Online Japanese Unknown Morpheme Detection using Orthographic Variation

Yugo Murawaki, Sadao Kurohashi

Graduate School of Informatics, Kyoto University
Yoshida-honmachi, Sakyo-ku, Kyoto, 606-8501, Japan
murawaki@nlp.kuee.kyoto-u.ac.jp, kuro@i.kyoto-u.ac.jp

Abstract

To solve the unknown morpheme problem in Japanese morphological analysis, we previously proposed a novel framework of online unknown morpheme acquisition and its implementation. This framework poses a previously unexplored problem, online unknown morpheme detection. Online unknown morpheme detection is a task of finding morphemes in each sentence that are not listed in a given lexicon. Unlike in English, it is a non-trivial task because Japanese does not delimit words by white space. We first present a baseline method that simply uses the output of the morphological analyzer. We then show that it fails to detect some unknown morphemes because they are over-segmented into shorter registered morphemes. To cope with this problem, we present a simple solution, the use of orthographic variation of Japanese. Under the assumption that orthographic variants behave similarly, each over-segmentation candidate is checked against its counterparts. Experiments show that the proposed method improves the recall of detection and contributes to improving unknown morpheme acquisition.

1. Introduction

Dictionaries are indispensable resources in natural language processing. This is especially true for Japanese morphological analysis because it is not just part-of-speech (POS) tagging but segmentation is also required. Japanese, like Chinese and Thai, does not delimit words by white space, and due to boundary ambiguities, the joint task of segmentation and POS tagging has a much larger search space than simple POS tagging. In order to limit the search space, the enumeration of morpheme candidates is done by looking up a pre-defined dictionary.

Historically, extensive human resources were used to build high-coverage dictionaries (Yokoi, 1995). They now cover almost all but rare proper nouns in newspaper articles. Thus research concentrated on finding an optimal path when a high-coverage dictionary is available, and the F-score of nearly 99% was achieved (Kurohashi et al., 1994; Asahara and Matsumoto, 2000; Kudo et al., 2004).

Manually-constructed dictionaries do not, however, suffice for texts other than newspaper articles, web pages in particular, where morphological analysis is prone to more errors owing to unknown morphemes, or morphemes not in a dictionary. For example, the unknown verb "ググる" (`gugu-ru`, "to google") is erroneously segmented into "ググ" (`gugu`) and "る" (`ru`).

One solution to the problem is to automatically augment the dictionary by acquiring unknown morphemes from text (Mori and Nagao, 1996). We previously proposed the novel framework of online acquisition of unknown morphemes (Murawaki and Kurohashi, 2008). Unlike traditional batch extraction (Mori and Nagao, 1996), the proposed method has the ability to acquire unknown morphemes in an online mode. The lexicon acquirer processes text on a sentence by sentence basis. It directly updates the dictionary of the analyzer when it successfully disambiguates an unknown morpheme.

The framework of online acquisition poses a previously unexplored problem, online unknown morpheme detection. It is a task of finding morphemes in each sentence that are not listed in a given lexicon. Unlike in English, it is a non-trivial task because, again, Japanese does not delimit words by white space. We have to compare the whole sentence, not words, with the morphemes registered in the dictionary. In this paper, we first present a baseline method of detection that simply uses the output of the morphological analyzer. The baseline method can easily detect most unknown morphemes because they cannot be interpreted as registered morphemes. We then show that it fails to detect some unknown morphemes because they are over-segmented into shorter registered morphemes due to the overly simple sound structure of Japanese. For example, the unknown adjective "うざい" (`uza-i`, "annoying") is over-segmented into the combination of registered morphemes, "う" (`u`) and "ざい" (`zai`). To cope with this problem, we propose the use of orthographic variation of Japanese. Under the assumption that orthographic variants behave similarly, each over-segmentation candidate is checked against its counterparts. Experiments show that the proposed method improves the recall of detection and contributes to improving unknown morpheme acquisition.

2. Related Work

2.1. Morpheme Extraction from Text

Various methods are proposed to extract morphemes from text. For languages delimited by white space, they can be extracted from a word[1] list (Kurimo et al., 2006; Poon et al., 2009).

For languages where even word boundaries are unmarked, two major approaches are used. One is to segment the whole corpus and to build the dictionary, or the list of morphemes, from the segmented corpus. Segmentation models can be learnt from a manually-segmented training cor-

[1] Throughout this paper, we distinguish words from morphemes. Each word consists of one or more morphemes.

pus (Asahara and Matsumoto, 2000; Kudo et al., 2004) and from raw text by unsupervised methods (Goldwater et al., 2009; Zhao and Kit, 2008; Mochihashi et al., 2009). Unsupervised segmentation can incorporate supervised segmentation using it as the initial model (Xu et al., 2008). Unsupervised segmentation are usually evaluated in terms of token (corpus segmentation) and type (the list of unique morphemes). It is reported that type accuracy is considerably lower than token accuracy, suggesting that low frequency morphemes tend to be wrongly segmented.

Another approach is to directly extract morphemes from a raw corpus that satisfy certain criteria. Mori and Nagao (1996) and Feng et al. (2004) examine the surrounding context of each morpheme candidate to evaluate how likely it is a true morpheme. To improve precision, candidates with low frequencies are usually discarded.

In both approaches, a morpheme list is extracted from a corpus in a batch mode. If we have a manually constructed dictionary, those not in the dictionary are considered unknown morphemes. Here we face a dilemma. Since the manually constructed lexicon covers basic morphemes, unknown morphemes to be extracted generally occur infrequently, but they are often misidentified or ignored in these approaches.

Practical applications of these approaches are automatic speech recognition (ASR) and kana-kanji (phoneme-to-text) conversion, where the nosiness of the extracted data is not critical (Kurata et al., 2006; Kurata et al., 2007; Sasada et al., 2008). In fact, Kurata et al. (2007) point out that most of the morpheme candidates are just useless and meaningless character strings. In these tasks, segmented text is just an intermediate representation between the input (speech/phoneme) and the output (unsegmented text). Incorrectly segmented morphemes in the language models can produce correct unsegmented text. For example, even if the language model is build from a corpus where "うざい" (`uza-i`, "annoying") is always segmented into "う" (`u`) and "ざい" (`zai`), the system would wrongly recognize the input `u.za.i` as "う" (`u`) and "ざい" (`zai`) with high probability. However, this is transformed into a correct unsegmented output "うざい" (`uzai`) and it is indeed judged correct in the standard Character Error Rate (CER) evaluation. By contrast, Japanese morphological analysis requires a clean dictionary, and segmentation errors in morphological analysis have a serious negative effect on its applications such as dependency parsing and named entity recognition. Thus this approach cannot directly be applied to morphological analysis.

2.2. Unknown Morpheme Processing

Another line of research focuses on identifying unknown morphemes on demand. In Japanese morphological analysis, the analyzer enumerates morpheme candidates with unknown morpheme processing in addition to dictionary look-up, as illustrated in Figure 1. Unknown morpheme candidates generated by unknown morpheme processing are given the special POS tag *UNK*. When the analyzer selects an optimal path, registered morphemes are generally preferred. However, if they do not explain the input well, *UNK* morphemes are selected.

The widely adopted heuristics in unknown morpheme processing are based on character types because Japanese is written with several different scripts, or character types, such as hiragana and katakana (syllabaries), and kanji (logographs). Hiragana is used for functional elements while content words are usually written in kanji with some supplementary hiragana. Loan words are written in katakana. The choice of these scripts gives some clues on morpheme boundaries.

In the case of the morphological analyzer JUMAN,[2] a sequence of katakana characters becomes one *UNK* morpheme, while hiragana and kanji are segmented per character. For example, the katakana loan word "グーグル" (`gûguru`, "Google") out of "グーグルが" (`gûguru ga`, plus NOM) is listed as an *UNK* morpheme.

These heuristics are simple and effective, but far from perfect. The hiragana noun "ようつべ" (`youtsube`, "YouTube" in an unconventional spelling) is wrongly divided into "よ" (`yo`), "うつ" (`u-tsu`) and "べ" (`be`), where the last element is *UNK*. In addition, they can never identify mixed-script morphemes, verbs and adjectives correctly. For example, the verb "ググる" (`gugu-ru`) is wrongly divided into the katakana *UNK* "ググ" (`gugu`) and the hiragana suffix "る" (`ru`).

More sophisticated unknown morpheme models can be introduced to morphological analysis (Nagata, 1999; Uchimoto et al., 2001; Asahara and Matsumoto, 2004; Nakagawa, 2004). However, it is difficult and computationally expensive to identify both the boundaries and POS of each unknown morpheme. In fact, Asahara and Matsumoto (2004) and Nakagawa (2004) only identify the boundaries. Even so, the accuracy of unknown morpheme identification is not high.

3. Online Unknown Morpheme Acquisition

We previously proposed the novel framework of online acquisition of unknown morphemes (Murawaki and Kurohashi, 2008). This framework is in line of on-demand identification of unknown morphemes, but further relaxes the requirement of identification; the detection of unknown morphemes does not require correct boundary identification. Instead of trying to identify the boundaries and POS of a single unknown morpheme, detected unknown morphemes are accumulated and compared with each other to solve the ambiguity.

The key idea behind this framework is that although each instance of an unknown morpheme is ambiguous in terms of both boundaries and POS, we can solve the ambiguity by accumulating its multiple instances and comparing them. Take the verb "ググる" (`gugu-ru`) for example. The goal is to identify its stem and POS tag: <`gugu`, consonant-*r* verb>. When the lexicon acquirer receives its instance in text "ググってみた。" (`gugu-Qte mi-ta`, "to have tried to google"), it enumerates its morphologically acceptable interpretations including <`gugu`, consonant-*r* verb>, <`gugu`, consonant-*w* verb>, and <`guguQte`, consonant-*m* verb> (note the different stem candidates). The acquirer then receives other instances such as "ググるのは"

[2] `http://nlp.kuee.kyoto-u.ac.jp/nl-resource/juman-e.html`

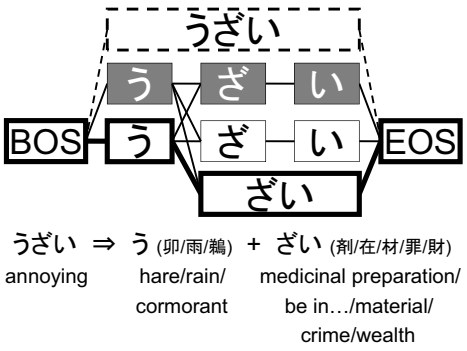

Figure 1: A lattice of morphemes. The dashed lines show the correct path, which is not enumerated by the analyzer. The selected path is indicated by bold lines. Registered morphemes are represented by white rectangles and *UNK*s are by gray ones.

(gugu-ru no ha, "to google TOPIC") and "ググらずに" (gugu-ra zu ni, "without googling"), and it becomes clear to the acquirer that only <gugu, consonant-*r* verb> can explain these instances.

The lexicon acquirer processes text on a sentence by sentence basis and accumulates examples of unknown morphemes. When it successfully disambiguates an unknown morpheme, it directly updates the dictionary of the analyzer, and the acquired morpheme will be used in subsequent analysis.

4. Unknown Morpheme Detection

4.1. Task Definition

Online unknown morpheme detection is a subtask of online unknown morpheme acquisition. In this setting, we have a manually-constructed dictionary that needs to be augmented with unknown morphemes, or morphemes not registered in it. Here unknown morphemes refer to those at the morphology level. A proper noun that is registered only as a common noun is out of the scope of acquisition because the distinction between common and proper nouns is at the semantics level.

Online detection is the task of finding any morpheme in a sentence that is not in the current dictionary. The input is a sequence of characters and the output is its subregion that *roughly* corresponds to an unknown morpheme. The detected region need not be the exact region of the stem of an unknown morpheme because it will be identified in later stages of online acquisition. Formally, let $[s_d, e_d]$ be the detected region in the sequence of characters and $[s_u, e_u]$ be the exact region of the stem. In our framework (Murawaki and Kurohashi, 2008), the detection is correct if $s_u \leq s_d \leq e_u$. For example, the stem of "うざい" (uza-i, "annoying") is "うざ" (uza), but it is acceptable to detect "う" (u) or "ざい" (zai).

4.2. Baseline Method

As a baseline method of online detection, we can use the output of the morphological analyzer. As seen in Section 2.2., the morphological analyzer uses unknown morpheme processing to generate *UNK* morphemes. Although *UNK*s are often incorrect in terms of segmentation, they usually meet the criterion of unknown morpheme detection. *UNK* is tagged wholly or partially to their stems, as in "ググる" (gugu-ru) and "ようつべ" (youtsube).

We apply morphological analysis to every input sentence and scan the resultant morpheme sequence to find *UNK*. If morpheme m_i in the morpheme sequence $m_0, \cdots m_{M-1}$ is *UNK*, then we detect region $[s_{m_i}, e_{m_i}]$, where s_{m_i} and e_{m_i} are the starting and ending positions of m_i in the character sequence. For simplicity, we will say that m_i is detected.

To be precise, the simplest method of detection would be string matching of the dictionary on the input sentence. The regions not covered by the matching would correspond to unknown morphemes. However, this method allows combinations of registered morphemes that are morphologically unacceptable. These phenomena can be suppressed to some degree by using the morphological analyzer that implicitly or explicitly utilizes grammatical knowledge.

4.3. Over-segmentation Problem

The baseline method cannot detect some unknown morphemes because they are over-segmented into shorter registered morphemes. Some of them are loan words written in katakana:

- カースト (kâsuto, "caste")
  ⇒ カー (kâ, "car")
  + スト (suto, abbr. of "strike")

- モニタリング (monitariNgu, "monitoring")
  ⇒ モニタ (monita, "monitor")
  + リング (riNgu, "ring")

In these examples, single loan words are divided into multiple loan words. The knowledge on the source languages (e.g. original spellings) would be useful for detecting them, but we do not discuss it in this paper.

Another type of over-segmentation typically involves hiragana characters:

- うざい (uza-i, "annoying" in plain form of conjugation)
  ⇒ う (u, "hare," "rain" or "cormorant")
  + ざい (zai, "medicinal preparation," "residing," "material," "guilt" or "wealth")

- うざくて (uza-kute, "annoying" in type-*ta* continuative *te*-form)
  ⇒ う (u, "hare," "rain" or "cormorant")
  + ざ (za, "seat")
  + く (ku, "ward" or "pain")
  + て (te, "hand")

- めんどかった (meNdo-kaQta, "tiresome" in *ta*-form)
  ⇒ めん (meN, "evasion," "cotton," "plane" or "noodle")
  + ど (do, "'degree')
  + かった (ka-Qta, "buy," "raise animals," "win," "reap," "hunt" or "'drive' in *ta*-form)

- かぐや姫 (kaguyahime, "Princess Kaguya")
  ⇒ かぐ (kagu, "furniture")
  + や (ya, "and/or") + 姫 (hime, "princess")

The lattice of the first example is illustrated in Figure 1. One thing that contributes to over-segmentation is the overly simple sound structure of Japanese: the hiragana syllabaries consist of only about 80 characters, and many hiragana morphemes are of one or two characters.

Among the above four examples, ''furniture and princess" is plausible even semantically, but the rest look semantically unacceptable to us even if we simulate the morphological analyzer by narrowing our scope to bigram. The bigrams, "う" (u) and "ざい" (zai), and "めん" (meN) and "ど" (do) seem highly unlikely.

In order to tackle the over-segmentation problem, it would be useful to investigate the reason why the analyzer does not notice these mismatches. The morphological analyzer does not employ lexicalized bigrams except for some functional morphemes but only uses POS bigrams (Kudo et al., 2004). While POS bigrams successfully solve ambiguity among registered morphemes, it is difficult to distinguish unknown morphemes from overlapping registered morphemes. In fact, the bigram of "う" (u) and "ざい" (zai) is just the bigram of a noun and a noun, which is morphologically acceptable and fairly common. To detect these semantic mismatches, we need some form of lexical knowledge.

For lexical knowledge, we start with simple lexicalized bigrams. Building N-grams requires segmented text, but it is not readily available since Japanese is a non-segmenting language. There are some manually annotated corpora but they are too small. N-grams used in automatic speech recognition (ASR) and other applications are usually build from automatically segmented corpora.

The question arises, then, whether we can use the output of morphological analysis to detect its errors. The answer is "unlikely" because N-grams aggregate systematic errors. Since "うざい" (uza-i) is almost always segmented into "う" (u) and "ざい" (zai), N-grams wrongly suggest that the bigram of "う" and "ざい" is probable.

5. Proposed Method

5.1. Orthographic Variation

In order to detect over-segmented unknown morphemes, we propose the use of orthographic variation. In Japanese, one morpheme can be spelled in various ways since its writing system allows us a great deal of flexibility in the choice of scripts. For example, the hiragana morpheme "う" (u, "hare") has a kanji counterpart "卯." Most hiragana morphemes have orthographic variants written with kanji, kanji and hiragana, or katakana, which are less likely to be affected by over-segmentation.

In this paper, we refer to groups of orthographic variants by slash-separated "repname." In the cases of "卯" and "う," both are contained by the repname "卯/う." Hiragana morphemes are often polysemous and contained by more than one repname. Other than "卯/う," the hiragana morpheme "う" is contained by repnames "雨/う" ("rain") and "鵜/う" ("cormorant").

We assume that orthographic variants behave similarly. For each over-segmentation candidate, we check the occurrences of its orthographic counterparts in a corpus, and determine if it is a correct segmentation. For example, if "うざい" (uza-i) actually consists of "う" (u) and "ざい" (zai), it is expected that its variants such as "卯ざい," "卯剤," and "雨ざい" also appear in a corpus. This is indeed not the case and we can detect the unknown morpheme.

We formalize the idea as follows. Suppose that we have a mapping from morpheme m_i to the set of its orthographic variants V_{m_i} (e.g. $V_{\text{う}} = \{$ 卯, 雨, 鵜 $\}$). When we find m_i in morpheme sequence $\cdots m_{i-1}, m_i, m_{i+1}, \cdots$ given by the analyzer, it is checked as an over-segmentation candidate. We first examine the forward bigram m_i, m_{i+1}, calculating the log likelihood ratio,

$$L_f(m_i, r_{i+1}) = log \frac{P(r_{i+1}|m_i)}{P(r_{i+1}|V_{m_i})},$$

where r_{i+1} is the repname of m_{i+1}. We detect m_i if the ratio is greater than a threshold. Due to polysemy, m_{i+1} can also have more than one repname ("ざい" is contained by repnames "剤/ざい," "在/ざい," and three others.). In such a case, we check the possible combinations of $L_f(m_i, r_{i+1})$, and detect m_i if all of them satisfy the above condition.

The bigram probabilities can be estimated using the maximum likelihood method,

$$P(r_{i+1}|m_i) = \frac{f(m_i, r_{i+1})}{f(m_i)},$$

and

$$P(r_{i+1}|V_{m_i}) = \frac{\sum_{m_i\prime \in V_{m_i}} f(m_i\prime, r_{i+1})}{\sum_{m_i\prime \in V_{m_i}} f(m_i\prime)}.$$

Similarly, we check the backward bigram m_{i-1}, m_i,

$$L_b(r_{i-1}, m_i) = log \frac{P(r_{i-1}|m_i)}{P(r_{i-1}|V_{m_i})}.$$

5.2. Training the Model

Given the mappings of orthographic variation, we prepare the initial N-grams. Note that we can also update the frequency counts of the N-gram model during detection.

We use texts segmented and tagged by the morphological analyzer. For the over-segmentation candidate m_i and its variant $m_i\prime \in V_{m_i}$, we need the frequency counts $f(m_i)$, $f(m_i, r_{i+1})$ and $f(r_{i-1}, m_i)$ We scan the morpheme sequence and increment the corresponding counts.

5.3. Detection

In online detection, every sequence of morphemes output by the analyzer is given as an input. The sequence is scanned from beginning to end to detect unknown morphemes. At each position i, the baseline method based on unknown morpheme processing is first applied. If they do not match m_i and it has orthographic variants, then orthographic variation is examined. The forward bigram m_i, m_{i+1} is checked, and if it satisfies the condition, m_i

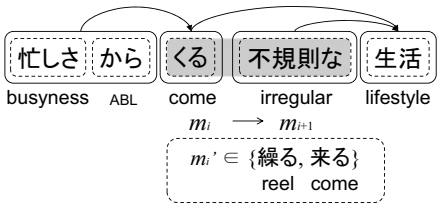

Figure 2: A bigram that crosses a *bunsetsu* boundary. *Bunsetsu* are indicated by the rounded rectangles, and the dashed rounded rectangles show morphemes. The rectangle at the bottom shows the orthographic variants of m_i.

is detected. Otherwise, the backward bigram m_{i-1}, m_i is checked in a similar way.

When a morpheme is detected, detection is skipped unless a clear boundary marker like punctuations comes to the position. Although this decreases recall, we can avoid detecting the same unknown morpheme twice or more.

5.4. Smoothing

The use of N-grams in detection is different from that in ASR and other applications, so a different strategy is required to handle the zero frequency problem. In ASR, non-zero probability is always necessary, but in detection, we want to determine whether a zero frequency N-gram is unacceptable for grammatical, semantic or other reasons, or it is just by accident.

We can virtually ignore zero frequency unigrams because only over-segmentation candidates and their counterparts are concerned. If the bigram count $f(m_i, r_{i+1})$ is large and its counterpart $\sum_{m_i' \in V_{m_i}} f(m_i', r_{i+1})$ is zero, it is highly likely that this is an over-segmentation. For $f(m_i, r_{i+1})$, we found in our preliminary experiment that it sometimes became zero even if we trained the model with a large scale web corpus. We attribute it to flexible constituency of the Japanese language. As shown in Figure 2, the pair of morphemes in a bigram can be syntactically unrelated when it crosses the boundary of phrasal unit called *bunsetsu*.

We smooth the probability estimates when a *bunsetsu* boundary is drawn between m_i and r_{i+1}. We interpolate the forward bigram probability $P(r_{i+1}|m_i)$ as follows.

$$P_{interp}(r_{i+1}|m_i) = \lambda P(r_{i+1}|m_i) + (1-\lambda)P(B|m_i)P(r_{i+1}|B),$$

where B is a *bunsetsu* boundary. The orthographic variants and the backward bigram are smoothed in similar ways. In training, we count $f(B)$, $f(m_i, B)$, $f(B, m_i)$, $f(r_i, B)$ and $f(r_i, B)$ in addition to the frequency counts described in Section 5.2..

6. Experiments

We evaluate the proposed method in terms of (1) the performance of unknown morpheme detection and (2) its contribution to unknown morpheme acquisition.

6.1. Data

We used the default dictionary of the morphological analyzer JUMAN as the initial lexicon. It contained 30 thousand basic morphemes. If spelling variants were expanded and proper nouns were counted, the total number of morphemes was 120 thousands.

For repnames or groups of orthographic variants, we used those listed in the dictionary of JUMAN version 5.0 or later. We semi-automatically constructed the mappings of orthographic variation as follows. First morphemes extracted from the dictionary were grouped by repname. Next, over-segmentation candidates were selected from each repname with some hand-written rules. Finally the mappings were manually corrected. We selected short hiragana, mixed-script spellings and some katakana morphemes as over-segmentation candidates. We obtained 12,082 over-segmentation candidates (conjugation variation of verbs and adjectives are not distinguished).

We trained the N-gram model on the web corpus that consists of 100 million pages. To keep the data size manageable, all bigram counts below the threshold 10 were ignored. We updated the counts during online acquisition.

We used the dependency parser KNP,[3] to obtain *bunsetsu* boundaries. KNP chunked morphemes into *bunsetsu* in preprocessing.

6.2. Detection

6.2.1. Settings

We evaluate unknown morpheme detection with precision and recall. For the evaluation of Japanese morphological analysis, Kyoto Text Corpus[4] is widely used. This is the very reason that it is not applicable to the evaluation of unknown morpheme detection. It contains a unnaturally small number of unknown morphemes since the morphological analyzer JUMAN and its dictionary have been developed using it as the benchmark corpus (Kurohashi and Nagao, 1998).

In order to evaluate performance concerning unknown morphemes, we need a large annotated corpus because unknown morphemes occur infrequently in general. However, fully annotating a large amount of text is too time-consuming and costly. We adopt an approximate but more efficient approach instead.

Precision is measured by manually judging the system output. We omitt from judgment detected morphemes that consist solely of katakana characters because they are overwhelming in number, generally correct and easily detected with the baseline method. We randomly select 500 detected morphemes for evaluation.

As for recall, we only focus on over-segmentation. We create a gold standard by manually correcting automatically extracted over-segmentation candidates. First, over-segmentation candidates are automatically extracted from text in the following steps.

1. Segment and tag each sentence with the morphological analyzer JUMAN.
2. Scan each morpheme sequence and extract as candidates the pairs of morphemes which any of the following rules matches:

[3] http://nlp.kuee.kyoto-u.ac.jp/nl-resource/knp-e.html

[4] http://nlp.kuee.kyoto-u.ac.jp/nl-resource/corpus-e.html

Table 1: Result of unknown morpheme detection.

	baseline	proposed
recall	346 / 1,004 (34.5%)	**723** / 1,004 (**72.0%**)
precision	**452** / 500 (**90.4%**)	412 / 500 (82.4%)
total	13,952	**15,612**
excl. katakana	3,206	**4,727**

(a) One-character hiragana + one-character hiragana,

(b) Two-character hiragana + one-character hiragana, and

(c) One-character hiragana + two-character hiragana.

3. Filter out candidates that are unlikely to be unknown morphemes. We use as "stop words" the pairs of morphemes that are extracted from Kyoto Text Corpus using the same rules.

We manually check every over-segmentation candidate. 2,870 over-segmentation candidates were extracted, and 1,004 unknown morphemes were manually tagged.
The system output is judged correct if it satisfies the condition described in Section 4.1.. For the evaluation of recall, we do not skip detection after one morpheme is detected. Note that the F-score cannot be calculated due to the above approximations.

6.2.2. Results

Table 1 shows the recall and precision of detection. The proposed method significantly improved recall over the baseline while the number of detected morphemes increased only moderately (by 11.9%). This is because an overwhelming number of unknown morphemes were written in katakana. If katakana ones were excluded, the proposed method considerably increased the number of detected morphemes.
Newly detected true positives include (detected morphemes are marked by underscores):

- かもめ (kamome, "gull")
  ⇒ かも (kamo, "duck") + め (me, "eye")
- ちゃちい (chachi-i, "cheap"; colloquial)
  ⇒ ちゃ (cha, "tea")
  + ちい (chii, "status" or "earch shock")
- よさこい (yosakoi, a folk song)
  ⇒ よ (yo-, "good" in bare stem form)
  + さ (sa, nominal predicative suffix)
  + こい (koi, "love," "intent" or "curp")

Most false negatives can be classified into two types. The first is the lack of orthographic variation. For example, "あずみ" (azumi, a given name) is segmented into "あ" (a, interjection) and "ずみ" (zumi, nominal suffix), and neither has orthographic variants. The second one is that the local scope of bigram does not suffice for detection. Unknown morphemes that fall into this type often contain particles in decomposed forms:

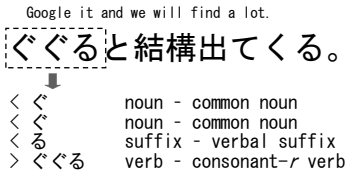

Figure 3: A "diff" block in a sentence.

- めも (memo, "memo")
  ⇒ め (me, "eye") + も (mo, "too")
- でかい (deka-i, "jumbo")
  ⇒ で (de-, "go out" in plain continuative form) + かい (kai, final particle)

These segmentations alone are completely natural. Since in most cases they are inconsistent with the whole sentences, wider concistency should be considered in future work.
The detection of some unknown morphemes depends on their surrounding context. For example, "はてな" (hatena, "question mark") alone is not detected because "な" (na) is interpreted as a particle. On the other hand, it is detected from "はてなが" (plus NOM) and "はてなを" (plus ACC), where "na" is interpreted as a noun by the analyzer because the particle cannot be followed by a case marker.
Errors of the baseline method in precision evaluation included 16 informal spelling alternates, 4 sentence extraction errors and one typo. One example of informal spellings was "な〜んてね" (nâNte ne, "just kidding"), an emphasized form of conventional "なんてね" (naNte ne). Some of these spellings can probably be filtered out with heuristic rules, but this problem should ultimately be solved with more robust morphological analysis.

6.3. Acquisition

6.3.1. Settings

Next, we evaluate the detection method by incorporating it into online unknown morpheme acquisition (Murawaki and Kurohashi, 2008). We examine the accuracy of acquired morphemes and their contribution to the improvement of morphological analysis.
We use domain-specific corpora as target texts because efficient acquisition is expected. If target texts share a topic, relevant unknown morphemes are frequently used. We use search engine TSUBAKI (Shinzato et al., 2008) and cast the search results as domain-specific corpora. For each query, our system sequentially reads pages from the top of the result and acquires morphemes. We terminate the acquisition at the 1,000th page and analyze the same 1,000 pages with the augmented lexicon. The queries used are "捕鯨問題" (whaling issue), "赤ちゃんポスト" (baby hatch) and "ジャスラック" (JASRAC, a copyright collective).
A morpheme is judged correct if both segmentation and POS are correct. Since segmentation criteria are a non-trivial problem for evaluation and not necessarily important in practice (Murawaki and Kurohashi, 2008), the segmentation is judged correct unless morpheme boundaries are clearly wrong.

Table 3: Evaluation of "diff" blocks.

	baseline					proposed				
query	E → C	C → C	E → E	C → E	total	E → C	C → C	E → E	C → E	total
whaling issue	**111**	121	0	11	243	**137**	79	1	15	232
baby hatch	**158**	38	11	5	212	**149**	39	10	7	205
JASRAC	**100**	81	21	9	211	**124**	67	13	6	210

(Legend – C: correct; E: erroneous)

Table 2: Accuracy of acquired unknown morphemes.

query	baseline	proposed
whaling issue	225/226 (99.6%)	244/246 (99.2%)
baby hatch	89/91 (97.8%)	90/92 (97.8%)
JASRAC	534/538 (99.3%)	570/580 (98.3%)

To examine the effect of acquisition, we analyze the target texts with both the initial lexicon and the augmented lexicon. Then we check differences between the two analyses and extract sentences that were affected by the augmentation. For each query, we use for evaluation 200 sentences randomly selected from them. We check the accuracy of each "diff" block, which is illustrated in Figure 3. Katakana blocks are, again, omitted from judgment. A "diff" is judged correct if for all morphemes in the block, both segmentation and POS are correct. We compare the baseline method and the proposed method with smoothing.

6.3.2. Results

Table 2 shows the results of acquisition. Compared with the baseline method, the proposed method slightly increased the number of acquired morphemes without seriously hurting accuracy. This improvement may look small, but it is because the overwhelming majority were katakana morphemes (75.6–79.7% of in the baseline method).

Table 3 shows the evaluation of "diff" blocks. The randomly selected data show almost no difference, but the numbers of sentences which contain "diff" blocks were increased by 7–38%.

Few false positives in detection led to wrong acquisition. Actually some erroneously detected registered morphemes accumulated enough examples for acquisition, but they were dropped at the time of acquisition simply because they conflicted with the registered morphemes.

Unknown morphemes newly acquired in the proposed method include "めんどくさい" (meNdokusa-i, "tiresome"), "わんこ" (waNko, "doggy"), "ねとらじ" (netoraji, abbr. of "internet radio"), "かがみん" (kagamiN, a person name) and "ドラえもん" (doraemoN, a robot cat; note the mixed-script spelling). These morphemes are much smaller in number than katakana morphemes like "グーグル" (gûguru, "Google"). However, they play more important role in NLP applications since the misidentification of these morphemes causes a serious negative effect on dependency parsing and other applications. For example, if "かがみん" (kagamiN) is not registered in the dictionary, it is transformed into a nonsensical parse tree that can be interpreted as "Summer did not see."

7. Conclusion

In this paper, we examine the previously unexplored problem of unknown morpheme detection. In order to detect unknwo morphemes that are over-segmented into shorter registered morphemes, we present a simple solution, the use of orthographic variation of Japanese. Complete detection remains unresolved because we know of no grammar or form of linguistic knowledge that exactly recognizes the set of acceptable languages. Yet we demonstrate that simple bigrams can detect a significant portion of over-segmentation.

8. References

Masayuki Asahara and Yuji Matsumoto. 2000. Extended models and tools for high-performance part-of-speech tagger. In *Proc. of COLING 2000*, pages 21–27.

Masayuki Asahara and Yuji Matsumoto. 2004. Japanese unknown word identification by character-based chunking. In *Proc. COLING 2004*, pages 459–465.

Haodi Feng, Kang Chen, Xiaotie Deng, and Weimin Zheng. 2004. Accessor variety criteria for Chinese word extraction. *Computational Linguistics*, 30(1):75–93.

Sharon Goldwater, Thomas L. Griffiths, and Mark Johnson. 2009. A bayesian framework for word segmentation: Exploring the effects of context. *Cognition*, 112(1):21–54.

Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. 2004. Applying conditional random fields to Japanese morphological analysis. In *Proc. of EMNLP 2004*, pages 230–237.

Gakuto Kurata, Shinsuke Mori, and Masafumi Nishimura. 2006. Unsupervised adaptation of a stochastic language model using a Japanese raw corpus. In *Proc. of ICASSP 2006*, volume 1, pages 1037–1040.

Gakuto Kurata, Shinsuke Mori, Nobuyasu Itoh, and Masafumi Nishimura. 2007. Unsupervised lexicon acquisition from speech and text. In *Proc. of ICASSP 2007*, volume 4, pages 421–424.

Mikko Kurimo, Mathias Creutz, Matti Varjokallio, Ebru Arisoy, and Murat Saraçlar. 2006. Unsupervised segmentation of words into morphemes–Challenge 2005, an introduction and evaluation report. In *Proceedings of the PASCAL Challenge Workshop on Unsupervised Segmentation of Words into Morphemes*.

Sadao Kurohashi and Makoto Nagao. 1998. Building a Japanese parsed corpus while improving the parsing system. In *Proc. of LREC 1998*, pages 719–724.

Sadao Kurohashi, Toshihisa Nakamura, Yuji Matsumoto, and Makoto Nagao. 1994. Improvements of Japanese morphological analyzer JUMAN. In *Proc. of The Inter-*

national Workshop on Sharable Natural Language Resources, pages 22–38.

Daichi Mochihashi, Takeshi Yamada, and Naonori Ueda. 2009. Bayesian unsupervised word segmentation with nested Pitman-Yor language modeling. In *Proc. of ACL-IJCNLP 2009*, pages 100–108.

Shinsuke Mori and Makoto Nagao. 1996. Word extraction from corpora and its part-of-speech estimation using distributional analysis. In *Proc. of COLING 1996*, volume 2, pages 1119–1122.

Yugo Murawaki and Sadao Kurohashi. 2008. Online acquisition of Japanese unknown morphemes using morphological constraints. In *Proc. of EMNLP 2008*, pages 429–437.

Masaaki Nagata. 1999. A part of speech estimation method for Japanese unknown words using a statistical model of morphology and context. In *Proc. of ACL 1999*, pages 277–284.

Tetsuji Nakagawa. 2004. Chinese and Japanese word segmentation using word-level and character-level information. In *Proc. of COLING 2004*, pages 466–472.

Hoifung Poon, Colin Cherry, and Kristina Toutanova. 2009. Unsupervised morphological segmentation with log-linear models. In *Proc. of NAACL 2009*, pages 209–217.

Tetsuro Sasada, Shinsuke Mori, and Tatsuya Kawahara. 2008. Extracting word-pronunciation pairs from comparable set of text and speech. In *Proc. of INTERSPEECH 2008*, pages 1821–1824.

Keiji Shinzato, Tomohide Shibata, Daisuke Kawahara, Chikara Hashimoto, and Sadao Kurohashi. 2008. TSUBAKI: An open search engine infrastructure for developing new information access methodology. In *Proc. of IJCNLP-08*, pages 189–196.

Kiyotaka Uchimoto, Satoshi Sekine, and Hitoshi Isahara. 2001. The unknown word problem: a morphological analysis of Japanese using maximum entropy aided by a dictionary. In *Proc. of EMNLP 2001*, pages 91–99.

Jia Xu, Jianfeng Gao, Kristina Toutanova, and Hermann Ney. 2008. Bayesian semi-supervised Chinese word segmentation for statistical machine translation. In *Proc. of COLING 2008*, pages 1017–1024.

Toshio Yokoi. 1995. The EDR electronic dictionary. *Communications of the ACM*, 38(11):42–44.

Hai Zhao and Chunyu Kit. 2008. An empirical comparison of goodness measures for unsupervised Chinese word segmentation with a unified framework. In *Proc. of IJCNLP 2008*, pages 9–16.