Pinyomi: Dictionary lookup via orthographic associations

Lars Yencken
NICTA Research Lab
University of Melbourne
`lljy@csse.unimelb.edu.au`

Zhihui Jin and **Kumiko Tanaka-Ishii**
Graduate School of
Information Science and Technology
University of Tokyo
`{jin,kumiko}@i.u-tokyo.ac.jp`

Abstract

Bilingual dictionaries provide meaning associations between the words of two languages, those of an ideal bilingual speaker. Learners can use these associations to look up foreign equivalents of known native words, but are forced to use script-based lookup methods when faced with unknown foreign words. This paper presents the Pinyomi Chinese-Japanese dictionary interface, which uses a novel method of using associations with the learner's native script to look up foreign language words.

1 Introduction

Bilingual dictionaries can be considered as bipartite graphs, where the nodes are the words of the two languages, and the links between them correspondences in meaning, as shown in Figure 1. Example (a) shows that the general relationship is many-to-many, since words may have many senses, and the distribution of these senses differs from language to language. In this case the French *adresse* shares senses of the English *address* (a description of location; a speech), but can also correspond to *skill* or *dexterity*. This example also highlights that associations of sound and of orthography also occur for a word pair when either features are similar enough.

Where enough similarity occurs, learners can frequently guess part or all of a foreign word's meaning based on its associations to native words. This is a form of beneficial *language transfer* from their native language to their second language. When sound or orthography associations do not coincide with meaning associations, these misleading pairs are called *false friends*. In example (c), the Chinese 手纸 could be naively guessed to mean "letter" by a Japanese speaker, instead of its real meaning "toilet paper".

Regardless of their semantic accuracy, associations can still be useful for lookup, particularly in cases where the lookup target is difficult for the user to input directly, as is the case in Japanese and Chinese[1]. Typically users must painstakingly look up each character in the foreign word by stroke count and radicals before they can look up the word as a whole. Instead, they may query via associated words or characters in their native language, which are easy for them to input.

In this paper we present the Pinyomi system, a dictionary interface for Chinese learners studying Japanese, and Japanese learners studying Chinese, which allows these learners to quickly lookup an unknown word using pronunciation from their own native script. It takes advantage of the script similarities between these languages, allowing users to lookup a foreign word by the orthographic associations that word provides. Since such associations are often noisy, Pinyomi uses a statistical model to recover candidates for the desired foreign word. This form of lookup is significantly more convenient than naive script-based methods.

For example, suppose a Japanese speaker was trying to look up the Chinese word 趣闻. The first character exists in Japanese, and the second is similar to the Japanese 聞. Together these might be read [shumoN] in Japanese. Searching Pinyomi with this reading yields the desired word 趣闻 [qùwén] "interesting news" and its Japanese translation.

[1] In some cases, this can be circumvented on computer by cutting and pasting the unknown word into an electronic dictionary.

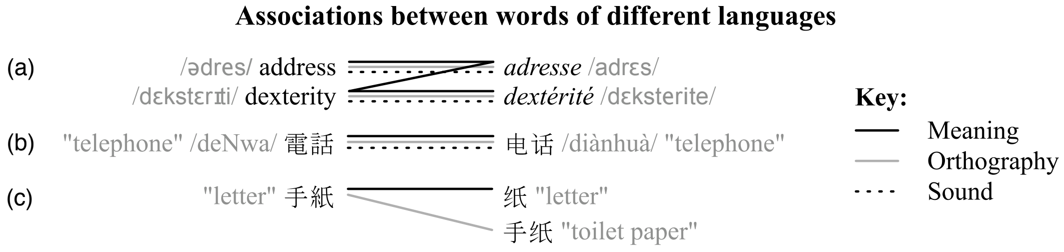


Figure 1: Bilingual dictionaries as bipartite graphs, and different forms of word associations which can occur. Example (a) is English-French, and examples (b) and (c) are Japanese-Chinese.

The remainder of this paper is structured as follows. We firstly introduce related work in transliteration and dictionary lookup (Section 2), before discussing how associations form between the Japanese and Chinese languages (Section 3). We provide an overview of the system and its architecture (Section 4), then delve into the lookup model itself (Section 5). Finally we evaluate our system (Section 6) and discuss its performance (Section 7).

2 Related research

In order to search for a foreign word, Pinyomi requires the user to convert the foreign characters into native characters using any associations they may have upon seeing the foreign characters. The general task of converting a word from one script to another is called *transliteration*. The task of recovering the original word given the converted word is called *back-transliteration*, and is considerably more difficult[2]. It is this latter task which Pinyomi performs to recover a foreign word given a transliteration into the user's native script.

There is much interesting work on these tasks between various language pairs, including English-Japanese (Knight and Graehl, 1998; Brill et al., 2001; Qu et al., 2003), English-Chinese (Li et al., 2004), and many others. The conversion between scripts which Pinyomi users perform has some similarity to transliteration, but is chiefly different its conversion between *logographic* scripts of shared origin.

Since both Japanese and Chinese have a vast wealth of distinct symbols to draw upon in writing, less information is lost in transliterating between these languages. The problem is then closer to Chinese-to-Chinese conversion in difficulty (Halpern and Kerman, 1999). The lack of research in transliterating between this pair also indicates that importing of names and technical terms is less productive compared with other language combinations.

Pinyomi also shares features of high-accessibility dictionaries which allow queries based on noisy or partial user input, a good example being the FOKS dictionary for Japanese (Bilac et al., 2002). FOKS allows a user to guess the compositional reading of a set of characters based on any of their individual readings, and can correct for many common deviations from the correct pronunciation. Pinyomi also complements recent use of associations within monolingual dictionary lookup, aiding writers with the tip-of-the-tongue problem (Ferret and Zock, 2006), by extending this work to more general associations and a bilingual context.

3 Associating Han characters

Pinyomi aims to make use of the natural associations learners form when seeing an unknown word. The type and number of associations that form necessarily depend on the language pair targeted. For Japanese and Chinese, we now consider their writing systems in more detail.

Japanese and Chinese both use a logographic script, called *kanji* in Japanese and *hanzi* for Chinese (where we refer to simplified forms). Kanji in general consist of traditional Chinese characters, with a smaller number of native Japanese additions and simplifications. Kanji and hanzi are thus related historically as branches of an earlier traditional Chinese script. For this reason, we call such characters *Han characters*, and words consisting of such characters *Han words*. For kanji, pronunciation can be expressed in a phonetic *kana* script,

[2] The different phonetic and symbolic inventories of any two languages make transliteration lossy in general. This creates ambiguity for back-transliteration.

and for hanzi the *pinyin* romanisation is typically used (see Figure 2).

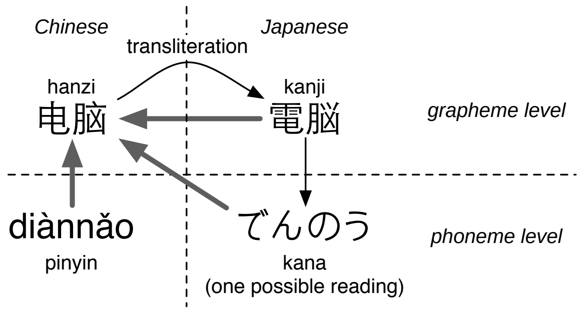


Figure 2: An example of Chinese-Japanese lookup using Pinyomi for the Chinese word meaning "computer". Grey arrows represent lookup methods.

There are two significant types of character-level associations which learners may have when seeing a foreign Han character, shape-based and semantic, which we now discuss with the aid of Table 1. Shape-based associations are made when the shape of a foreign Han character is similar to one from the learner's native language. The simplest example of this is characters shared by both scripts. Other pairs have such similar shapes that they are easily recognisable to natives of one script learning the other. A larger set become associated if some script simplification rules for Chinese are understood. For example, once a learner discovers that 东 is a simplification of 東, it will be very easy for him to associate 冻 with its cognate 凍, and many more follow. These correspond to the "Shape-based" category in Table 1.

Type	**Chinese**	**Japanese**
Chinese-only	你 乒	
Shape-based	主 冻	主 凍
Semantic	卖 站	売 駅
Ambiguous	发 馆	発 發 舘 館
Japanese-only		込 峠

Table 1: Examples of Han characters, with and without equivalence.

Pairs similar in shape are usually so because they are *cognates*, modern variants of what was once a single character. Furthermore, since character meanings have been quite stable over time in both languages, the meaning of cognate pairs is usually strongly related. Sub-components of characters also influence whole-character semantics, so a strong similarity of shape may correspond to a similarity of meaning even amongst non-cognates.

On the other hand, there are pairs whose shape may differ significantly, but whose meaning may be very similar. For example, 站 [zhàn] and 駅 [eki] both have the main meaning "station", although 站 can be used for a bus stop, whereas 駅 cannot. This semantic association can take place even though the pair are not cognates[3]. Some cognates may also be difficult to determine due to shape differences. However, given a character in context, the surrounding characters can provide the strong cues needed for such an association to form. These correspond to the "Semantic" category in Table 1.

When potential associations exist with several native characters, the learner may transliterate in several possible ways, forming the "Ambiguous" category in Table 1. This often occurs when more than one native cognate exists, possible in the Chinese-Japanese direction since hanzi have been more simplified than kanji, merging many historically distinct characters together. It also occurs where the meaning in one or both languages has drifted substantially, making room for non-cognate semantic equivalents.

There also exist characters which learners will not associate strongly with any in their native script. This may be particularly true of characters created in the two countries since the time the scripts diverged, such as 你 [nǐ] "you" in Chinese and 込 [ko] "crowded" in Japanese. In Table 1 these cases are marked as "Chinese-only" or "Japanese-only". This effect also occurs when a cognate does exists, but it is much rarer in the other language.

Once the learner has mentally produced such a transliteration, how might they pronounce it? This problem is small in the Japanese-Chinese direction, since most hanzi have a unique pronunciation, but is significant for kanji, which feature many individual readings, and even non-compositional readings in compounds. Figure 3 depicts this ambiguity for individual characters,

[3] 站 exists but is rare in Japanese. 駅 is cognate with 驿 in Chinese via the traditional Chinese character 驛 [yì] "remount station".

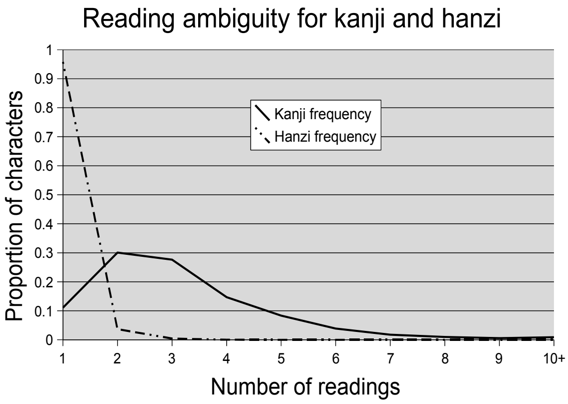


Figure 3: The distribution of reading ambiguity for Chinese and Japanese characters. Hanzi are taken from the GB 2312 set, kanji from JIS X 0208-1990 and JIS X 0212-1990 sets.

where the horizontal axis indicates the number of distinct readings, and the vertical axis indicates the proportion of characters that have this number of readings. For non-word character sequences or rare words, any combination of possible readings could be correct, and should be considered valid inputs.

4 Overall system

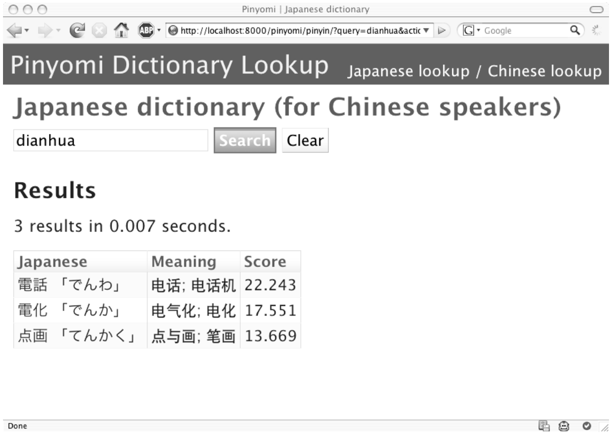
In this section, we provide an overview of Pinyomi, firstly from a user's perspective, and secondly from an engineering perspective.

4.1 User interface

Pinyomi presents two user interfaces: a Chinese dictionary supporting lookup by kana, and a Japanese dictionary supporting lookup by pinyin. Figure 4 below indicates the simple layout of results from a Japanese-Chinese query, with results scored by probability.

In the reverse direction, to look up the Chinese 电话, a Japanese person can follow shape similarity or simplification rules to determine correspondence with the Japanese 電話. They thus enter its reading, “でんわ” [deNwa] into the input box, and click “Search”. The results displayed contain the desired Chinese word and its translation.

Despite a Japanese learner's natural ability to transliterate from Chinese to Japanese, there will be some characters for which no association is strong. In these cases, the user replaces that character's reading with a special wild-card character “?”, and can still find the target word.


Pinyomi | Japanese dictionary
http://localhost:8000/pinyomi/pinyin/?query=dianhua&actic
Google
Pinyomi Dictionary Lookup
Japanese lookup / Chinese lookup
Search
Clear
Done

Japanese dictionary (for Chinese speakers)

dianhua

Results

3 results in 0.007 seconds.

Japanese	Meaning	Score
電話 「でんわ」	电话; 电话机	22.243
電化 「でんか」	电气化; 电化	17.551
点画 「てんかく」	点与画; 笔画	13.669

Figure 4: Pinyomi's Japanese-Chinese dictionary interface, where the user is searching for 電化.

4.2 Architecture

Details of candidate generation and scoring are provided in the following section, but an overview of Pinyomi's construction is as follows. For each foreign Han word in the dictionary, we:

1. Determine the word's corpus frequency.
2. Exhaustively determine and store transliteration candidates for the word.
3. For each transliteration candidate, exhaustively determine and store potential readings for this candidate.

This procedure constructs the database, which is queried at run-time by the web interface. The resources used for the previous steps are:

- A Chinese-Japanese and Japanese-Chinese bilingual dictionary.
- A Japanese corpus: we used 220 MB from the Nikkei corpus, taken from the Nikkei financial newspaper.
- A Chinese corpus: we used 200 MB from the Peking University corpus for Chinese, consisting of a wide selection of popular print media (Center for Chinese Linguistics, 2007).
- A reading dictionary for Japanese: the Kanjidic dictionary[4].

[4] http://www.csse.monash.edu.au/~jwb/kanjidic.html

- A GBK pinyin reading table for Chinese.
- A table of hanzi and kanji equivalence candidates (Goh et al., 2005).

We now describe the scoring model in detail.

5 Candidate generation and scoring

5.1 General model

The general model is provided in Equation 1 below, where s is the word a user would like to look up, t is its transliteration into their native language, and r is the reading of that transliteration in their native phoneme representation. The model determines which candidates s to display to the user if they enter r as their query, and orders them by $\Pr(s|r)$.

$$\begin{aligned} s &: \text{word in foreign language} \\ \phi(s) &: \text{all transliterations candidates of s} \\ t &: \text{a transliteration candidate} \\ r &: \text{reading in native language} \end{aligned}$$

$$\begin{aligned} \Pr(s|r) &\propto \Pr(r|s)\Pr(s) \\ &= \Pr(s) \sum_{t \in \phi(s)} \Pr(r,t|s) \\ &= \Pr(s) \sum_{t \in \phi(s)} \Pr(r|t,s)\Pr(t|s) \\ &= \Pr(s) \sum_{t \in \phi(s)} \Pr(r|t)\Pr(t|s) \qquad (1) \end{aligned}$$

In the first line, $\Pr(r)$ is not important for ordering results, so it is omitted. After the user transliterates the foreign word s to native pseudo-word t, the reading r chosen by the user is only dependent on t, so we can simplify $Pr(r|t,s) = Pr(r|t)$.

The final line in Equation 1 requires three submodels to be defined, namely:

- A language model to calculate $\Pr(s)$.
- A transliteration model to calculate $\Pr(t|s)$.
- A reading model to calculate $\Pr(r|t)$.

We now discuss these additional models.

5.2 Language model – $\Pr(s)$

$Pr(s)$ is the probability that the user will look this foreign word up, relative to other words in the dictionary. This may not coincide fully with corpus frequency, since some common words are unlikely to be looked up if their meaning is obvious to these learners. Nonetheless, corpus frequency is a good approximation which makes no assumptions about the user's proficiency level.

5.3 Transliteration model – $\Pr(t|s)$

We aim to model how users will perform their mental transliteration, using the associations mentioned in Section 3. In fact there are two related problems this model must solve: what candidates to use, and how to score them.

Ideally, we would use a parallel corpus between Chinese and Japanese to automatically extract transliterations, and then use this data to solve both of these problems simultaneously. By assuming the process is Markovian, we could train an n-gram model to perform context-sensitive transliteration. However, in the absence of such a corpus, we are limited to simpler approximations.

To determine the candidate set, we use a mapping table provided by Goh et al. (2005), which was constructed by mostly using cognate pairs, but also shape-similarity, and semantic correspondence – the same associations we discussed earlier. This mapping table itself is incomplete: it is one-to-many from hanzi to kanji, although general associations should be many-to-many. This makes the model ambiguous when used to transliterate from hanzi to kanji, but not in the reverse direction. Nonetheless, it is useful as a starting point for future work.

Given foreign Han word $s = s_1 \ldots s_n$, we model the user's transliteration into the native pseudo-word $t = t_1 \ldots t_n$ on a character by character basis.

$$\begin{aligned} \Pr(t|s) &= \Pr(t_1 \ldots t_n | s_1 \ldots s_n) \\ &= \prod_{i=1}^{n} \Pr(t_i | s_1 \ldots s_n) \\ &= \prod_{i=1}^{n} \Pr(t_i|s_i), \quad t_i \in \phi(s_i) \qquad (2) \end{aligned}$$

For each foreign Han character, there may be several associated native characters which they might use. Here we assume that the t_i is only dependent on s_i, so we reduce $\Pr(t_i|s_1 \ldots s_n)$ to $\Pr(t_i|s_i)$. For each foreign Han character s_i, the set of corresponding native Han character candidates $\phi(s_i)$ can be found in the mapping table. The mapping probability is estimated by using relative corpus frequency:

$$\Pr(t_i|s_i) = \frac{\#t_i}{\sum\limits_{t_j \in \phi(s_i)} \#t_j}$$

Since the mapping table only contains one-to-many mapping from hanzi to kanji, when we convert a kanji s_i to hanzi t_i, we only have one choice, so in this case $\Pr(t_i|s_i) = 1$.

When a foreign characters s_i yields no associations, the user can instead use a wild-card character "?". Thus any potentially unreadable character should have "?" as a transliteration candidate. This occurs when: a) all potential associations are with rare, unknown characters, or b) the shape is very different, making the association difficult to form. To estimate the native characters that each user would know, we used high-school-level character sets for each country, as determined by government education standards.

Suppose that k characters in total can be potentially replaced by "?". Then when the user maps a foreign character s_i to "?", we simply calculate the mapping probability using a uniform distribution $\Pr(?|s_i) = \frac{1}{k}$.

5.4 Reading model – $\Pr(r|t)$

Given the pseudo-word t in native language, the user must choose a reading for this word, as discussed in Section 3. Here we use a simple unigram model, which suffices to distinguish candidates in both lookup directions. We leave the possible impact of higher order models as an open question for future research.

$$\begin{aligned} \psi(t) &: \text{ all readings of t} \\ \Pr(r|t) &= \Pr(r_1 \ldots r_n | t_1 \ldots t_n) \\ &= \prod_{i=1}^{n} \Pr(r_i | t_1 \ldots t_n) \\ &= \prod_{i=1}^{n} \Pr(r_i|t_i), \quad r_i \in \psi(t_i) \qquad (3) \end{aligned}$$

A crucial feature of the Pinyomi kana interface is that it requires kana readings to be input by the user in character-by-character, space-separated manner. This rules out the use of compositional readings, even where the t sequence forms a natural word in Japanese. Adding such a constraint on the user is acceptable, since the users are native speakers, and are highly likely to know many readings for a given kanji.

	No wild-cards		**Using ?**		
$\lvert w \rvert$	Best	Random	Best	Random	%
1	9.41	12.40	56.31	58.50	25.7
2	1.63	2.02	4.48	6.04	49.3
3	1.01	1.01	1.13	1.19	12.2
≥4	1.00	1.00	1.00	1.01	12.7
all	3.47	4.43	16.94	18.28	100

Table 2: Mean rank by word length, using either the most likely kana query, or a random kana query for each Chinese word in the dictionary.

Japanese is non-segmenting, and each kanji can have a readings of various length. For this reason, determining corpus frequency for our unigram model was slightly more complicated than in the pinyin direction. We took frequency counts for kanji readings from the 1990 Nikkei corpus, which was segmented and annotated with readings by Chasen, and finally grapheme-phoneme aligned using a simple TF-IDF based alignment model (Yencken and Baldwin, 2005). Counts for compositional readings were discarded, readings outside those provided by Kanjidic were filtered as noise or obscure name readings, and any remaining readings had their frequency counted. These frequencies were used to train the unigram model above.

6 Evaluation and Analysis

We evaluate Pinyomi in two primary ways. The first is some basic analysis of the reading database itself, and the second is through a user experiment, where we asked users to give native readings for foreign Han words.

6.1 Reading set analysis

For each word in the Chinese dictionary, we selected from our model of Japanese readings the most probable reading, and queried the database to determine the rank of the original word. We then repeated, but with a random reading from our model. Averaging this rank over all words in the dictionary gives the results in Table 2. In this table, the "No wild-cards" category used readings without "?", whereas the "Using ?" category replaced a query reading with "?" whenever possible. This analysis was only performed for Chinese lookup by kana, since this direction is more ambiguous and thus more difficult.

Notably, the mean rank is quite large for single character words, of which there are many more in Chinese than Japanese. However, in both cases the mean rank was quite small for words of length 2 or greater, as queries became more constraining. The "Best" reading was intended to capture an ideal case, and the "Random" reading a normal case, however the difference in ranking between the two is small. This indicates that transliterating appropriately is more important than choosing a good reading candidate, since any reading candidate is sufficiently constraining to give a decent ranking.

The very large ranking given to one-character words looked up by wild-card indicate that using "?" alone is simply not constraining enough. Without using wild-cards, the ranking is still not ideal, due to the large number of homophone kanji in Japanese. This lookup method thus is no replacement for looking up individual characters, but instead has large advantages for words which are compounds. For lookup of Japanese words, the frequency of kanji compounds which are one-character is instead at 3%, reducing the problem.

6.2 Reading experiment

Although simulated queries give some information, we are interested in the real-world performance of the system. For this reason, we conducted two symmetric user experiments where we firstly explained the system, then asked users to provide search queries in kana (for Japanese participants) and pinyin (for Chinese participants) for foreign Han words.

For stimulus, 100 random Han words of length two or greater were taken from each of the two bilingual dictionaries, of which 30 were randomly presented to each user. Each user entered between one and three readings in their native language, and told to use "?" if they couldn't read a particular foreign character. Reading input boxes were segmented so that users gave readings on a per-character basis.

In total, 28 Chinese native speakers and 12 Japanese native speakers completed the experiment, most of whom had never studied each-other's language before. These users covered 98 stimulus for kana, and all 100 for pinyin[5].

To determine how well Pinyomi performs on these queries, we performed each query on the Pinyomi system. Each word successfully found has a rank, and by averaging the rank over these words we get a mean rank statistic. We calculated this statistic using just the first reading the user gave, and also using all three. In the latter case, we only used an additional reading when lookup failed, simulating repeated attempts to find the same word. The percentage of queries where the query located the desired word is indicated by the coverage, as given in Table 3.

Method	r	c	n	r/n
Kana	1.99	0.64	10.01	0.20
Pinyin	1.30	0.83	2.54	0.51

Table 3: Rank and coverage for the readings given by experimenters. r is the mean rank of a successful queries, c is the coverage, and n is the mean number of candidates returned.

The mean rank statistic for both lookup methods is very good, indicating that when the query is successful, the desired word is likely to be in the top few results. To evaluate the ranking independently of the number of candidates returned, we consider the relative ranking r/n. For kana lookup, the desired word is typically ranked well in the top 20% of results, compared to pinyin lookup, where the word is in the top 51% of results. The low number of pinyin candidates prevents this from being problematic, but if future enhancements increase the number of candidates, care may be required to maintain a high mean rank.

Of primary concern is the low coverage for kana queries. In order to improve coverage, we need to determine what the cause of the failed queries is.

6.3 Error Analysis

In order to better understand the gaps in coverage, we asked a native Japanese speaker and a native Chinese speaker to analyse the failed kana and pinyin searches respectively, grouping into common error types. The results of these analyses are provided in Table 4.

The following error categories are used. "Reading ambiguity" refers to typical alternations used when native speakers articulate the pronunciation of a word or psuedo-word. Chinese speakers have trouble distinguishing between two close pinyin sound representations, for example zhàn and zàn. Japanese has reading alternation effects which occur when characters are combined in a compound.

[5] Due to the method of random allocation of stimulus to each user, coverage of all stimulus words was not guaranteed. Two words in the kana direction were not covered by any user responses for this reason.

Error Type	Pinyin	Kana
Reading ambiguity	37.6%	14.5%
Radical or shape	22.0%	43.5%
Unknown character	10.8%	25.8%
Miscellaneous	29.6%	16.1%
Total	100.0%	100.0%

Table 4: Error analysis for failed pinyin and kana searches.

For the Chinese word 森林, a Japanese user entered “もりばやし” [mori bayashi] instead of “もりはやし” [mori hayashi]. This effect is called *sequential voicing*, and is one of several in this error category.

“Radical or shape” indicates an error in our transliteration model, since the user formed an unexpected character association not present in our table. “Unknown character” refers to the use of “?” for a character which Pinyomi mistakenly predicted the user could read. For example, one subject had the query “? よう” for the foreign word 称雄, even though the first character 称 is reasonably frequent in Japanese. All other errors are “Miscellaneous”, and include typing mistakes and various unclear cases considered as noise.

7 Discussion and future work

A potential criticism of this work is that whilst it provides better accessibility for languages pairs that are quite close, where often learners could accurately guess a foreign word’s meaning, perhaps our efforts could be better spent on more distinct language pairs. However, even with languages that share words of identical written form, their senses, frequency of use, pronunciation and connotations may still differ wildly. Further, there is indication that guessing *then* consulting a dictionary has advantages in both retention and accuracy over guessing alone (Fraser, 1999). The dictionary is still a time-consuming staple for these learners.

Given script similarity, why is the coverage firstly asymmetrical and secondly not higher? Since Japanese and traditional Chinese share a vast number of characters, and speakers of simplified Chinese also know some traditional characters, they have a strong advantage over Japanese learners of Chinese. Hence their coverage is 83%, compared to 64% for the latter.

As to the general coverage level, we note that the experiment participants overwhelmingly had no previous exposure to the foreign language. In this light coverage is remarkably high. Furthermore, coverage should improve as learners pick up more cognate relationships, although more extensive evaluation is needed to explore the extent of this improvement. For non-learners, our evaluation suggests a theoretical ceiling coverage of around 95% in both directions[6].

Reading error correction is not difficult, and has precedent in the FOKS dictionary system (Bilac et al., 2002). The main challenge lies in reducing “Radical or shape” errors, particularly important for improving Chinese-Japanese lookup. For Pinyomi, use of sub-components seems the most promising avenue to explore.

Finally, this general approach of traversing associations could be extended to other sufficiently similar language pairs. Pairs which share many cognates are ideal, because of the fertile phonetic, orthographic and semantic similarity neighbourhoods they provide. It remains an interesting question as to how to best leverage these similarity associations for each language pair in order to achieve better outcomes for learners or cross-lingual applications.

8 Conclusion

The Pinyomi dictionary system provides a useful and innovative method for dictionary lookup, using the natural associations that learners build as they learn a second language. In this way it achieves excellent rank and decent coverage, even for non-learners, and provides convenient and accessible lookup of Japanese and Chinese words.

Acknowledgements

We are very grateful to Mrs Chooi-Ling Goh for her kindness in providing us with the kanji/hanzi conversion table from Goh et al. (2005), which greatly aided the construction of this system.

References

Slaven Bilac, Timothy Baldwin, and Hozumi Tanaka. 2002. Bringing the dictionary to the user: the FOKS system. In *Proceedings of the 19th International Conference on Computational Linguistics*, pages 85–91. Taipei, Taiwan.

Eric Brill, Gary Kacmarcik, and Chris Brockett. 2001. Automatically harvesting Katakana-English term pairs from search engine query logs. In *Proceedings of the Sixth Natural Language Processing Pacific Rim Symposium*, pages 393–399. Tokyo, Japan.

[6] Assuming that all correctable errors are modelled and compensated for.

Center for Chinese Linguistics. 2007. Chinese corpus. Visited 2007, searchable from `http://ccl.pku.edu.cn/YuLiao_Contents.Asp`.

Olivier Ferret and Michael Zock. 2006. Enhancing electronic dictionaries with an index based on associations. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the ACL*, pages 281–288. Sydney, Australia.

Carol A. Fraser. 1999. Lexical processing strategy use and vocabulary learning through reading. *Studies in Second Language Acquisition*, 21(02):225–241.

Chooi-Ling Goh, Masayuki Asahara, and Yuji Matsumoto. 2005. Building a Japanese-Chinese dictionary using kanji/hanzi conversion. In *Proceedings of the 2nd Internatioal Joint Conference on Natural Language Processing*, pages 670–681. Jeju Island, Korea.

Jack Halpern and Jouni Kerman. 1999. The pitfalls and complexities of Chinese to Chinese conversion. In *Proceedings of the 14th International Unicode Conference*. Cambridge, Massachusetts, USA.

Kevin Knight and Jonathan Graehl. 1998. Machine transliteration. *Computational Linguistics*, 24(4):599–612.

Haizhou Li, Min Zhang, and Jian Su. 2004. A joint source-channel model for machine transliteration. In *Proceedings of the 42nd Meeting of the Association for Computational Linguistics*, pages 159–166. Barcelona, Spain.

Yan Qu, Gregory Grefenstette, and David A. Evans. 2003. Automatic transliteration for Japanese-to-English transliteration. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 353–360. Toronto, Cananda.

Lars Yencken and Timothy Baldwin. 2005. Efficient grapheme-phoneme alignment for Japanese. In *Proceedings of the Australasian Language Technology Workshop 2005*, pages 143–151. Sydney, Australia.