A Multi-lingual Segmenter by Using Viterbi Algorithm

Dept. of Electrical Engineering and
Computer Science
University of Michigan

Dept. of Linguistics
University of Michigan

Abstract

The segmentation of texts into words is generally regarded as the first step in processing languages written without spaces, such as Chinese and Japanese. Also, it is a supplemental processing step in languages written with spaces. In Koreantexts, the segmentation can be used to correct spacing errors. In this paper, we present a multilingual segmenter by implementing Viterbi algorithm and supplementing it with inword probability and automatically extracted linguistic rules. We apply the segmenter to both Chinese segmentation and Korean spacing task. Experimental results show the efficiency of the multi-lingual segmenter. For Chinese segmentation task, both the training data and the test data are from PKU. The recall, precision, and F-measure of our system are 0.95, 0.941, and 0.946 exclusively. For Korean spacing task, we use Korean news articles. The F-measures of
two different criteria are 0.902 and 0.958.

1 Introduction

Word segmentation is to find word boundaries not marked by any delimiters in texts. It is the first step of text processing in languages without spaces as delimiters such as Chinese, Japanese and Thai. The fact that a sequence of characters can be grouped in several ways makes the segmentation task difficult. There is a sizable literature dealing with word segmentation. For example, Chinese word segmentation has been studied for decades with varieties of methods (Fan and Tsai, 1988; Sproat et al., 1996; Wu,
2003).

Even for those languages with delimiters, word segmentation is also a necessary step in text
processing. In Korean, though spaces are delimiters of Eojeol1 boundaries, automatic spacing is required in sentences with spacing errors in order to increase the readability and communication accuracy. Spacing errors cause misinterpretation to readers. For example, if (once in a while) is written as (to go once in a while), both its meaning and part-of-speech change. Spacing errors are common in Korean. It is hard to get correct word spacing even for human because people tend to use morphemes incorrectly. Consider (whole family). (whole) is an adjective and it should be detached from (family), but people often misuse as a noun, and they either attach or detach to. 2 In the literature, several methods have been used to deal with spacing (Kang et al., 2001; Lee et al., 2003). The task of word spacing can be taken as the task of word segmentation (Lee et al., 2003). In order to space a text, all the spaces are eliminated first and then the text is segmented into Eojeols delimited with correct spaces.

In this paper, we built a multilingual segmenter for Chinese word segmentation task and Korean spacing problem. We applied Viterbi algorithm (Viterbi, 1967), inword probability (Chen, 2003) and automatic linguistic rules as a plus.

2 Viterbi Algorithm

Viterbi algorithm is a dynamic programming algorithm to find the best path through a probabilistic network given observed evidences. In word segmentation task, given a string of characters n c c C ,..., 1 = , there are one or more possible segmentations m w w W ,..., 1 = based on the dictionary with unigram word probability distribution. Viterbi algorithm picks up the segmentation with the highest probability as in (1).
) | (max arg 1 1 1 n m w w c c w w P m L L L (1) To find the best path, Viterbi algorithm computes the probability of all possible segmentations and ends up in the probability) (.. 1 j s P of the most probable string spreading from the start up to each position j, which can possibly be updated dynamically by higher probability later on. Equation (2) illustrates the iteration process to calculate and update the best probability.) ( )), ( ) (max( ) (
.. 1 .. 1 .. 1 .. 1 j j i i j s P P P s P s P s P = = (2) where j i s .. is the fragment of string starting from ith character and ending with jth character in the input. If) (.. j i s P) 1 (j i is used to update) (.. 1 j s P , j i s .. is the best word ending with jth character and it is stored in j w , the word array that contains the best word ending at position j. Table 1 illustrates how to compute the best probability among all possible words in Chinese string. Take position j=4 as an example. When the input is 4 .. 3 s,) (4 .. 1 s P is the product of P and P, and the 4 s stores as the best word. However, when input string is the last character 4 .. 4 s, the) (4 .. 1 s P yields to the product of P and P) which produces the highest probability. Therefore the input string is segmented into. After the best probability and best words are
ready, the algorithm then searches backwards through words to return the best probability path. For words not in the dictionary, we used add-one smoothing by adding 1 to frequencies of all entries and increasing total frequency N accordingly.

3 Inword probability

For words not in the dictionary, Viterbi algorithm prefers the longest fragment. If 1 u , 2 u , and 2 1u u are all unknown words, Viterbi algorithm segments the string into 2 1u u because) (2 1u u P is higher than) ( ) (2 1 u P u P based on add-one smoothing. However, the longest segment is not necessarily correct. Thus, we split all the words which are not in the dictionary into single characters, putting off the decision on whether to combine those single characters in later steps.

In the second step of our system, we applied inword probability to combine a sequence of single characters into words. The in-word probability of a character is the probability that the character occurs in a word whose length is more than one, as in (3).) ( ) ( ) (c freq c freq c P inword inword = (3) where) (inword c freq is the number of times that a character c occurs inword, and) (c freq is the number of times that c occurs in training data.

We built up an inword probability hash table for every character in the training data. Consecutive single characters are combined into a word if the inword probability of each character is over the threshold. According to our experiment over training data, we set 0.84 for Chinese inword probability threshold and 0.90 for Korean. Take a string of two single Chinese characters (family) (banquet) as an example. Our dictionary has no entry for, but the inword probability of is 0.87 and of is 0.93. So the two single characters are segmented as one word.

We extended inword probability to the recognition of numeric type compounds, including number compounds and time compounds. Since ASCII numbers are not included in our Chinese

dictionary, the segmentation of numbers becomes the task of new words recognition. We combined the consecutive single numbers, including both digital numbers and Chinese character numbers together as one single word. We set inword probability for numbers as 1.0 if it is preceded or followed by another single numeric character or a certain suffix with inword probability assigned as 1.0, such as %. If the string is 1 2 %, then the inword probability will combine the three single characters together as one word 12% because all the inword probabilities of the three single characters are 1.0, above the threshold. If the compound of numbers is followed by a time unit, such as (year), (month), (day), the inword probability of the time unit is also assigned to 1.0. In this way, date and time compounds are combined into one word. For example, 2003 ? is segmented as 2003 ? (the year of 2003).

In Korean Eojeol spacing, inword probability helps to combine unknown transliterated names

such as (carpet), (Persian). In , the two syllables3 and have inword probability above 0.98. A syllable has much higher inword probability when it is specifically used for pronunciation of a foreign character, because foreign characters often have unusual combinations of consonants and vowels.

4 Further recognition of unkown words

4.1 Unknown words for Chinese

In Chinese segmentation task, we applied linguistic knowledge as the final procedure to recognize unknown words after the implementation of in-word probability. First, we collected 50 suffixes from training data by implementing simple suffix extractor. The set of suffixes covers district units such as (country), (province), geographic suffixes such as (river), (mountain), road suffixes such as (road), (lane), and other suffixes such as (prize), (team). We attached a suffix to the previous word of two characters. Secondly, we extracted 100 family names from the training data. In the PKU training data, family names are separated from given names. If A is a family name in a sequence of single characters A B C, B and C are combined together as a given name only if C is not a family name and C does not belong to a small set of context words, such as (say), (and). The first restriction on C avoids the wrong segmentation of concatenated person names. In the sequence since the third character is also a family name, the given name segmentation does not combine and, because the latter could be the family name of another person name followed by.

4.2 Unknown words for Korean

As mentioned in the introduction section, Korean Eojeols can consist of more than one word. In the type of Eojeols consisting of a noun and a postposition, a certain noun can be attached by any postposition. For instance, the noun (lecturer) can be combined with different postpositions, and thus different Eojeols and meanings are generated. To list a few: is direct objective postposition), (of the lecturer), (to the lecturer), and (with the lecturer). This type of arbitrary combinations undoubtedly

yields some Eojeols not in the dictionary. We extracted a set of postpositions and built up the postposition attachment rule to solve this problem. If C in the segmented output A/BC after the second step is a postposition, the Eojeol A is combined with the Eojeol BC only if) (AB P is higher than) ( ) (B P A P ? . Thus if AB is a known word and B is an unknown word, the system correctly combine AB and C. Suppose that we have dictionary entries for and no entries for and. If an Eojeol composed of a noun (lecturer) and a subjective postposition, is the segmented target, we cannot get the correct segmentation based on our dictionary. The output of Viterbi algorithm is. The inword probability cannot correct the segment because is a known word. Our rule can correct the segment because the probability P is higher than P*P, then we attach to.

While resolving this problem, we followed the same procedure as the Chinese suffix extraction to maintain the consistency of our system. We collected set of postpositions from training data. Those postpositions are used to mark subject, direct object, indirect object, possession, location, direction, means, and groupings.

5 Experiment and result

5.1 Chinese word segmentation

Both the training data and test data are from PKU corpora used in first international Chinese segmentation bakeoff4. The training data has 1.1M words and the test data has 17K words. The encoding of the corpora is simplified GBK. First, we made a unigram dictionary with distribution probability from the PKU training corpus. We did not include number digits and English letters in the GBK code, nor did we collect ASCII sequences. We also built up an inword probability list for each character in the training data. We did three steps in Chinese word segmentation. First, we applied Viterbi Algorithm. Secondly, we combined sequence of single unknown characters using inword probability. Finally we applied automatic rules of suffixes attachment and person names. Table 2 shows the recall, precision and Fmeasure after each step.

The performance of our system is promising compared to the first International Chinese segmentation bakeoff (Sproat et al., 2003). Table 3 shows the baseline, average and the highest scores of the bakeoff.

Among the errors our system produced, a small portion was caused by the inconsistent examples found in annotated segmentation between the training and test data, and the major errors came from our system.

The performance of the segmenter increased 4.9% after we implemented the inword probability
algorithm. However, we could not avoid creating some segmentation errors. For example, the
sequence of (cat) (winter) was combined incorrectly together as a single word because both of them have high inword probabilities. In the third step, segmentation errors appeared due to overgeneration and undergeneration of linguistic rules. The suffix attachment rule can be overgenerated when a suffix has ambiguous meanings. has two meanings, one means festival, used as a suffix, and the other means phase, used as a common noun. We did not make use of any context clues to distinguish the two meanings of. As a result, we attached in some wrong cases. The system segmented (the first phase), but the correct segmentation should be. As for segmentation of person names, we restricted that the third character could not be a family name. This rule can undergenerate some given names because characters used for family names can be used in given names also. In the name string, the last character is a family name, so this string was not recognized as a person name in our system. Some other errors were from segmentation of transliterated person names.

5.2 Korean Eojeol spacing We used training data from a collection of Korean Press Agency news articles from January 3, 1999, to December 31, 1999. We applied the same steps as we did on the Chinese segmentation. We collected unigram Eojeol probability and inword probability from the training corpus. Unlike Chinese segmentation, in Korean Eojeol spacing task, there are no specialized test collections for the experiment. We chose the last article from March 20, 2000 in the same collection for the test data. We constructed our test data by eliminating all the spaces existing in the original news article. We segmented the test data and compared the result with the original article. Table 4 shows the evaluation result.

When evaluating the Korean spacing, both syllable based precision and Eojeol based precision
can be used. Syllable based precision is the ratio of the number of correctly spaced syllables over total number of syllables. Eojeol based precision evaluates the ratio of the correctly spaced Eojeols over the Eojeols from the system output. We adopted Eojeol based precision to maintain the consistency with Chinese segmentation domain.

In the evaluation of Korean spacing, a compound noun can be treated either as a whole or separate nouns. Thus we relaxed the definition of a compound noun to either case, and did the evaluation again. For a compound noun with three nouns (high-speed internet), and are also counted as the correct segmentations. Table 5 shows the improvement with the relaxation.

6 Conclusions

In this paper, we built a multi-lingual segmenter for both Chinese segmentation and Korean spacing task by implementing Viterbi algorithm and supplementing it with inword probability and automatic linguistic rules. It is the first try to segment languages with word boundaries and without boundaries in one segmenter. Experimental results show the efficiency of the multipurpose system.