AN AUTOMATIC METHOD FOR LEARNING A JAPANESE LEXICON FOR RECOGNITION OF SPONTANEOUS SPEECH

Laura Mayfield Tomokiyo and Klaus Ries

Interactive Systems Laboratories
Carnegie Mellon University and Universität Karlsruhe
{laura,ries}@cs.cmu.edu

ABSTRACT

When developing a speech recognition system, one must start by deciding what the units to be recognized should be. This is for the most part a straightforward choice in the case of word-based languages such as English, but becomes an issue even in handling languages with a complex compounding system like German; with an agglutinative language like Japanese, which provides no spaces in written text, the choice is not at all obvious. Once an appropriate unit has been determined, the problem of consistently segmenting transcriptions of training data must be addressed. This paper describes a method for learning a lexicon from a training corpus which contains no word-level segmentation, applied to the problem of building a Japanese speech recognition system. We show not only that one can satisfactorily segment transcribed training data automatically, avoiding human error, but also that our system, when trained with the automatically segmented corpus, showed a significant improvement in recognition performance.

1. INTRODUCTION

How does one go about defining the fundamental units to recognize in a language that does not distinguish word boundaries text? This is the first problem we are faced with when developing a speech recognition system for languages like Japanese, Chinese, or Thai. One can approximate words by mapping onto English words, but this is an artificial solution and may hide characteristics of the language that would help in recognition. One option is to recognize syllables or other small clusters of phonemes, but longer segments are easier to recognize and are better predictors of subsequent words than short ones are. Automatic segmenters based on morphological analysis are available, but in the case of Japanese tend to produce many segments that are very small. Dictionaries can be used to provide stand-alone words like nouns; however, this does not solve the problem of how to break down the lengthy agglutinative inflections that are common in Japanese. Hand-processing requires time and human experts, and we experienced significant inconsistency with manual processing.

In this paper we present an unsupervised method for simultaneously segmenting a raw (non-segmented) corpus and learning a lexicon for recognition. We build on work introduced in [7], in which we find a lexicon that is constructed of morae [1] and sequences of morae (see also Sec. 3). The task of finding sequences is achieved by searching for the two sequences which when joined optimize the perplexity of the bigram model of the training data. We show that we can build a speech recognizer with our original modeling idea [7] and improve the accuracy of recognition as scored at the mora level.

In Sec. 2 we describe related work; in Sec. 3 we describe the segmentation problem for Japanese. Sec. 4 reviews the algorithm and presents some new developments. In Sec. 5, the speech recognition experiments are described, and we conclude with Sec. 6.

2. RELATED WORK

Our process is similar to the procedures described by Lauer [4] and Ito and Kohda [2], but we use the more powerful perplexity evaluation criterion, maximizing the predictive power of the m-gram directly. Ries [11] showed that a variation of this measure outperforms classical measures often used to find phrases. Masataki and Sagisaka [6] describe work on word grouping, although what he describes is critically different in that they are grouping previously defined words into sequences, not defining new words from scratch. Nobesawa presents a method for segmenting strings in [9] which uses a mutual information criterion to identify meaningful strings. He evaluates the correctness of the segmentation by cross-referencing with a dictionary, however, and the approach seems to depend to a certain extent on grammar conventions. Moreover, a breaking-down approach is less suitable for speech recognition applications than a building-up one because the risk of producing out-of-vocabulary strings is higher. (Conversely, with a building-up approach one risks not being able to reproduce all useful sequences.) Teller and Batchelder [13] describe another segmentation algorithm which uses extensively knowledge about the type of a character (hiragana/katakana/kanji, etc). This work, though, as well as Nobesawa's, is designed for processing Japanese text, and not speech.

The problem is not limited to Japanese. Palmer [10] proposes a method for transformation-based segmentation of Chinese and Thai. He evaluates this method by comparing the automatically-produced segmentations against the same text segmented by native speakers. This assumes that there exists a "correct" segmentation that can be consistently reproduced by native speakers, which we did not find to be the case with Japanese. Also for Chinese, Law

This research was performed at the University of Karlsruhe and at Carnegie Mellon University, Pittsburgh. The authors were supported by project VerbMobil through the German BMBF. We gratefully acknowledge their support. The researchers also would like to thank Professor Kurematsu of the University of Electro-communications in Japan for providing the environment for this research as well as valuable advice.

[1] A mora is a unit basically equivalent to a syllable; in most cases one mora corresponds to one character of the Japanese syllabary (kana) and is never more than three phones long.

and Chan [5] combine a measure similar to ours with a tagging scheme since the basic dictionary consisted of 80,000 words.

3. JAPANESE

The Japanese language has historically been written without any white space delineating words.

teemaokimenakerebanaranainodeodenwasaseteitadaitaNdesu
icalledbecausewehavetodecideonthetopic

This means that before one can even start designing a recognition or translation system for Japanese, the *units* that will be recognized, or translated, must be defined. Many character strings, such as nouns, can clearly be stand-alone words, but deciding where to segment verbs and adjectives, which are constructed through an agglutinative inflectional process, is not straightforward.

Japanese has typically been segmented in variations on four ways for the purposes of recognition and parsing. To illustrate these we consider the segment from the above phrase *kimenakerebanaranai*, "(SUBJ) must decide," which our method segments as ⟨kime⟩ ⟨nakereba⟩ ⟨nara⟩ ⟨nai⟩.

Phrase/Bunsetsu level

The bunsetsu is a syntactic unit in Japanese which generally consists of a meaning sequence on the left and a function unit on the right. Bunsetsu are long enough for accurate recognition, and capture common patterns, but require a dictionary entry for each possible phrase, causing a vocabulary explosion. Bunsetsu in our database (described in Sec. 5.1) averaged 10 phones in length. Systems that have used the bunsetsu as a unit of representation include [8] and [3].

kimenakereba naranai

"Word" level

This is a level of abstraction based not on syntactic principles but rather intended to maximize the usefulness of the segment to a speech recognizer. It is this unit that our baseline system uses [12]. "Words" are partially hand-picked to have semantic value, be long enough not to cause confusion, and short enough to capture generalizations. Disadvantages, as with other sorts of knowledge engineering, that manual processing takes time and can be inconsistent.

kime nakere ba naranai [2]
kime nake reba nara nai
kime nake re ba nara nai
kime na kere ba nara nai

Morpheme level

The morpheme is a syntactic unit that is well-defined in principle but can be difficult for non-experts to apply (do alternating endings go with the verb stem or the inflection that requires them?). Morphological taggers are quite good and useful for segmentation, although if segmented in the strictest sense morphemes can be single phonemes, which we prefer to avoid when possible. It is possible to augment commercially available taggers with rules indicating which inflections should be left attached to the root, for example, but for a limited-domain system like the one we work with, we would not want to apply these rules unconditionally but rather based on features of the data. This method also does not tell us which sequences are important in our domain, which is a benefit of our approach. The ungrammatical character of spontaneous speech can also complicate syntactic analysis.

kime na kere ba na ra na i

Phoneme cluster level

Since the set of morae in Japanese is small and closed, this is one level of abstraction that can be extremely useful. The JANUS KSST[3] system uses the syllable, the set of which is much larger than the set of morae in Japanese, successfully in recognition of Korean. (This system started out recognizing at the bunsetsu level, but the vocabulary growth quickly became unmanageable.) Using the syllable, or mora, as the basic unit of recognition means that one needs only a very short dictionary; disadvantages include high confusability, although acoustic confusability seems less of a problem for Japanese than for some other languages.

ki me na ke re ba na ra na i

4. ALGORITHM

The algorithm we use to find mora sequences starts at the mora level and builds up sequences of mora units bottom up. Initially the whole text consists of "multi-moras" that consist of exactly one mora. In every single step we try to combine a pair of mora to form one new "multi-mora": We select the pair of multi-moras that will give us the best bigram model if we replace all instances of this pair in the database by a new multi-mora symbol. This new multi-mora can be used in future joins of multi-moras. To build more and longer sequences we replace all instances of the best pair with a new multi-mora and continue from the beginning.

Ries [11] shows that the best pair of mora to join can be determined quickly by doing leaving-one-out estimate of the bigram perplexity of all models resulting from a join of any two multi-mora. This is calculated efficiently from a trigram table of the corpus. The estimation of the final model is also fairly simple: we mark the multi-mora in the corpus and use a standard backoff model.

Recently, we have tried a number of generalizations to this algorithm. An assumption we originally shared with Lauer [4] was that it is not always appropriate to reduce a pair of sequences to a new sequence; it might be better to replace the pair by one of the components of the pair. This idea is reminiscent of the head principle in linguistics. We constructed an algorithm that could map the pair either on a new symbol or on the left or the right element of the pair. When we applied this algorithm to sequence finding of English words we found that (a) most of the time a new symbol was created and (b) the generalization did not yield better results.

In another set of experiments we tried to separate two effects of the sequences on the estimation. The first effect is that we are estimating models that are of higher word-order when we use sequences. The second effect is that the fallback steps a backoff

[2] Examples of all four segmentation patterns appeared in our human-transcribed data.

[3] Korean Spontaneous Scheduling Task; SST described more fully in Sec. 5.1

model takes are different when sequences are part of the context. We have observed that we can improve results on Switchboard (SWB) [1]: we can get very good perplexity results if we do not extend contexts that already use a trigram context on the word level. We were not, however, able to improve recognition accuracy using this technique. The application of these and similar models to Japanese is future work.

5. EXPERIMENTS

We performed several experiments to evaluate the appropriateness of our approach to segmentation of Japanese. As our operations were text-based, we first examined the effects of different sequencing procedures on the language model by measuring the change in bigram perplexity of the resulting corpus. This showed us how much the predictive power of our model was improved with the addition of the different sequences. We then trained a system based on the learned lexicon corresponding to the best (lowest perplexity) language model and measured the effect on recognition performance.

Only experiments involving romanized corpora are presented. Kana (romanized or not) are themselves useful levels of abstraction, and working with kanji would introduce other problems, not the least of which being deciding which readings to assign singleton kanji or kanji combinations. Kanji are an extremely informative form of representation, though, and we will continue to look for ways to incorporate them in future work.

5.1. Task

The Spontaneous Scheduling Task (SST) databases are a collection of dialogues in which two speakers are trying to schedule a time to meet together. Speakers are given a calendar and asked to find a two-hour slot given the constraints marked on their respective calendars; the majority of negotiations take between 15 and 25 turns. Dialogues have been collected for English (ESST), German (GSST), Spanish (SSST), Korean (KSST) and Japanese (JSST). The entire JSST database consists of 800 dialogues.

5.2. LM-level

5.2.1. Test corpora

Six language models were created for the scheduling task JSST [12]. The models were drawn from six different segmentations of the same corpus, as described below. The mora segmentation task was completely unambiguous given our transcription conventions. Sequences were found using the compounding algorithm described in Sec. 4.

C1: Only romanized mora syllables. A romanization tool was run over the original kanji transcriptions; the romanized text was then split into kana (morae).

C2: C1 with sequences incorporated.

C3: Sequences that were learned *before* romanization. The sequenced kanji text was then run through the same romanization tool.

C4: A hand-edited version of C3, where some word classes (like day of the week - if only "tuesday" existed in the corpus the rest of the days were added by hand) were fleshed out and meaningless sequences removed.

C5: The hand-segmented text used in the current JSST system

C6: C5 + sequences from C4

5.2.2. Perplexity Results

If the standard test corpus has length n and the new test corpus has length n' we define for the test corpus

$$PP^{rel} = PP^{\frac{n'}{n}}.$$

The relative perplexities reported below are all normalized with respect to corpus C1, measured for an unseen test set.

The results in Table 1 clearly indicate that we can do at least as well or even better than human segmentations using automatically derived segmentations from the easily definable mora level. Note that the sequence trigram is better than a 4-gram; this indicates that the sequences play a critical role in the calculation of the model.

5.3. Recognition

The ultimate measure of improvement in a speech recognition system is word accuracy. Significant reductions in perplexity of a language model have been known to have little or no effect on the recognition performance of the system. If our modifications had resulted in a decrease in recognition accuracy, we would suspect that an automatic segmentation method cannot match the power of human experts, despite the drawbacks of knowledge engineering. We found, however, that training with the re-segmented training data and learned lexicon gave a small *increase* in word accuracy, allowing us to conclude that ours is a solid technique.

5.3.1. System Description

The systems used in this experiment were derived from a context-dependent system developed using the Janus Recognition toolkit (JRTK) and described in [12]. The speech data was re-labeled retrained using the new dictionary and training corpus, the codebooks clustered and a second training iteration performed. The same training sequence was done on the original system to eliminate bias due to additional training; it is this re-trained system that we refer to as the baseline system. Training data was a reduced set of 190 dialogues.

The database used to develop the original system was segmented using the Chasen morphological tagger. A native expert was needed to check and correct the output of this tool.

5.3.2. Recognition Results

Testing this system on a test set of 10 unseen dialogues gave a relative improvement of 13.0% in word accuracy over the baseline system (Table 2). We evaluated mora accuracy, breaking both sets of hypotheses down to the mora level.

	Baseline	New
Word accuracy	88.2	89.6
Perplexity (rel.)	5.775	4.447
Percent OOV	1.493	0.03039

Table 2: Comparison of performance on an unseen test set. Note that this was not the same test set as was used in Table 1.

	mora			kanji	hand-edit	"words"	
	C1	C1-4gram	C2	C3	C4	C5	C6
PP^rel	6.1	4.7	4.5	4.7	4.6	6.3	6.0
corpus size	38963	39995	16070	19400	19135	25951	25575
vocab size	189	189	1058	1118	977	2357	3286

Table 1: Comparison of trigram language models built with the six corpora described in Sect. 5.2.1
.

This is an exciting result because it shows that we can use an automatically segmented training text for developing a Japanese speech recognition system with no negative effects on recognition performance. Note also the low out-of-vocabulary rate. Since the set of morae in Japanese is small (under 200) and closed, we can include all morae in the dictionary as our basic units of recognition, thereby eliminating the possibility of out-of-vocabulary (OOV) words[4].

One of the concerns that we had about using morphologically-based segmenting tools was that too many small units were being produced. Table 3 gives word length figures for the baseline and new systems, and we can see that word length increased and number of extremely short words decreased in the new system. Words in the hypotheses produced during decoding were on average longer than those in the training data.

	Baseline		New	
	training	decoder	training	decoder
Ave. word length	3.92	4.10	5.0	5.77
2-3 phone words	32534	3063	18289	2212
1-phone words	3000	115	1470	25
Total words	72598	1231	57947	525

Table 3: Word length statistics from training data and decoder output.

6. CONCLUSION

We have presented a statistical method for learning a lexicon and segmenting a training corpus for development of a Japanese speech recognizer, and shown that recognition performance on the system developed using this automatically produced lexicon and training data outperformed a comparable system trained with morphologically-segmented and hand-edited data. Such a method is valuable because it allows us to bypass time-consuming and inconsistent manual processing and produces units that are of a desirable length and composition for speech recognition. We evaluated our segmentation technique using the well-known measures of perplexity and word accuracy. We believe that we were able to achieve these good results because our technique uncovers characteristics of spoken Japanese that help in recognition.

We are very much interested in representing kanji directly, instead of working with romanized text, and are working toward this goal. Early experiments with finding sequences of Japanese characters were not encouraging, but this may be because the scoring convention does not require that the kanji be correct, only that the transliteration be accurate, creating a bias against the more difficult kanji recognition task. Additional directions for future work include application of this technique to other languages and experimentation with other sequence-finding techniques.

[4]In theory, OOV for the new system should be zero; all OOV occurrences here were caused by transcription errors in the training data.

7. REFERENCES

[1] Geutner, Petra and Rob Malkin and Klaus Ries. The JanusRTk Switchboard/CallHome System – Language Modeling. In *Proceedings of LVCSR Hub 5 Workshop*, May, 1997.

[2] Ito, Akinori and Masaki Kohda. Language Modeling by String Pattern N-gram for Japanese Speech Recognition. In *ICSLP*, 1996.

[3] Kameda, Masayuki. A Portable & Quick Japanese Parser: QJP. In *COLING*, Copenhagen, 1996.

[4] Lauer, Mark. Corpus Statistics Meet the Noun Compound: Some Empirical Results. In *ACL*, 1995.

[5] Law, Hubert Hin-Cheung and Chorkin Chan. Ergodic Multigram HMM Integrating Word Segmentation and Class Tagging for Chinese Language Modeling. In *ICASSP* 1996, Vol.1, pp. 196-199.

[6] Masataki, Hirokazu and Yoshinori Sagisaka. Variable-order N-gram Generationi by Word-class Splitting and Consecutive Word Grouping. In *ICASSP* 1996, Vol. 1, pp. 188-191.

[7] Mayfield Tomokiyo, Laura and Klaus Ries. What's in a Word: Learning Base Units in Japanese for Speech Recognition. In *Proceedings of the ACL Workshop on Natural Language Learning*.

[8] Morimoto, Tsuyoshi et al. ATR's Speech Translation System: ASURA. In *Eurospeech*, 1993.

[9] Nobesawa, Shiho et al. Segmenting Sentences into Linky Strings using D-bigram statistics. In *COLING*, Copenhagen, 1996.

[10] Palmer, David. A Trainable Rule-based Algorithm for Word Segmentation. In *ACL*, Madrid, 1997, pp. 321-328.

[11] Ries, Klaus and Finn Dag Buø, and Alex Waibel. Class Phrase Models for Language Modeling. In *ICSLP*, Philadelphia, 1996.

[12] Schultz, Tanja and Detlef Koll. Spontaneously Spoken Japanese Speech Recognition with Janus-3 In *EUROSPEECH*, Rhodes, 1997.

[13] Teller, Virginia and Eleanor Olds Batchelder. A Probabilistic Algorithm for Segmenting Non-Kanji Japanese Strings. In *AAAI* pp. 742-747, Seattle, 1994.