Phrase-based Machine Transliteration

Andrew Finch
NiCT-ATR
"Keihanna Science City"
Kyoto, JAPAN
andrew.finch@atr.jp

Eiichiro Sumita
NiCT-ATR
"Keihanna Science City"
Kyoto, JAPAN
eiichiro.sumita@atr.jp

Abstract

This paper presents a technique for transliteration based directly on techniques developed for phrase-based statistical machine translation. The focus of our work is in providing a transliteration system that could be used to translate unknown words in a speech-to-speech machine translation system. Therefore the system must be able to generate arbitrary sequence of characters in the target language, rather than words chosen from a pre-determined vocabulary. We evalauted our method automatically relative to a set of human-annotated reference transliterations as well as by assessing it for correctness using human evaluators. Our experimental results demonstrate that for both transliteration and back-transliteration the system is able to produce correct, or phonetically equivalent to correct output in approximately 80% of cases.

1 Introduction

Dictionaries and corpora are only able to cover a certain proportion of language. Those words and phrases that are unknown to a translator/machine translation system present a problem. Examples of such words include people's names, place names, and technical terms. One solution to the problem is to transcribe the source language and use the transcription directly in the target language. Usually these transcrptions will be phonetically similar. This process of transcription is known as *transliteration* and in this paper we will present a technique for automatically transliterating between English and Japanese, although the technique is general and is able to be appied directly to other language pairs. Of particular interest to us is the application of such a system within a speech-to-speech machine translation (MT) system. Typically words not seen by the MT system, known as out of vocabulary words (OOVs), are either left untranslated or simply removed from the output. Common examples of OOVs are named entities such as personal names, place names and technical terms, unknown occurences of which could benefit from being transliterated into the MT system's output during translation between Japanese and English. Moreover, in the case of a transation system that translates directly to speech, the transliteration system does not necessarily need to produce the correct transliteration as any one of a set of phonetically equivalent alternatives would be equally acceptable.

1.1 English-Japanese Transliteration

In Japanese there are three separate alphabets, *kanji* (the Chinese character set), *hiragana* (used as an alternative to the kanji, and to express functional elements such as particles etc.) and *katakana* (used to express foreign loan words, and relatively new words in the language, for example "karaoke"). Figure 1 shows some examples, the first line is the English source, the second line is the Japanense and the last line is a direct transcription of the Japanese katakana into the roman alphabet with spaces delimiting the character boundaries. As can be seen from the examples, transliteration is not a straghtforward process. Example 1 of Figure 1 shows an example of a transliteration which is a reasonably direct phonetic transfer. The word "manga" in English is a loan word from Japanese and has more-or-less the same pronunciation in both languages. In Example 2 we have an ambiguity, the "aa" at the end of the word *kompyutaa*, corresponds to the

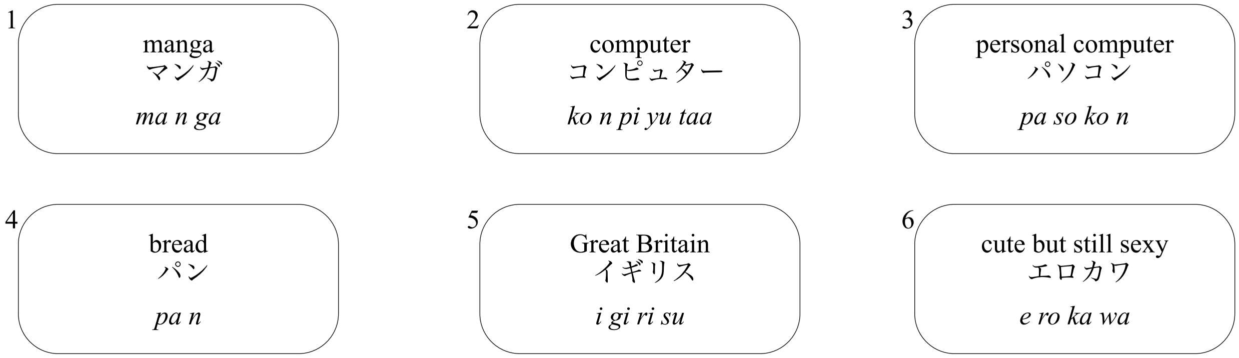

Figure 1: Example English-Japanese Transliterations

"er" of "computer". However, although incorrect the sequences *kompyuta* or *kompyuuta* are also plausible transliterations for the word. Example 4 shows a contraction. The English word has been transfered over into Japanese, and then shortened. In this case "personal" has been shortened to *paso* and "computer" has been contracted into *con*. In Example 4 the Japanese loan word has come from a language other than English, in this case French, and these words are usually transliterated according to the pronunciation in their native language. In Example 5, the etymology is quite complex. The word has entered the language from the Portugese for "English": *inglese*, but has come to mean "Great Britain". Example 6 is a creative modern mixture of an imported loan word *ero* a contraction of the transliteration *erochikku* of the English word "erotic", concatenated with a contraction of the Japanese word *kawaii* (usually written in kanji/kana) meaning "cute". Not only is the English phrase phonetically unrelated in this case, but the expression is difficult to translate without using a number of English words since it represents quite a lot of information.

2 Related Work

2.1 Machine Transliteration

This paper is directly related to an important paper by Knight and Graehl (1996). Their transliteration system was also evaluated by English-Japanese (back-)transliteration performance. Our system differs from theirs in a number of aspects. The most important of which is that their system outputs word sequences whereas our system outputs character sequences in the target language. The difference reflects the intended application of the transliteration system. Their system was intended to transliterate from the output of an OCR system, and must therefore be robust to errors in the input, whereas our system has been developed with machine translation in mind, and the input to our system is likely to consist of out-of-vocabulary words. This flexibility is a double-edged sword in that: on the one hand our system is able to handle OOVs; whereas on the other hand our system is free to generate non-words. A second difference between the approaches is that, Knight and Graehl's model models the pronunciation of the source word sequences using a pronunciation dictionary in an intermediate model. Our system transforms the character sequence from one language into another in a subword-level character sequence-based manner. Our systems relies on the the system being able to implicitly learn the correct character-sequence mappings through the process of character alignment. Our system is also able to re-order the translated character sequences in the output. The system can be easily constrained to generate the target in the same order as the source if necessary, however, often in Japanese names (including foreign names) are written with the family name first, therefore for the purposes of our experiments we allow the system to perform reordering.

2.2 Phrase-based Statistical Machine Translation (SMT)

Our approach couches the problem of machine transliteraion in terms of a character-level translation process. Character-based machine translation has been proposed as a method to overcome segmention issues in natural language

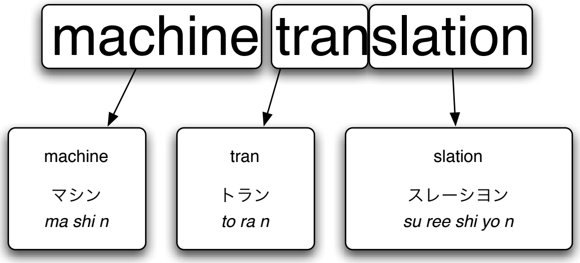

Figure 3: The phrase-translation process

processing (Denoual and Lepage, 2006) and character-based machine translation systems have already been developed on these principles (Lepage and Denoual, 2006). Our system also takes a character-based approach but restricts itself to the translation of short phrases. This is to our advantage because machine translation systems struggle in the translation of longer sequences. Moreover, the process of transliteration tends to be a monotone process, and this assists us further. We will give only a brief overview of the process of phrase-based machine translation, for a fuller account of statistical machine translation we refer the reader to (Brown et al., 1991) and (Koehn, Och, and Marcu, 2003).

During the process of phrase-based SMT the source sequence is segmented into sub-sequences , each sub-sequence being translated using bilingual sequence pairs (called phrase pairs when the translation proceeds at the word-level). The target generation process (for English-to-Japanese) at the character level is illustrated in Figure 3. The example is a real system output from an unseen phrase. The source sequence is segmented by the system into three segments. The translations of each of these segments have been gleaned from alignments of these segments where they occur in the training corpus. For example "machine⇒マシン" may have come from the pair "Turing machine⇒チューリングマシン (*chi yuu ri n gu ma chi n*)" that is present in the Wikipedia component of the training corpus. The "slation" in this example certainly came from the film title "Lost in Translation" since the Japanese translation of the English word "translation" is usually written in kanji.

3 Experimental Methodology

3.1 Experimental Data

The data for these experiments was taken from the publicly available EDICT dictionary (Breen, 1995) together with a set of katakana-English phrase pairs extracted from inter-language links in the Wikipedia[1]. These phrase pairs were extracted in a similar fashion to (Erdmann, et al., 2007) who used them in the construction of a bilingual dictionary. An inter-language link is a direct link from an article in one language to an article in another. Phrase-pairs are extracted from these links by pairing the titles of the two articles. We collected only phrase pairs in which the Japanese side consisted of only katakana and the English side consisted of only ASCII characters (thus deliberately eliminating some foreign language "English" names that would be hard to transliterate correctly). Data from both sources was combined to make a single corpus. Thus corpus was then randomly sub-divided into training (33479 phrase pairs), development (2000 phrase pairs) and evaluation (2000 phrase pairs) sub-corpora. For the human evaluation a sample of 200 phrase-pairs was chosen randomly from the test corpus. In addition a small corpus of 73 US politicians' names was collected from a list of US presidents and vice presidents in the Wikipedia. Duplictate entries were removed from this list and the training set was also filtered to exclude these entries.

3.2 Back-transliteration Accuracy

Following Knight and Graeh (1996), we evaluated our system with respect to back-transliteration performance. That is, word sequences in katakana were used to generate English sequences. As a point of reference to the results in this paper, we back-transliterated a list of American polititians' names. The results are shown in Table 1. The number of exacty correct results is lower than the system of Knight and Graehl, but the total number of correct + phonetically equivalent results is about the same. This can be explained by the fact that our system is able to generate character sequences more freely in order to be able to handle unknown words. Altogether around 78% of the back-transliterations were judged either correct or phonetically equivalent to a correct result. We included a class to respesent those results that were not equivalent in terms of English phonology but

[1] http://www.wikipedia.org

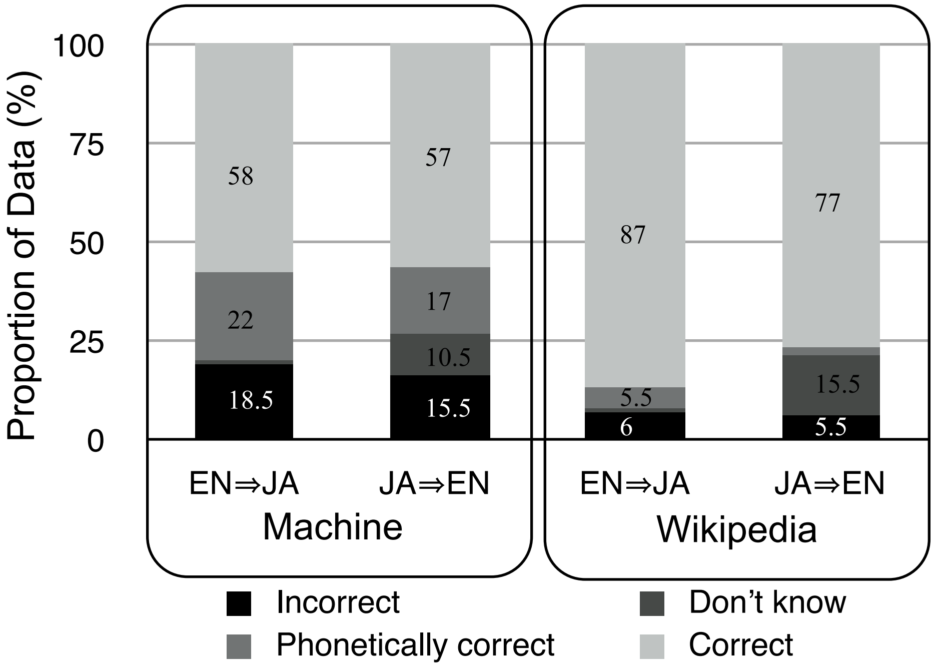

Figure 2: Human Judgement of Quality Transliteration Performance and Wikipedia Data

were “reasonable errors” in terms of Japanese phonology, for example “James Polk” was back-transliterated as “James Pork”, the “r” and “l” sound being hard to discrimitate in Japanese becasue the two sounds are combined into a single sound. The reason for making this distinction was to identify the proportion (around 10%) of more pathological errors caused by errors such as incorrect phrase pairs extracted due to erroneaous word alignments.

3.3 Human Assessment

Figure 2 shows the results of the human evaluation. Transliterated text from English to Japanese was graded by a professional translator who was fluent in both languages but native in Japanese. Conversely the back-transliterated phrases were judged by a native English-speaking translator who was also fluent in Japanese. The evaluation data was graded into 4 categories:

(1) The transliteration or back-transliteration was correct.

(2) The transliteraton was not correct however the result was phonetically equivalent to a correct result.

Correct	57.53%
Phonetically equivalent (EN)	20.54%
Phonetically equivalent (JA)	10.96%
Incorrect	10.96%

Table 1: Back-transliteration performance on politicians’ names

(3) The transliteration or back-transliteration was incorrect.

(4) The annnotator was unsure of the correct grade for that example.

Transliteration examples:

Grade 1: worm gear ⇒ *u oo mu gi ya*
Grade 2: worm gear ⇒ *waa mu gi a*
Grade 3: marcel desailly ⇒ *ma ru se ru de sa i ri*
Grade 4: agnieszka holland ⇒ ?

	BLEU	NIST	WER	PER	GTM	METEOR	TER
EN⇒JA	0.627	9.17	0.31	0.29	0.8	0.81	30.67
JA⇒EN	0.682	10.023	0.277	0.237	0.83	0.81	27.14

Table 2: System performance according to automatic machine translation scoring schemes

The example of Grade 1 is the Wikipedia entry and is the normal way of expressing this phrase in Japanese. The Grade 2 example is output from our system, the pronunciation of the string is almost the same as the Grade 1 version, however the form of expression is unusual. The Grade 3 example is also a system output. Here the system has made a reasonable attempt at generating the katakana, but has transliterated it in terms of the English pronunciation rather than the French from which the name dervies. The correct transliteration from this name would be: *ma ru se ru de sa ii*. This problem has been caused by the nature of the training data which contains mainly English expressions. The word “desailly” had not occurred in the training data.

The results reveal several things about the data, the task and the system performance. Looking at the scoring of the Wikipedia data, there is a reasonable level of disagreement between the two annotators, but the overall number of pairs judged as correct (back-)transliterations is nonetheless reasonably high; in the 80-90% range. Secondly, the annotators judged the quality of the transliteration and back-transliteration systems to be approximately the same. We found this result surprising since the English generation, intuitively at least, appears to be harder than Japanese generation because there are fewer constraints on graphemic structure. The most significant result is that the number of cases labelled “correct” or “phonetcially equivalent to a correct result” was around 80% for both systems, which should be high enough to allow the system to be used in a speech translation system, especially since by visual inspection of the data, many of the the “incorrect” results were near misses that would be easy for a user of the system to understand. For example the transliteraton *ko-roo-ra* for “Corolla” was judged correct, however *ko-ro-ra* was judged incorrect and not phonetically equivalent.

3.4 Assement using automatic machine translation evaluation methods

Table 2 shows the results from evaluating the output of our transliteration and back-transliteration systems according to a range of commonly-used automatic machine translation scoring schemes. We believe these techniques are an effective way to evaluate transliteration quality, and are therefore provided here as a reference. The difference between the WER and PER scores is interesting here as the WER score takes sequence order into account when comparing to a reference whereas PER does not. There is a larger difference when the target is English indicating that this process has more issues related to character order.

4 Conclusion and Future Directions

This paper has demonstrated that transliteration can be done effectively by a machine translation system, and has quantified this empirically. It is clear that by leaving the system ‘open’ and free to generate any sequence of characters in the target language there is a price to pay since the system is able to generate non-words. On the other hand, restricting the system so that it is only able to generate words is for many applications unrealistic, and in particular it is necessary for the speech translation application this system has been developed for. Our results show that our system generates correct or phonetically correct transliterations around 80% of the time. This figure serves as a lower bound estimate for the proportion of practically useful transliterations it will produce. Perhaps a compromise between these two approaches can be achieved by introducing a lexically-based language model into the system in addition to the existing high-order character-based language model. Furthermore, we are also interested in investigating the use of the models generated by training our system in the process of word alignment for statistical machine translation, and as a precursor to this the models might be used in filtering the training data in a pre-processing stage. Lastly it is important to mention that Wikipedia (which provided us with most of our corpus), is growing very rapidly, and considerably more training data for statistical transliteration systems should be available in the near future.

References

J.W. Breen. 1995. Building an electronic Japanese-English dictionary. *Japanese Studies Association of Australia Conference*. Queensland, Australia.

Peter Brown, S. Della Pietra, V. Della Pietra, and R. Mercer (1991). The mathematics of statistical machine translation: parameter estimation. Computational Linguistics, 19(2), 263-311.

Etienne Denoual and Yves Lepage. 2006. The character as an appropriate unit of processing for non-segmenting languages, *Proceedings of the 12th Annual Meeting of The Association of NLP*, pp. 731-734.

Maike Erdmann, Kotaro Nakayama, Takahiro Hara,and Shojiro Nishio. 2007. Wikipedia Link Structure Analysis for Extracting Bilingual Terminology. *IEICE Technical Committee on Data Engineering*. Miyagi, Japan.

Kevin Knight and Jonathan Graehl. 1997. Machine Transliteration. *Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, pp. 128-135, Somerset, New Jersey.

Yves Lepage and Etienne Denoual. 2006. Objective evaluation of the analogy-based machine translation system ALEPH. *Proceedings of the 12th Annual Meeting of The Association of NLP*, pp. 873-876.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical Phrase-Based Translation. *In Proceedings of the Human Language Technology Conference 2003* (HLT-NAACL 2003), Edmonton, Canada.