Support Vector Machine Based Orthographic Disambiguation

Eiji ARAMAKI    Takeshi IMAI    Kengo Miyo    Kazuhiko Ohe
University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8655, Japan
aramaki@hcc.h.u-tokyo.ac.jp

Abstract
Orthographic variation can be a serious problem for many natural language-processing applications. Japanese in particular contains orthographic variation, because the large quantity of transliteration from other languages causes many possible spelling variations. To manage this problem, this paper proposes a support vector machine (SVM)-based classifier that can determine whether two terms are equivalent. We automatically collected both positive examples (sets of equivalent term pairs) and negative examples (sets of inequivalent term pairs). Experimental results yielded high levels of accuracy (87.8%), demonstrating the feasibility of the proposed approach.

1 Introduction
Orthographic variation can be a serious problem for many natural language-processing (NLP) applications, such as information extraction (IE), question answering (QA), and machine translation (MT). For example, many example-based machine translation (EBMT) (Nagao, 1984) methods, such as (Somers, 1999; Richardson et al., 2001; Sumita, 2001; Carl and Way, 2003; Aramaki and Kurohashi, 2004; Nakazawa et al., 2006), utilize a translation dictionary during bilingual text alignment. Also, several statistical machine translation (SMT) (Brown et al., 1993) methods set initial translation parameters using a translation dictionary. When consulting a dictionary, a system must disambiguate orthographic variation.

The following terms are an example of Japanese orthographic variation, corresponding to the term “Avogadro’s number”:

1. アヴォガドロ数
   (A VO GA DO RO SU),
2. アボガドロ数
   (A BO GA DO RO SU).

Although both terms are frequently used (term (1) resulted in 25,700 Google hits and Term (2) resulted in 25,000 Google hits¹), translation dictionaries contain only one of the terms, resulting in low levels of accuracy with dictionary-based bilingual text alignment.

This paper focuses on Japanese orthographic disambiguation. Japanese orthographic variance is closely related to transliteration, because transliteration relies on pronunciation, the great differences between the sounds made in Japanese and in Western languages (mainly English) results in a variety of possible spellings.

Researchers have already proposed methods to solve this problem. For example, Knight (1998) developed a back-transliteration method using a probabilistic

¹We got the results on May 14, 2007.
model. Goto et al. (2004) also developed a probabilistic model, which takes into account surrounding context. Lin and Chen (2002) developed a perceptron learning algorithm for back-transliteration. While these methods differ, they all share the same goal: being able to back-transliterate a given term into another language.

By contrast, this paper proposes a new task schema: given two Japanese terms, the system determines whether they are equivalent. Figure 1 illustrates our task schema: a foreign term can be transliterated into Japanese in several ways. While previous methods can yield suitable back-transliteration for a term, our system determines whether a pair of Japanese terms originates from the same foreign word. We expect our task-setting is more direct and practical for many applications, such as dictionary consulting in MT, IE, and so on.

For this process, our proposed method uses a machine learning technique (support vector machine, hereafter SVM (Vapnik, 1999)), which requires the two following types of data:

1. Positive examples: a term pair, which are spelled differently, but have the same meaning; and,

2. Negative examples: a term pair, which are spelled differently and have differing meanings.

While previous methods have utilized only positive examples, our proposed method also
incorporates negative examples. Both examples can be generated automatically from translation dictionaries using spelling similarity and heuristic rules.

Experimental results yielded high accuracy (87.8%), demonstrating the feasibility of the proposed approach.

Although we investigated the performance in the medical terms, the proposed method does not depend on the target domain.

Section 2 of this paper describes how training data are built. Section 3 describes the learning method, and Section 4 presents the experimental results. Section 5 discusses related work, and Section 6 presents our conclusions.

2 Automatic Example Building

This section describes how training data are built; Section 2.1 discusses positive examples, and Section 2.2 discusses negative examples. Note that the latter is a novel task.

2.1 Positive Examples

Our method uses a standard approach to extract positive examples. The basic idea is that orthographic variants should (1) have similar spelling, and (2) share the same English translation.

The method consists of the following two steps:

STEP 1: First, using two or more translation dictionaries, we extract a set of Japanese terms with the same English translation.

STEP 2: Then, for each extracted set, we generate possible two term pairs \((\text{term}_1 \text{ and } \text{term}_2)\), and calculate the spelling similarity between them. Spelling similarity is measured using the following edit-distance based similarity \(SIM(\text{term}_1, \text{term}_2)\):

\[
SIM(\text{term}_1, \text{term}_2) = 1 - \frac{\text{EditDistance}(\text{term}_1, \text{term}_2) \times 2}{\text{len}(\text{term}_1) + \text{len}(\text{term}_2)},
\]

where \(\text{len}(\text{term}_1)\) is the length (the number of characters) of \(\text{term}_1\), \(\text{len}(\text{term}_2)\) is the length (the number of characters) of \(\text{term}_2\), \(\text{EditDistance}(\text{term}_1, \text{term}_2)\) is the minimum number of point mutations required to change \(\text{term}_1\) into \(\text{term}_2\), where a point mutation is one of: (1) a change in a character, (2) the insertion of a character, and (3) the deletion of a character. For details, see (Levenshtein, 1965).

Any term pair with more than a threshold (TH) similarity is considered a positive example \(^2\).

2.2 Negative Examples

As mentioned in Section 1, generating negative examples is a novel process in this field.

One simple way is to select two words from a dictionary randomly. However, such a simple method would generate a huge quantity of meaningless examples. Therefore, as in our collection of positive examples, we collected only term pairs with similar spellings.

Another problem is a balance of the example quantity. In the preliminary experiments, the number of negative examples was about three times as the number positive examples, leading to a negative bias.

Therefore, we investigated the Google hits of each term pair by using a query, such as “アヴォガドロ 数 アボガドロ 数”.

Then, we utilize only negative examples with many Google hits, and reject low-hits examples, because of the following two reasons:

1. **Popularity**: We expect that a more popular term pair is more informative.

2. **Reliability**: We hypothesize that an orthographic pair rarely appears in one document, because one document usually has an orthographic consistency. Therefore, we can expect that if two terms co-occur in one document, they are not orthographic variants, ensuring reliability for negative examples.

The detailed steps are as follows:

\(^2\)We set \(TH = 0.8\).
STEP 1: First, using two or more translation dictionaries, we extract a set of Japanese terms with different English translations.

STEP 2: Then, for each extracted set, we generate possible pairs, and calculate the spelling similarity between them. Any term pair exceeding a threshold \((TH)\) similarity is considered a negative example candidate.

STEP 3: Finally, we investigate the Google hits for each candidate. We only use the top \(K\)-hit candidates as negative examples\(^3\).

3 Leaning Method

Application of the method described in Section 2 yields training data, consisting of triple expressions \(<\text{term}_1, \text{term}_2, +1/−1>\), in which “+1” indicates a positive example (orthographic variants), and “−1” indicates a negative example (different terms). Table 1 provides some examples.

The next problem is how to convert training data into machine learning features. We regard the different parts and context (window size ±1) as features:

1. **DIFF**: differing characters between two translations;
2. **PRE-CONTEXT**: previous character of DIFF; and
3. **POST-CONTEXT**: subsequent character of DIFF.

Figure 2 provides examples of these features. Since the different part is a gray area (“VO(ヴォ)” and “BO(ボ)”), we consider DIFF to be “VO:BO (ヴォ:ボ)” itself, PRE-CONTEXT to be “A (ア)” in a dotted box, and POST-CONTEXT to be “GA (ガ)” also in a dotted box.

Figure 3 provides another example; the insertion/deletion of a character can be considered the DIFF using \(\phi\), such as “\(\phi\) A (\(\phi\) −)”. In addition, the start \([\text{SOT}]\) or end \([\text{EOT}]\) of a term can be considered a character.

Note that both PRE-CONTEXT and POST-CONTEXT consist of one character pair, while the DIFF can be a pair of \(n : m\) characters \((n \geq 0, m \geq 0)\).

In learning, we can use a back-off technique to prevent problems related to data sparseness. As a result, each different point utilizes the following four features:

- **DIFF + PRE-CONTEXT + POST-CONTEXT**
- **(1-back-off-a) DIFF + PRE-CONTEXT**
- **(1-back-off-b) DIFF + POST-CONTEXT**
- **(2-back-off) DIFF**

Figure 4 presents some examples.

4 Experiments

4.1 Test-set

To evaluate the performance of our system, we manually built a test-set as follows:

First, we extracted 5,013 similar spelling term pairs, that have more than \((SIM > 0.8)\), from two dictionaries (Nanzando, 2001b), Ito et al., 2003).

Then, for each pair, we annotated whether it is an equivalent pair (orthographic variants) or not (different terms).

Finally, we randomly extracted 883 pairs form it. We regard it as a test-set. The test-set consists of 312 positive examples and 571 negative examples. The others (4,130 examples) are used for training in comparative methods (BYHAND and COMBINATION mentioned in Section 4.3).

4.2 Training-set

By using the proposed method (in Section 2), we automatically built a training-set from two translation dictionaries (Japan Medical Terminology English-Japanese(Nanzando, 2001a) and 25-thousand-terms Medical Dictionary(MEID, 2005)). As a result, we got a training-set, consisting of 68,608 examples (21,380 positive examples and 47,228 negative examples).
<table>
<thead>
<tr>
<th>P/N*</th>
<th>Term₁</th>
<th>Term₂</th>
</tr>
</thead>
<tbody>
<tr>
<td>+1</td>
<td>ヨードヒラセット (YO O DO PI RA SE TTO; iodopyracet)</td>
<td>ヨードヒラセット (YO O DO PI RA SE TO; iodopyracet)</td>
</tr>
<tr>
<td>+1</td>
<td>マイクロメーター (MA I KU RO ME E TA A; micrometer)</td>
<td>マイクロメータ (MA I KU RO ME E TA; micrometer)</td>
</tr>
<tr>
<td>+1</td>
<td>アンプリファイア (A N PU RI FA I A; amplifier)</td>
<td>アンプリファイア (A N PU RI FA I YA A; amplifier)</td>
</tr>
<tr>
<td>+1</td>
<td>オシロスコープ (O SI RO SU KO O PU; oscilloscope)</td>
<td>オシロスコープ (O SSI RO SU KO O PU; oscilloscope)</td>
</tr>
<tr>
<td>+1</td>
<td>動的コンプライアンス (DO U KO N PU RA I A N SU; dynamic compliance)</td>
<td>動的コンプライアンス (DO U TE KI KO N PU RA I A N SU; dynamic compliance)</td>
</tr>
<tr>
<td>+1</td>
<td>湿潤圧性ショック (SI N TO O A TU SE I SYO KKU; osmotic shock)</td>
<td>湿潤圧性ショック (SI N TO O A TU SYO KKU; osmotic shock)</td>
</tr>
<tr>
<td>+1</td>
<td>マルブルグウイルス (MA R U BU RU GU U I RU SU; Marburg virus)</td>
<td>マルブルグウイルス (MA RU BU RU GU U I RU SU; Marburg virus)</td>
</tr>
<tr>
<td>+1</td>
<td>ドールトンの法則 (DO O R U TO N NO HO O SO KU; Dalton law)</td>
<td>ドールトンの法則 (DO RU TO N NO HO O SO KU; Dalton law)</td>
</tr>
<tr>
<td>-1</td>
<td>型肝炎 (BI I GA TA KA N E N; hepatitis B)</td>
<td>型肝炎 (SI I GA TA KA N E N; hepatitis C)</td>
</tr>
<tr>
<td>-1</td>
<td>ドランス (TO RA N SU; trance)</td>
<td>ドランススタ (TO RA N JI SU TA; transistor)</td>
</tr>
<tr>
<td>-1</td>
<td>ビタミン (BI TA MI N PI I; vitamin P)</td>
<td>ビタミン (BI TA MI N SI I; vitamin C)</td>
</tr>
<tr>
<td>-1</td>
<td>カルドミウム (KA DO MI U MU; cadmium)</td>
<td>カルドミウム (KA RU SI U MU; calcium)</td>
</tr>
<tr>
<td>-1</td>
<td>グルコース (A RU KO O RU; alcohol)</td>
<td>グルコース (GU RU KO O SU; glucose)</td>
</tr>
<tr>
<td>-1</td>
<td>メラトニン (ME RA TO NI N; melatonin)</td>
<td>セロトニン (SE RA TO NI N; serotonin)</td>
</tr>
<tr>
<td>-1</td>
<td>グローニ (KU RO O N; clone)</td>
<td>グローニ (KU RA A RE; curare)</td>
</tr>
<tr>
<td>-1</td>
<td>ケトン生成 (KE TO N SE E I SE I; ketogenesis)</td>
<td>ケトン生成 (ME TA N SE I SE I; methanation)</td>
</tr>
<tr>
<td>-1</td>
<td>リード指数 (RI I DO SI SU U; Reid index)</td>
<td>リード指数 (RI BI I SI SU U; Livi index)</td>
</tr>
<tr>
<td>-1</td>
<td>ヘマチン (TO MA CHI N; tomatine)</td>
<td>ヘマチン (HE MA CHI N; haematin)</td>
</tr>
<tr>
<td>-1</td>
<td>バルーン法 (BA RU U N HO; balloon method)</td>
<td>バルーン法 (RA GU U N HO; lagoon method)</td>
</tr>
</tbody>
</table>

Table 1: Some Examples of Training-set.

* “+1” indicates positive examples, and “-1” indicates negative examples.
4.3 Comparative Methods

We compared the following methods:

1. **EDITDISTANCE**(TH): an edit-distance-based method, which regards an example with a spelling similarity \( \text{SIM}(\text{term}_1, \text{term}_2) > \text{TH} \) as an orthographic variants. The performance of this method changes, depending on \( \text{TH} \).

2. **BYHAND**: a SVM-based method, trained by manually annotated corpus, consists of 4,130 examples.

3. **AUTOMATIC**: a SVM-based method, trained by an automatically build training-set.

4. **COMBINATION**: a SVM-based method, trained by both BYHAND corpus and AUTOMATIC corpus.

For SVM learning, we used TinySVM\(^4\) with a linear kernel\(^5\).

4.4 Evaluation

To evaluate our method, we used three measures, precision, recall and accuracy, defined as follows:

\[
\text{Precision} = \frac{\text{# of pairs found and correct}}{\text{total # of pairs found}},
\]

\[
\text{Recall} = \frac{\text{# of pairs found and correct}}{\text{total # of pairs correct}},
\]

\[
\text{Accuracy} = \frac{\text{# of pairs correct}}{\text{total # of pairs in test-set}}.
\]

4.5 Results

First, we checked the performance of EDITDISTANCE(TH) in various \( \text{TH} \) values. Figure 5 presents the results. While the precision is basically proportional to the spelling similarity \( \text{(TH)} \), it drops down in the high \( \text{TH} \ (\text{TH} \div 0.96) \), indicating a highly similar spelling term pair not always have to be the orthographic variants.

Table 2 presents the performance of all methods. AUTOMATIC did not obtain a higher accuracy than BYHAND, the combination of them is the highest accuracy, demonstrating the basic feasibility of our approach. The precision-recall graph (Figure 6) also shows the advantage of COMBINATION.

4.6 Error Analysis

We investigated the errors from COMBINATION, and found that many errors came from...
a verbal omission, which is different phenomenon from transliteration.

For example, a test-set has the following positive example:

1. カルシウム・チャネル
(calcium channel; KA RU SI U MU CHA NE RU),

2. カルシウムイオン・チャネル
(calcium ion channel; KA RU SI U MU I O N CHA NE RU).

Because a term “ion” is without saying inferable in this case, it can be omitted. Capturing such an operation requires a very high level of understanding of the meaning of the terms.

To focus on a transliteration problem, we manually removed such examples from our test-set, and built a sub-set of it, consisting of only transliterations. The result is shown in Table 3. The accuracy of COMBINATION is higher than 90%.

It is difficult to compare this accuracy to that of the previous studies because (1) their corpus were different from ours and (2) previous studies focused on back-transliteration. However, we can say that the present accuracy is, at least, not behind from the previous researchers (64% by (Knight and Graehl, 1998) and 87.7% by (Goto et al., 2004)). We expect that the present accuracy is practical in many applications.

Finally, we investigate the differences between AUTOMATIC and BYHAND results (the AUTOMATIC accuracy is much lower than the BYHAND by 8.5 points in Table 2). One of the reasons is dictionary specific styles, such as numerous expression variants (“8.8.8.8.8.8.8.8.8.8 (Japanese number expression)”)

hyphenation variants (“-,.-,=,≠,”) and so on. Because the BYHAND training-set and the test-set came from the same dictionaries, BYHAND already knows such variants are meaningless differences. However, AUTOMATIC, using different dictionaries, sometimes suffered from unseen number expression/hyphenation variants.

Note that in transliteration accuracy (in Table 3), their accuracies (BYHAND and AUTOMATIC) are not so different.
Table 2: Results

<table>
<thead>
<tr>
<th>methods</th>
<th>Precision</th>
<th>Recall</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>EDIT-DISTANCE(0.91)</td>
<td>67.2%</td>
<td>52.6%</td>
<td>70.9%</td>
</tr>
<tr>
<td>BYHAND</td>
<td>70.4%</td>
<td>88.4%</td>
<td>82.7%</td>
</tr>
<tr>
<td>AUTOMATIC</td>
<td>65.7%</td>
<td>56.7%</td>
<td>74.2%</td>
</tr>
<tr>
<td>COMBINATION</td>
<td>82.9%</td>
<td>82.6%</td>
<td>87.8%</td>
</tr>
</tbody>
</table>

* The performance in EDIT-DISTANCE(0.91) showed the highest accuracy in various TH values.

Table 3: Results of a sub-set (Transliteration Only)

<table>
<thead>
<tr>
<th>methods</th>
<th>Precision</th>
<th>Recall</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>BYHAND</td>
<td>67.7%</td>
<td>91.0%</td>
<td>80.3%</td>
</tr>
<tr>
<td>AUTOMATIC</td>
<td>77.3%</td>
<td>81.3%</td>
<td>83.9%</td>
</tr>
<tr>
<td>COMBINATION</td>
<td>90.6%</td>
<td>90.7%</td>
<td>91.9%</td>
</tr>
</tbody>
</table>

5 Related Works

As noted in Section 1, transliteration is the field most relevant to our work, because many orthographic variations come from borrowed words. Our proposed method differs from previous studies in the following three ways: (1) task setting, (2) negative examples, and (3) target scope.

5.1 Task Setting

Most previous studies have involved finding the most suitable back-transliteration of a term.

For example, given an observed Japanese string $\sigma$ by optical character recognition (OCR) software, Knight and Graehl (1998) finds a suitable English word $w$. For this process, they developed a probabilistic model that decomposed a transliteration into sub-operations as follows:

$$P(w)P(e|w)P(j|e)P(k|j)P(o|k),$$

where $P(w)$ generates written English word sequences, $P(e|w)$ pronounces English word sequences, $P(j|e)$ converts English sounds into Japanese sounds, $P(k|j)$ converts Japanese sounds to KATAKANA writing, and $P(o|k)$ introduces misspellings caused by OCR.


Such methods are not only applicable to Japanese; it can also be used for Arabic(Stalls and Knight, 1998; Sherif and Kondrak, 2007), Chinese(Li et al., 2007), Persian(Karimi et al., 2007).

The task-setting involved in our method differs from previous methods. Our methodology involves determining whether two terms in the same language are equivalent, making our task-setting more direct and suitable than previous methods for many applications, such as dictionary consulting in MT and information retrieval.

Note that Yoon et al.(2007) also proposed a discriminative transliteration method, but their system determines whether a target term is transliterated from a source term or not.

5.2 Negative Examples

Our task setting requires negative examples, consisting of term pairs with similar spellings, but different meanings.

By contrast, previous research involved only positive examples. For example, Masuyama et al.(2004) collected 178,569 Japanese transliteration variants (positive examples) from large corpora. However, they paid little attention to negative examples.
5.3 Target Scope
As mentioned above, orthographic variation in Japanese results mainly from transliteration. However, our target includes several different phenomena, such as verbal omissions mentioned in Section 4.6. Although the accuracy for omissions is not enough, our method addresses it easily, while previous methods are unable to handle this kind of phenomenon.

6 Conclusion
In this paper, we proposed a SVM-based orthographic disambiguation method. We also proposed a method for collecting both positive and negative examples. Experimental results yielded high levels of accuracy (87.8%), demonstrating the feasibility of the proposed approach.

Acknowledgments
Part of this research is supported by Grant-in-Aid for Scientific Research of Japan Society for the Promotion of Science (Project Number:16200039, F.Y.2004-2007 and 18700133, F.Y.2006-2007) and the Research Collaboration Project (#04710001247) with Japan Anatomy Laboratory Co.Ltd.

References


