Generating Diverse Katakana Variants Based on Phonemic Mapping

Kazuhiro Seki
Kobe University
1-1 Rokkodai, Nada, Kobe
657-8501, Japan
seki@cs.kobe-u.ac.jp

Hiroyuki Hattori
Google Inc.
26-1 Sakuraoka, Shibuya
150-8512, Japan
hattori@ai.cs.kobe-u.ac.jp

Kuniaki Uehara
Kobe University
1-1 Rokkodai, Nada, Kobe
657-8501, Japan
uehara@kobe-u.ac.jp

ABSTRACT

In Japanese, it is quite common for the same word to be written in several different ways. This is especially true for katakana words which are typically used for transliterating foreign languages. This ambiguity becomes critical for automatic processing such as information retrieval (IR). To tackle this problem, we propose a simple but effective approach to generating katakana variants by considering phonemic representation of the original language for a given word. The proposed approach is evaluated through an assessment of the variants it generates. Also, the impact of the generated variants on IR is studied in comparison to an existing approach using katakana rewriting rules.

Categories and Subject Descriptors

H.3.3 [**Information storage and retrieval**]: Information Search and Retrieval—*Query formulation*; I.2.7 [**Artificial intelligence**]: Natural Language Processing—*Language models*

General Terms

Algorithm, Experimentation, Languages

Keywords

Information Retrieval, Katakana Variants, Transliteration

1. INTRODUCTION

The Japanese language uses three different types of characters, i.e., kanji, hiragana, and katakana. Kanji characters are ideograms borrowed from Chinese, whereas hiragana and katakana are phonograms representing sounds. Among them, katakana is distinctive in a sense that it is often used for transliterating foreign words. An example is ディテール (*diteeru*) for an English word "detail."

These katakana words often have several notational variants. For instance, ディテール could be also written as ディテイル (*diteiru*), デテール (*deteeru*), etc. This variation becomes critical for automatic processing such as IR. Ideally, all the variants should be associated with a single concept corresponding to the original term "detail." In other words, given any variant as a query, an ideal IR system should also retrieve documents containing other variants. A similar problem is spelling correction, which changes a given word to its correct form when misspelled (e.g., Alisha Keys → Alicia Keys). The problem of katakana variants is different from spelling correction in that there is no definite canonical form for the former; all the variants are valid even though some may be preferred.

There are roughly two approaches to the katakana variant problem in the context of IR. One is to normalize all the variants to a single form in both indexing and searching [3]. This way, those variants are no longer distinguished and are treated as the same concept. The other is to obtain variants for a given query in searching [2]. This approach typically utilizes a precompiled variant dictionary and/or katakana rewriting rules based on surface characters (e.g., エー (*ee*) → エイ (*ei*)). The obtained variants are used for expanding the original query. This work focuses on the latter and proposes a novel approach for generating diverse katakana variants for a given katakana word through "backward-forward transliteration" described shortly.

2. KATAKANA VARIANT GENERATION

One of the factors that yield katakana variants is the different speech sounds between foreign languages and Japanese. Some phonemes used in a non-Japanese language are not used in Japanese, and vice versa. For example, English /æ/ does not exactly correspond to any Japanese vowels, and thus, when heard by Japanese speakers, it could be understood as any Japanese sounds close to it, such as /a/ and /e/, resulting in notational variants. Based on the observation that katakana orthographic variation is partially influenced by the speech sounds of the original language, we generate katakana variants by transliterating backward a katakana word of foreign origin and subsequently by transliterating it back to Japanese. In the following, we restrict ourselves to English loanwords but in theory the approach could be applied to other languages.

2.1 Backward-forward transliteration

Given a katakana word, we first convert it to the Latin alphabets (i.e., romanization), which is then partitioned into a sequence of Japanese sounds that have approximate English equivalents. Because a Japanese sound sequence can be split into different units corresponding to different English sounds, there are usually multiple ways of partitioning. For instance, Japanese sound "*ru*" as in "*diteeru*" can be either treated as a single unit "*ru*" or two separate units "*r*" and "*u*," each corresponding to different English sounds.

For each sequence of partitions (e.g., *d-i-t-ee-ru*) derived from the previous step, there are again multiple corresponding English sound sequences. In probability theory, given a Japanese sound sequence $J = j_1 \ldots j_n$ (where j_i represents a sound unit), the most likely English sound sequence $\hat{E} = e_1 \ldots e_n$ can be defined as in:

$$\begin{aligned}\hat{E} &= \arg\max_{E} P(E|J) \\ &= \arg\max_{E} P(J|E) \cdot P(E)\end{aligned} \tag{1}$$

SIGIR'08, August 20–24, 2008, Singapore.
ACM 978-1-60558-164-4/08/07.

Assuming the independence among Japanese sounds and 1st-order Markov model, the following equation holds:

$$\arg\max_{E} P(J|E) \cdot P(E) = \prod_{i} P(j_i|e_i) \cdot P(e_i|e_{i-1}) \tag{2}$$

where we define $P(e_1|e_0) = P(e_1)$ for notational convenience. The first and second factors in the right-hand side of the equation correspond to symbol emission and state transition probabilities, respectively, in Hidden Markov Models.

As the probability estimates for $P(j_i|e_i)$, we used mapping probabilities derived from 8,000 English-Japanese pairs [1], whereas for $P(e_i|e_{i-1})$ we used 127,000 English sound sequences from the CMU dictionary.[1]

Given the most likely English sound sequence $\hat{E}$ obtained in the previous step, we transform it back to Japanese sound sequences (J') and subsequently to katakana words (K'), where each J' is uniquely converted to a katakana word K'. Although each resulting K' is potentially a variant of the original katakana word, there are typically many K's that are invalid. We filter them out based on two criteria. One is a probability that $\hat{E}$ matches to J' and consequently K' is generated, defined as $P(K') \cdot \prod_i P(j'_i|\hat{e}_i)$. The first factor can be estimated based on a character-based *n*-gram model. In this study, we set *n*=3 and built the model on 13,124 katakana words in the EDICT dictionary.[2] (See Equation (2) for the second factor.) Another criterion is the the actual use of K' as a katakana word. We regard the web as a large corpus and use the Yahoo! API to obtain the document frequency of K' (i.e., the number of hit by query K'). If K' is actually used in at least one web page, we consider it to be legitimate. For the previous example of ディテール (*diteeru*), our approach generated as variants, ディテイル (*diteiru*), デテール (*deteeru*), ディテル (*diteru*), and ディタル (*ditaru*), etc.; some seem to be valid and the others may not.

3. EVALUATION

3.1 Overview

We evaluated our proposed approach (denoted as *Phone*) from two aspects: the quality of the generated variants (denoted as *V*) and their effect on IR. For the former, we asked 17 human evaluators to judge if *V*'s were actually valid variants. For the latter, we used *V*'s for query expansion and asked the same evaluators to judge whether *V*'s were used in place of the original katakana words in the retrieved documents. We expect that the latter situation is more forgiving because even if a *V* is not commonly used as a variant, it may still be used by some authors where notation is barely controlled, as in the case of the web.

For evaluation, we chose 25 katakana queries, including メーキャップアーチスト (makeup artist), グラフィクス (graphics)，シャキラ (Shakira), etc., that were likely to have notational variants.

3.2 Assessment of generated variants

For each of the 25 katakana words, we generated variants, *V*'s, by *Phone*. Each evaluator judged each *V* to be one of *acceptable*, *weakly acceptable*, and *not acceptable*. Among 293 variants generated in total, 195 (66.6%) were judged as either acceptable or weakly acceptable by at least one evaluator. In other words, two-thirds of the generated variants were found to be possibly valid. Also, among the 195 variants, 175 (89.2%) could not be generated by an approach using katakana rewriting rules (see Section 3.3). This result indicates the ability of our proposed approach for generating diverse katakana variants that cannot be produced by existing approaches relying solely on surface clues.

3.3 Effect of generated variants on IR

In this section, we used the generated variants for the 25 katakana words as additional query terms for an IR system so as to evaluate its potential impact on IR. Here, we used the Yahoo! web search engine via the API as the base IR system.

First, we looked at how much more information could be found by using the 195 variants judged as either acceptable or weakly acceptable. Because all the 195 variants used in this experiment are potentially valid, the retrieved documents are also potentially relevant. For the 25 katakana queries, we performed searches with or without query expansion (QE), and the mean ratio of the number of retrieved documents with QE to that without QE was 60.1—approximately 60 times more documents were located, which would substantially improve recall.

Then, using all the 293 variants generated, we compared our proposed approach, *Phone*, and an existing approach using katakana rewriting rules, denoted as *Rule*. The rules for *Rule* were automatically acquired as frequent rewritten patterns from 750 pairs of katakana variants found in EDICT. For the 25 katakana words, *Rule* generated 172 variants in total. For this evaluation, we created a gold standard by manually judging the relevance of top 20 documents retrieved by *Rule* or *Phone*, and measured mean average precision (MAP) for top 100 retrieved documents. Despite the fact that *Phone* generated 1.7 (=293/172) times more variants than *Rule*—which could have lowered precision, *Phone* achieved 37% greater MAP (0.164) than *Rule* (0.120).

While the generated variants showed positive effects on IR in the above experiments overall, there were also cases where they caused a problem. Besides completely false variants, a problem observed was that the generated variant had a different meaning in popular use. For instance, "メイデー (*meidee*)" was generated as a variant of "メーデー (*meedee*)" meaning English "May Day." The variant appears to be valid but has a different meaning—a popular music title, resulting in many documents related to the music. Because it is difficult to guess users' intention as to which sense they refer to, the ambiguity may need to be interactively resolved.

4. CONCLUSION

We proposed a novel approach for generating katakana variants through backward-forward transliteration based on Japanese-English phonemic mapping. Although transliteration is not a new area of study, it has never been applied to katakana variant generation. Experimental results showed that among the 293 variants generated for 25 katakana words by our approach, 195 were found potentially valid, of which 175 could not be generated by an existing katakana rewriting rule-based approach, indicating the advantage of our approach for generating diverse variants. Also, when applied to IR, it achieved 37% greater MAP than the rule-based approach.

5. REFERENCES

[1] K. Knight and J. Graehl. Machine transliteration. *Computational Linguistics*, 24(4):599–612, 1998.

[2] C. Kubomura and H. Kameda. Information retrieval system with abilities of processing katakana allographs. *IEICE Trans. Inf. & Syst.*, J86-D-II(3):418–428, 2003. (In Japanese)

[3] M. Shishibori, K. Tsuda, and J. Aoe. A method for generation and normalization of katakana variant notations. *IEICE Trans. Info. & Syst.*, J77-D-II(2):380–387, 1994. (In Japanese)

[1] `http://www.speech.cs.cmu.edu/cgi-bin/cmudict`

[2] `http://www.csse.monash.edu.au/~jwb/j_edict.html`