# Exploiting Lexical Resources for Disambiguating CJK and Arabic Orthographic Variants

**Jack Halpern (春遍雀來)**
The CJK Dictionary Institute (日中韓辭典研究所)
34-14, 2-chome, Tohoku, Niiza-shi, Saitama 352-0001, Japan
E-mail: jack@cjki.org

**Abstract**

The orthographical complexities of Chinese, Japanese, Korean (CJK) and Arabic pose a special challenge to developers of NLP applications. These difficulties are exacerbated by the lack of a standardized orthography in these languages, especially the highly irregular Japanese orthography and the ambiguities of the Arabic script. This paper focuses on CJK and Arabic orthographic variation and provides a brief analysis of the linguistic issues. The basic premise is that statistical methods by themselves are inadequate, and that linguistic knowledge supported by large-scale lexical databases should play a central role in achieving high accuracy in disambiguating and normalizing orthographic variants.

## 1. Introduction

Various factors contribute to the difficulties in CJK and Arabic information processing, especially in the areas of information retrieval (IR), named entity recognition (NER), machine translation (MT), word segmentation (WS) and automatic transcription, referred to as NLP applications below. Some of the major issues include:

1. The lack of a standard orthography. To process the extremely large number of orthographic variants (especially in Japanese) requires support for advanced methodology such as cross-orthographic searching (Halpern, 2003).
2. The accurate conversion between Simplified Chinese (SC) and Traditional Chinese (TC), deceptively simple but in fact extremely difficult (Halpern, Kerman, 1999).
3. Morphological complexity poses a formidable challenge to the development of accurate morphological analyzers that can perform operations like stemming, conflation, and POS tagging.
4. The difficulty of performing accurate word segmentation, which involves identifying word boundaries by breaking a text stream into semantic units for dictionary lookup and indexing purposes. Good progress in this area is reported (Emerson, 2000; Yu, et al., 2000).
5. Miscellaneous retrieval technologies such as synonym expansion and cross-language information retrieval (CLIR) (Goto, 2001).
6. Proper nouns pose special difficulties as they are extremely numerous, difficult to detect without a lexicon and have an unstable orthography (Halpern, 2006).
7. The Arabic orthography is ambiguous for various reasons: the omission of short vowels, multiple ways of writing long vowels, and complex hamza rules. Arabic is also highly ambiguous morphologically, so that a string can often represent multiple words (Halpern, 2007).

## 2. Lexicon Driven Approach

The various attempts to tackle these tasks using purely statistical and algorithmic methods have had only limited success (Kwok, 1997). Indeed Kay (2004) argues that "statistics are a surrogate for knowledge of the world" and that "this is an alarming trend that computational linguists ... should resist with great determination." However, an important motivation for statistical methods has been the poor availability and high cost of large-scale lexical databases. Our approach is to use in-depth linguistic knowledge combined with statistically based comprehensive lexicons because we maintain that ultimately statistical methods by themselves are inadequate for dealing with the multi-dimensional complexities of the CJK and Arabic scripts. This paper summarizes the issues in CJK and Arabic orthographic variation and argues that a lexicon-driven approach exploiting large-scale lexical databases can offer a reliable solution.

## 3. Chinese Orthographic Variants

### 3.1 Multiple Scripts

The complexity of the Chinese writing system is well known. Some factors contributing to this include the large number of characters in common use, their complex forms, the major differences between **Traditional Chinese** (TC) and **Simplified Chinese** (SC) along several dimensions and the occurrence of orthographic variants in TC.

### 3.2 Script Conversion

Automatically converting SC to/from TC, referred to as **C2C conversion**, is full of complexities (Halpern, Kerman, 1999) and technical difficulties (Lunde, 1999). The conversion can be implemented on three levels in increasing order of sophistication, briefly described below.

### 3.2.1 Code Conversion

The simplest, but least reliable, method is on a code point-to-code point basis by looking the source up in a mapping table. Because of the numerous one-to-many ambiguities, the rate of conversion failure is unacceptably high.

| SC | TC1 | TC2 | TC3 | TC4 | Remarks |
|---|---|---|---|---|---|
| 门 | 們 | | | | one-to-one |
| 汤 | 湯 | | | | one-to-one |
| 发 | 發 | 髮 | | | one-to-many |
| 干 | 幹 | 乾 | 干 | 榦 | one-to-many |

Table 1: Code conversion

### 3.2.2 Orthographic Conversion

A more sophisticated approach to C2C conversion is to process larger orthographic units, rather than code points in a character set; that is, meaningful linguistic units, especially multi-character lexemes. While code conversion is ambiguous, orthographic conversion gives better results because the mapping tables enable conversion on the word level.

| English | SC | TC1 | TC2 | Incorrect |
|---|---|---|---|---|
| telephone | 电话 | 電話 | | |
| we | 我们 | 我們 | | |
| start-off | 出发 | 出發 | | 出髮 齣髮 齣發 |
| dry | 干燥 | 乾燥 | | 干燥 幹燥 榦燥 |
| | 阴干 | 陰乾 | 陰干 | |

Table 2: Orthographic conversion

The ambiguities inherent in code conversion are resolved by using orthographic mapping tables, which avoids invalid conversions such as shown in the **Incorrect** column above. Because of segmentation ambiguities, such conversion must be done with the aid of a segmentor that can break the text stream into meaningful units (Emerson, 2000).

### 3.2.3 Lexemic Conversion

A more sophisticated, and far more challenging, approach to C2C conversion is to map SC and TC lexemes that are **semantically**, not orthographically, equivalent. For example, SC 信息 (*xìnxī* ) 'information' is converted to the semantically equivalent TC 資訊 (*zī xùn*). This is similar to the difference between *lorry* in British English and *truck* in American English.

There are numerous lexemic differences between SC and TC, especially in technical terms and proper nouns (Tsou, 2000). For example, there are more than 10 variants for *Osama bin Laden.* Moreover, the correct TC is sometimes locale-dependent. Lexemic conversion is the most difficult aspect of C2C conversion and can only be done with the help of mapping tables.

| English | SC | Taiwan TC | HK TC | Incorrect TC (orthographic) |
|---|---|---|---|---|
| Software | 软件 | 軟體 | 軟件 | 軟件 |
| Taxi | 出租汽车 | 計程車 | 的士 | 出租汽車 |
| Osama bin Laden | 奥萨马本拉登 | 奧薩瑪賓拉登 | 奧薩瑪賓拉丹 | 奧薩馬本拉登 |
| Oahu | 瓦胡岛 | 歐胡島 | | 瓦胡島 |

Table 3: Lexemic conversion

### 3.2.4 Character Form Variants

Traditional Chinese has numerous variant character forms. Disambiguating these variants can be done by using mapping tables such as the one shown below. If such a table is carefully designed limiting it to cases of 100% semantic interchangeability for polysemes, it is easy to normalize a TC text by trivially replacing variants by their standard forms. For this to work, all relevant components, such as MT dictionaries, search engine indexes and the related documents should be normalized. An extra complication is that Taiwanese and Hong Kong variants are sometimes different (Lunde, 1999).

| Var. 1 | Var. 2 | English | Comment |
|---|---|---|---|
| 裏 | 裡 | inside | 100% interchangeable |
| 教 | 教 | teach | 100% interchangeable |
| 著 | 着 | particle | variant 2 not in Big5 |
| 為 | 爲 | for | variant 2 not in Big5 |
| 沉 | 沈 | sink; surname | partially interchangeable |
| 泄 | 洩 | leak; divulge | partially interchangeable |

Table 4: TC variants

## 4. Japanese Orthographic Variants

### 4.1 Variation Across Four Scripts

The Japanese orthography is highly irregular. Because of the large number of orthographic variants and easily confused homophones, the Japanese writing system is significantly more complex than any other major language, including Chinese. A major factor is the complex interaction of the four scripts, resulting in countless words that can be written in a variety of often unpredictable ways (Halpern, 2003).

Japanese is also a highly agglutinative language. Verbs can have numerous inflected and derived forms (tens of thousands), Japanese NLP applications must be capable of performing stemming, i.e. be capable of recognizing that 書き著さない is the negative form of 書き著す, and must be able to identify the many variations in inflected forms, such as 書き著わさない , 書著さない, and 書き著さない .

Table 5 shows the orthographic variants of 取り扱い *toriatsukai* 'handling', illustrating a variety of variation patterns.

| ***Toriatsukai*** | **Type of variant** |
|---|---|
| 取り扱い | "standard" form |
| 取扱い | okurigana variant |
| 取扱 | All kanji |
| とり扱い | replace kanji with hiragana |
| 取りあつかい | replace kanji with hiragana |
| とりあつかい | All hiragana |

Table 5: Variants of *toriatsukai*

An example of how complex this can get is the proverbial "A hen that lays golden eggs." The "standard" orthography would be 金の卵を産む鶏 (*Kin no tamago wo umu niwatori*). In reality, *tamago* 'egg' has four variants (卵, 玉子, たまご, タマゴ), *niwatori* 'chicken' three (鶏, にわとり, ニワトリ) and *umu* 'to lay' two (産む, 生む), which expands to 24 permutations like 金の卵を生むニワトリ, 金の玉子を産む鶏 etc. As can be easily verified by searching the web, these variants frequently occur in web pages. Clearly, the user has no hope of finding them unless the application supports orthographic disambiguation.

Linguistic tools that perform segmentation, MT, entity extraction and the like must identify and/or normalize such variants to perform dictionary lookup. Below is a brief discussion of the variant types and how such normalization can be achieved.

### 4.2 Okurigana Variants

One of the most common types of orthographic variation in Japanese occurs in kana endings, called 送り仮名 *okurigana*, that are attached to a kanji base or stem. Okurigana variants are numerous and unpredictable. Identifying them must play a major role in Japanese orthographic normalization. The most effective solution is to use a lexicon of okurigana variants, such as the one shown below:

| **English** | **Reading** | **Standard** | **Variants** |
|---|---|---|---|
| publish | *kakiarawasu* | 書き著す | 書き著わす, 書著わす, 書著す |
| perform | *okonau* | 行う | 行なう |
| handling | *toriatsukai* | 取り扱い | 取扱い, 取扱 |

Table 6: Okurigana variants

### 4.3 Cross-Script Variants

Japanese is written in a mixture of four scripts: **kanji** (Chinese characters), two syllabic scripts called **hiragana** and **katakana**, and **romaji** (the Latin alphabet) (Halpern, 2006). Orthographic variation across scripts, as illustrated in Table 7, is extremely common and mostly unpredictable, so that the same word can be written in hiragana, katakana or kanji, or even in a mixture of two scripts.

| | |
|---|---|
| Kanji vs. Hiragana | 大勢　おおぜい |
| Kanji vs. Katakana | 硫黄　イオウ |
| Kanji vs. hiragana vs. katakana | 猫　ねこ　ネコ |
| Katakana vs. hybrid | ワイシャツ　Yシャツ |
| Kanji vs. katakana vs. hybrid | 皮膚　ヒフ　皮フ |
| Kanji vs. hybrid | 彗星　すい星 |
| Hiragana vs. katakana | ぴかぴか　ピカピカ |

Table 7: Cross-script variants

### 4.4 Kana Variants

Recent decades have seen a sharp increase in the use of katakana, a syllabary used mostly to write loanwords. A major annoyance in Japanese information processing is that katakana orthography is often irregular; it is quite common for the same word to be written in multiple, unpredictable ways which cannot be generated algorithmically. Hiragana is used mostly to write grammatical elements and some native Japanese words. Some of the major types of kana variation are shown in Table 8.

| **Type** | **English** | **Reading** | **Standard** | **Variants** |
|---|---|---|---|---|
| Macron | computer | *konpyuuta* *konpyuutaa* | コンピュータ | コンピューター |
| Long vowels | maid | *meedo* | メード | メイド |
| Multiple kana | team | *chiimu, tiimu* | チーム | ティーム |
| Traditional | big | *ookii* | おおきい | おうきい |
| づ vs. ず | continue | *tsuzuku* | つづく | つずく |

Table 8**:** Katakana and hiragana variants

Other types of Japanese orthographic variants of less importance are described in (Halpern, 2006).

### 4.5 Lexicon-driven Normalization

Lexicon-driven normalization of Japanese orthographic variants can be achieved by orthographic mapping tables such as the one shown below, using various techniques such as:

1. Convert variants to a standardized form for indexing.
2. Normalize queries for dictionary lookup.
3. Normalize all source documents.
4. Identify forms as members of a variant group.

Table 9 shows the variants for 空き缶 /akikan/ 'empty can' mapped to a normalized form for use in indexing and dictionary lookup. Such tables are used by portals like Yahoo and Amazon Japan to ensure maximum recall in processing queries and for improving word segmentation accuracy.

| Headword | Reading | Normalized |
|---|---|---|
| 空き缶 | あきかん | 空き缶 |
| 空缶 | あきかん | 空き缶 |
| 明き罐 | あきかん | 空き缶 |
| あき缶 | あきかん | 空き缶 |
| あき罐 | あきかん | 空き缶 |
| 空きかん | あきかん | 空き缶 |
| 空きカン | あきかん | 空き缶 |
| 空き罐 | あきかん | 空き缶 |
| 空罐 | あきかん | 空き缶 |
| 空き鑵 | あきかん | 空き缶 |
| 空鑵 | あきかん | 空き缶 |

Table 9: Orthographic normalization table

Using statistical or algorithmic methods to achieve such normalization will produce poor or no results as it is not possible to identify such character sequences as 空きカン and あき缶, which don't share a single character, as being variants of each other. Other possibilities for normalization include advanced applications such as domain-specific synonym expansion, requiring Japanese thesauri based on domain ontologies, as is done by a select number of companies like Wand and Convera who build sophisticated Japanese IR systems.

## 5. Korean Orthographic Variants

Korean has a significant amount of orthographic variation. Combined with the morphological complexity of the language, this poses various challenges to developers of NLP applications. The issues are similar to Japanese in principle but differ in detail and scale. The details of Korean orthographic variation, described in (Halpern, 2006), are beyond the scope of this paper.

Briefly, Korean has variant hangul spellings in the writing of loanwords, such as 케이크 *keikeu* and 케잌 *keik* for 'cake', and in the writing of non-Korean personal names, such as 클린턴 *keulrinteon* and 클린톤 *keulrinton* for 'Clinton'. In addition, Korean is written in multiple scripts: hangul, Chinese characters (whose use has decreased) and the Latin alphabet. For example, 'shirt' can be written 와이셔츠 *wai-syeacheu* or Y 셔츠 *wai-syeacheu*, whereas 'one o'clock' hanzi can be written as 한시, 1 시 or 一時. Another issue is the difference between South and North Korean spellings, such as N.K. 오사까 *osakka* vs. S.K. 오사 카 *osaka* for 'Osaka', and the old (pre-1988) orthography versus the new, i.e. modern 일군'worker' (*ilgun*) used to be written 일꾼(*ilkkun*).

Lexical databases, such as normalization tables similar to the ones shown above for Japanese, are the only practical solution to identifying such variants, as they are in principle unpredictable.

## 6. Orthographic Ambiguity in Arabic

### 6.1 Why is Arabic ambiguous?

A distinguishing feature of the Arabic script is that words are written as a string of consonants with little or no indication of vowels, referred to as **unvocalized Arabic.** Though diacritics can used to indicate short vowels, they are used sparingly, while the use of consonants to indicate long vowels is ambiguous. On the whole, unvocalized Arabic is highly ambiguous and poses major challenges to Arabic information processing (Halpern, 2007).

### 6.2 Morphological Ambiguity

Arabic is a highly inflected language. Inflection is indicated by changing the vowel patterns as well as by adding various suffixes, prefixes, and clitics. A full paradigm for اآ تـب /kaatib/ 'writer' that we created (for an Arabic-English dictionary project) reaches a staggering total of 3487 valid forms, including affixes and clitics as well as inflectional syncretisms. Even without affixes, كاتـب can represent any of the following seven word forms: كَاتِـــب /kaatib/, كَاتَـــبَ /kaataba/, كَاتِـــبٍ /kaatibin/, كَاتِـــبٌ /kaatibun/, كَاتِـــبَ /kaatiba/, كَاتِـــبِ /kaatibi/, كَاتِـــبُ /kaatibu/.

### 6.3 Orthographical Ambiguity

On the orthographic level, Arabic is also highly ambiguous. For example, the string مو can theoretically represent 40 consonant-vowel permutations, such as *mawa, mawwa, mawi, mawwi....* etc., though in practice some may never be used. Humans can normally disambiguate this by context, but for a program the task is formidable. Various factors contribute to orthographical ambiguity, of which the most important ones are briefly described below.

1. The most important factor is the omission of short vowels; e.g., the unvocalized كاتـب can represent seven wordforms such as كَاتِـــب /kaatib/ and كَاتِـــبَ /kaatiba/. In contrast, some short vowels actually are represented. For example, *taa' marbuuTa* often indicates a short /a/, as in جامعة /jaami`a/, while in foreign names short and long vowels are normally written identically by adding ا ,ي or و, as in روســـيا /ruusiyaa/ 'Russia'.
2. Long /aa/ can be expressed in multiple ways, e.g., by *'alif Tawiila* (ا) as in ســـوريا, by (2) *'alif mamduuda* (آ) as in آســـيا, and by (3) *'alif maqSuura* (ى) as in آســـيا الوســـطى, but sometimes they are omitted, as as in هذا/haadha/.
3. Not all bare alifs represent long /a/. Some are nunated; e.g., را in شـــكرا represents /ran/, رأ , *not* رَا /raa/, *'alif alfaaSila* (otiose alif), added to the third person masculine plural forms of the past tense, is a mere orthographic convention and is not pronounced.

4. The diacritic *shadda* indicating consonant gemination is normally omitted, e.g., the un-vocalized محمد *Muhammad* (vocalized مُحَمَّـــد) provides no clues that the [m] should be doubled.
5. *Tanwiin* diacritics for case endings are normally omitted, e.g., in شـكرا /shukran/ (vocalized شُـــكْراً), the *fatHatayn* is not written.
6. The rules for determining the hamza seat are of notorious complexity. In transcribing to Arabic, it is difficult to determine the hamza seat as well as the short vowel that follows; e.g., hamzated *waaw* (ؤ) could represent /'a/, /'u/ or even /'/ (no vowel).
7. Phonological alternation processes such as assimilation that modify the phonetic realization. For example, الرجل الطويل 'the tall man' is realized as /'arrajulu-TTawiilu/, in which the ال is assimilated into ط /TTa/, not as /'alrajulu alTawiilu/.

### 6.4 Vowel Sequence Ambiguity

A special kind of ambiguity arises when transcribing into Arabic foreign names that contain vowel sequences. Such sequences are difficult to transcribe because they could represent diphthongs, monophthongs, or long vowels. In the analysis below Japanese place names are used in the examples. Though the examples are from Japanese, the principles apply to many other languages as well.

| No. | Arabic | Google hits | Transliteration |
|---|---|---|---|
| 1 | فوكوئـــي | 468 | fwkw}y |
| 2 | فوكـــوئ | 9 | fwkw} |
| 3 | فوكـــوي | 1950 | Fwkwy |
| 4 | فوكويـــي | 335 | Fwkwyy |

Table 10. Dipthong ambiguity for 福井 /fu-ku-i/

Table 10 shows some of the variation to expect in transcribing Japanese names into Arabic. As can be seen, when vowel sequences represent monophthongs, hamza is sometimes used and sometimes omitted. Though phonologically (2) is the most accurate, it is the least used. As expected, the diphthongized (3) is the most common form because of the tendency to avoid hamza in foreign names. Some important vowel sequence issues are:

1. There is a strong tendency not to use non-initial hamza, as in (1) and (2) above, in foreign names. One reason for this is insufficient knowledge of the phonology of the source language.
2. Japanese is especially problematic because it is moraic. Some Japanese mora sequences, such as あい /ai/ or うい /ui/, are often diphthongized in Arabic, though ideally the second vowel should be treated as a monophthong represented by hamza. That is, 福井/fu-ku-i/ should be written as (1) فوكوئـــي or (2) فوكـــوئ, rather than the more common (3) فوكـــوي.
3. In theory, a vowel sequence like /ai/ as in さい /sa-i/ can be written in five ways: ساي سي سائ سـائي سـايي. To accurately transcribe a name like Saitama (埼玉) it is necessary to know that it consists of four morae (/sa-i-ta-ma/ さいたま), rather than three syllables (/sai-ta-ma/). Ideally it should be transcribed as سـائيتاما, rather than the more common سـايتاما. That is, since /sa-i/ is a bimoraic syllable, the hamza over *yaa'* should be used to represent /i/ as a distinct monophthong, as in سائ. In reality, Saitama is normally spelled سـايتاما, so that /sa-i/ is diphthongized as ساي /say/.
4. In names like 福岡 /fu-ku-o-ka/ the sequence /ku-o/ represents distinct sounds that cannot be diphthongized. Following hamza rules, this should be written فوكوؤوكـا, but in fact it is commonly spelled فوكوأوكـا, in which أو, rather than ؤو, represents /u/.

### 6.5 Arabic Orthographic Variants

Both Arab and foreign names have orthographic variants in Arabic. These are of two kinds:

1. Orthographic variants are nonstandard ways to spell a specific variant of a name, like ابـو ظـبي instead of أبـو ظـبي for Abu Dhabi, in which the hamza is omitted.
2. Orthographic errors are frequently occurring, systematic spelling mistakes, like *yaa'* in ابـو ظـبي (Abu Dhabi) being replaced by *'alif maqSuura* in ابـو ظـبى.

| Standard | Transliteration | English | Variant | Error | Remarks |
|---|---|---|---|---|---|
| أبـو ظـبي | >bw Zby | Abu Dhabi | ابـو ظـبي | أبـو ظـبى<br>ابـو ظـبى | V: omit hamza<br>E: *'alif maqsura* replaces *yaa'* |
| الإســـكندرية | Al<skndryp | Alexandria | الاســـكندرية | الإســـكندريه | V: omit *hamza*<br>E: *haa'* replaces *taa' marbuuTa* |
| بـالو وألت | bAlw >ltw | Palo Alto | بـالو التـو<br>بـالو آلتـو | | V1: omit hamza<br>V2: *madda* replaces hamza |
| طوكيـو | Twkyw | Tokyo | | توكيـــو | E: *taa'* replaces *Taa'* |

Table 11: Orthographic variation in Arabic names

Table 11 shows examples of variants ("V") and errors ("E"). Though the difference between these cannot be rigorously defined, they are both of frequent occurrence based on statistical and linguistic analysis of MSA orthography. It should also be noted that the "standard form," though linguistically correct, is not necessarily the most common form (we are gathering statistics for the occurrence of each form). There are often many more variants than those shown above. For example, *Alexandria* can be written in about a dozen ways, the most frequent ones according to Google being الاســـــكندرية with الإســـــكندرية , 2,930,000 with 690,000, and الاســـــكندريه with 89,200 occurrences respectively.

## 7. The Role of Lexical Databases

Because of the orthographic irregularity and ambiguity of CJK languages and Arabic, procedures such as orthographic normalization cannot be based on probabilistic methods like bigramming and algorithmic methods alone. Many attempts have been made along these lines (Goto et al., 2001; Brill et al. 2001), with some claiming performance equivalent to lexicon-driven methods, while others report good results with only a small lexicon and simple segmentor (Kwok, 1997).

It has been reported that a robust morphological analyzer capable of processing lexemes, rather than bigrams or *n*-grams, must be supported by a large-scale computational lexicon (Emerson, 2000) in what is often referred to as the hybrid approach. This experience is shared by many of the world's major portals and MT developers, who make extensive use of lexical databases. Unlike in the past, disk storage is no longer a major issue. Many researchers and developers, such as Prof. Franz Guenthner of the University of Munich, have come to realize that "language is in the data," and "the data is in the dictionary," even to the point of compiling *full-form* dictionaries with millions of entries rather than relying on statistical methods. For example, Meaningful Machines uses a full form dictionary developed by our institute containing over ten million entries used in a human-quality Spanish-to-English context-based MT system, as reported by Carbonell (2006).

In line with our policy that lexical resources should play a central role in NLP applications, our institute is engaged in research and development to compile CJK and Arabic lexical databases (currently about nine million entries), with special emphasis on proper nouns, orthographic normalization, and technical terminology. These resources are being subjected to heavy use in real world applications, and the feedback thereof is used to expand these databases and fine tune them.

## 8. Conclusion

Because of the irregular orthography of the CJK and Arabic writing systems, NLP applications require not only sophisticated tools such as morphological analyzers, but also lexical databases to enable orthographic disambiguation. Achieving accurate orthographic normalization for information retrieval and named entity extraction, not to speak of C2C conversion and morphological analysis, is beyond the ability of statistical methods alone. Large-scale lexical databases fine-tuned to the needs of specific NLP applications should play a central role. The building of such resources consisting of even billions of entries has come of age. Since lexicon-driven techniques have proven their effectiveness, there is no need to overly rely on probabilistic methods. Comprehensive, up-to-date lexical resources are the key to achieving high accuracy in disambiguating and processing orthographic variants.

## References


Brill, E., Kacmarick, G., Brocket, C. (2001). Automatically Harvesting Katakana-English Term Pairs from Search Engine Query Logs. In *Proceedings of the Sixth Natural Language Processing Pacific Rim Symposium*, Tokyo, Japan.

Carbonell, J. et al (2006). Context-Based Machine Translation. In Proceedings of the 7th Conference of the Association of Machine Translation in the Americas, Cambridge, MA.

Emerson, T. (2000). Segmenting Chinese in Unicode. In *Proceedings of the 16th International Unicode Conference*, Amsterdam.

Goto, I., Uratani, N., Ehara T. (2001). Cross-Language Information Retrieval of Proper Nouns using Context Information. In *Proceedings of the Sixth Natural Language Processing Pacific Rim Symposium*, Tokyo, Japan.

Halpern, J., Kerman J. (1999). The Pitfalls and Complexities of Chinese to Chinese Conversion. In *Proceedings of the Fourteenth International Unicode Conference*, Cambridge, MA.

Halpern, J. (2003). The Challenges of Intelligent Japanese Searching, working paper (www.cjk.org/cjk/joa/joapaper.htm), The CJK Dictionary Institute, Saitama, Japan.

Halpern, J. (2006). The Role of Lexical Resources in CJK Natural Language Processing. In *Proceedings of COLING/ACL 2006*, Sydney.

Halpern, J. (2007). The Challenges and Pitfalls of Arabic Romanization and Arabization. In *Proceedings of Computational Approaches to Arabic Script-based Languages*, Palo Alto, CA.

Kay, M. (2004). Arabic Script based Languages deserve to be studied linguistically. In COLING 2004, Geneva.

Kwok, K.L. (1997). Lexicon Effects on Chinese Information Retrieval. In *Proceedings of 2nd Conference on Empirical Methods in NLP*, ACL 141-148.

Lunde, K. (1999). CJKV Information Processing. O'Reilly & Associates. Sebastopol, CA.

Tsou, B.K., Tsoi, W.F., Lai, T.B.Y., Hu, J., Chan S.W.K. (2000). LIVAC, a Chinese synchronous corpus, and some applications. In *2000 International Conference on Chinese Language Computing (ICCLC2000)*, Chicago, IL.

Yu, Shiwen, Zhu, Xue-feng, Wang, Hui (2000). New Progress of the Grammatical Knowledgebase of Contemporary Chinese. *Journal of Chinese Information Processing*, 15(1).