.

**Step B:** Calculate BB for every character string extracted at Step A.

**Step C:** Pick up character strings whose BB increases from the resultant set of strings at Step B. We denote the result as {Str}.

**Step D:** Sorting all strings in {Str} on the descending order of the product of BB multiplied by the frequency of Str multiplied by the log(length(Str)-1).

Figure 2. Algorithm to sort paraphrase candidates

### 4.4. Experimental results and evaluation

Our algorithm shown in Figure 2 indicates that we can only deal with a SAN(Ms) which has more than one Ws(Ms) because we use the frequency of {Str} at Step D. Due to this constraint, 4566 expressions located at the last part of mobile sentences are extracted from our aligned sentences. Now, we evaluate the paraphrases resulted in sorting algorithm described in Figure 2. As evaluation we test whether the resultant candidates of paraphrases are correct paraphrases in any context. This is done by hand because this correctness is known based on deep semantic analysis including even some consideration about contexts. Since we have more than 1000 SAN(Ms)s, we cannot evaluated every candidate of paraphrase for every SAN(Ms) by hand. Then we evaluate precisely the 10 most frequently used Japanese SA-hen nouns in our corpus:

Happyou (announce), Taiho (arrest), Kaidan (have a talk), Hyoumei(express, demonstrate), Sibou (die), Kettei (decide), Kyouchou (coordinate), Hanmei (proved to be, turn out to be, discover), Goui (agree), Kentou (examine).

If we closely look at these SA-hen nouns, all of them have several meanings even though they are roughly similar. Thus we expect to extract paraphrases that are similar but have a little bit distinct meaning of the original SA-hen nouns.

Actually, we test 20 highest ranked candidates for each of 10 SAN(Ms) resulted in by the sorting algorithm described in Figure 1 and 2 by hand and calculate precisions to N-th candidate defined by the following formula..

$$\text{Precision}(N) = \frac{1}{N}\sum_{i=1}^{N} C(i) \qquad (4)$$

where *C(i)* means the number of correct paraphrases within *i* highest candidates.

The result is shown in Figure 3. Nine out of the 10 highest ranked candidates are correct paraphrases. If we take three highest ranked candidates, almost 87% are correct. Moreover about 50% of 20 highest ranked candidates are correct. This result indicates that the expressions resulted in by our aligned sentences and extraction algorithm are high quality candidates of paraphrase. Thus if some experts with linguistic knowledge check them finally by hand, their burden is significantly reduced.

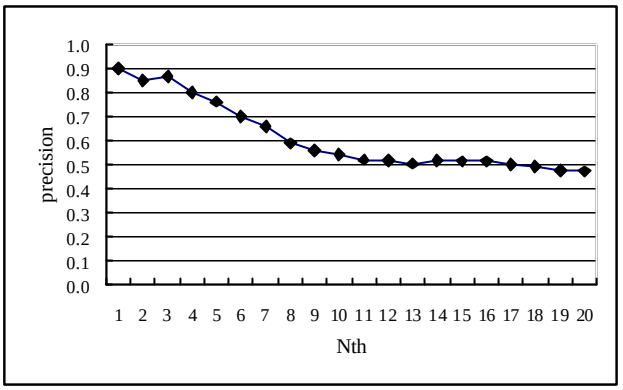


Figure 3. Precision(N) for 20 highest ranked candidates

As more comprehensive evaluation, we test the highest ranked candidates of paraphrases for 100 SA-hen noun. 84% of them are correct.

Finally we show some examples of extracted paraphrases.

Happyou(announce) => happyou-sita(made anouncemet), Suru-to happyou-sita(announce to do sth.), Akiraka-ni sita(disclosed)

Kettei(decide) => kime-ta(decided), suru-koto-wo kime-ta(decided to do sth.), kettei-sita(made a decision)

## 5. Conclusions

We collected and aligned Web news articles and news articles for mobile phones over two years. Using this aligned corpus, we extract character strings of paraphrases of SA-hen nouns appearing at the end mobile sentences based on the combination of branching factor, frequency and length. The samples of the result show high precision and indicate semi-automatic paraphrase extraction be possible.

## 6. References


Brazilay. R. and McKeown. K., 2001. Extracting paraphrases from a parallel corpus. *Proceedings of ACL-EACL2001*, 50-57.

Inui. K and Hermjakob. U. (eds.), 2003. *Proceedings of the Second International Workshop on Paraphrasing: Paraphrase Acquisition and Applications(IWP2003)* ACL2003, Sapporo.

Inui. K., Fujita. A., Iida. T., and Iwakura. 2003. Text Simplification for Reading Assistance: A Project Note. ibid. 9-16

Sato. S. and Nakagawa. H. (eds.), 2001. *Proceedings of Workshop: Automatic Paraphrasing: Theories and Applications*, NLPRS2001, Tokyo.

Yamamoto. K., 2002. Acquisition of Lexical Paraphrases from Texts, *Proceedings of COMPUTERM2 Workshop of COLING2002*,22-28