Word-based Partial Annotation for Efficient Corpus Construction

Graham Neubig Shinsuke Mori
Kyoto University
Yoshidahonmachi, Sakyo-ku, Kyoto, Japan
neubig@ar.media.kyoto-u.ac.jp forest@i.kyoto-u.ac.jp

1 Introduction

While empirical methods have led to great advances in natural language processing (NLP) technology, the cost of annotating training data for such methods can be immense. This is particularly true when applying methods to specialized domains, as annotators that are both familiar with the annotation standard and the target domain are hard-to-find and costly. When a corpus exists in the general domain, it is possible to create a partially annotated corpus in the target domain that augments the domain-specific areas not covered well by the general domain corpus. This helps reduce the amount of annotation required, allowing for efficient development of linguistic resources.

Much previous work in partial annotation has used the sentence as the basic unit of annotation (Ringger et al., 2007). This allows for easy integration with existing language processing techniques, which generally use sentences as the unit for training and analysis. However, as only one or two words in any particular sentence may be ambiguous, annotating entire sentences results in annotation of large amounts of redundant information. An obvious solution to this problem is to annotate not whole sentences, but only specific words that are able to effectively increase the accuracy of a system of interest.

In this paper, we examine the benefit of annotating only particular words in a sentence, as opposed to the entire sentence. Using the task of Japanese pronunciation estimation as an example, we devise a machine learning method that can be trained on data annotated word-by-word. This is done by dividing the estimation process into two steps (word segmentation and word-based pronunciation estimation), and introducing a point-wise estimator that makes each decision independent of the other decisions made for a particular sentence. In an evaluation, the proposed strategy is shown to provide greater increases in accuracy using a smaller number of annotated words than traditional sentence-based annotation techniques.

2 Task Definition and Traditional Methods

Estimating the pronunciation of words in written text is essential for creating systems such as text-to-speech (TTS) or automatic speech recognition (ASR). In languages such as Japanese, which are written without spacing between words, it is also necessary to segment unspaced character strings into appropriate units. Thus, the process of estimating the pronunciation of a Japanese essentially consists of two steps:

- Word Segmentation (WS): Dividing an unspaced character string into appropriate units.
- Pronunciation Estimation (PE): Estimation of the pronunciation of each segmented unit.

Traditional methods for Japanese PE generally treat WS and word-based PE as a single step, going directly from an unsegmented character string to a string of word/pronunciation pairs.

2.1 Sequence-based Models

Previous systems for Japanese PE generally use sequence-based methods such as n-gram models (Nagano et al., 2005). In addition, CRFs on POS-word pairs for morphological analysis (Kudo, 2009) can be extended to PE, but as of yet no experimental result of CRFs on PE have been reported. For training, methods such as n-grams and CRFs generally require fully annotated training sentences to estimate parameters. Morphological analyzers (Kudo, 2009), which are often used for PE, further require that sentences be annotated with parts of speech and other linguistic information.

In the evaluation, we used a tri-gram model based on word-pronunciation pairs as a representative of these sequence based models. This is similar to the method proposed in (Nagano et al., 2005)[1].

2.2 Required Resources

Generally speaking, three types of resources are required to train models for pronunciation estimation.

[1] which additionally uses triplets including accent sequences, or quadruplets including POS info.

- **Annotated general-domain corpus:**
A large corpus in the general domain annotated with word boundary information and phoneme sequences for each word.

- **Word pronunciation dictionary:**
A set of pairs containing written words and phoneme sequences. A particular surface form may correspond to multiple pronunciations.

- **Character pronunciation dictionary:**
A dictionary containing all possible phoneme sequences for each single character. This is used to predict the pronunciation of unknown words.

2.3 Domain Adaptation

In order to adapt a PE model to a new domain, in addition to the three general domain resources we need:

- **Annotated target-domain corpus:**
A small corpus in the target domain annotated with word boundary information and phoneme sequences for each word.

However, annotators capable of creating this data efficiently must be specialists in the target field that are familiar with the word segmentation standard and pronunciations, which makes creation of target domain annotated data costly. Thus, by reducing the amount of data required, we can reduce the amount of time required to annotate it, reducing the barriers to domain adaptation.

3 Point-wise Estimation and Word-Based Annotation

This paper proposes the combination of two separate techniques to achieve more efficient corpus annotation: point-wise estimation and word-based annotation.

3.1 Point-wise Estimation

While sequence-based methods require fully annotated training sentences, many sentences only contains a few ambiguous word boundaries or pronunciations. The fact that these systems are only able to utilize full-sentence annotations results in large amounts of unnecessary work on the part of the annotator, as only the ambiguous boundaries are likely to lead to significant increases in accuracy. One exception is Tsuboi et al. (2008), which proposes a method to learn CRFs from a corpus with partially-annotated sentences. This, however, requires a large amounts of memory and time when trained on very sparsely annotated sentences.

As an alternative method, we propose the learning of a point-wise estimator. Point-wise estimation assumes that every decision about a segmentation point or word pronunciation is independent from the other decisions. For example, the decision whether a word boundary lies between characters x_i and x_{i+1} can depend on any number of features based on the surrounding characters, but not on whether a boundary lies between characters x_{i-1} and x_i. Because each decision is independent, models can be trained on single annotated words, even if the neighboring words are not annotated. This is the key to the proposed word-based partial annotation strategy.

3.2 Point-wise Japanese PE

In order to make point-wise estimation possible, we divide sentence-based PE into two steps, WS and word-based PE.

Given an unsegmented character string $X = x_1, x_2, \ldots, x_n$ as input

$X =$ 大 分 は 今 日 は 快 晴 で す,

the characters are segmented into words by estimating whether a boundary between x_i and x_{i+1} exists for each i, $1 \leq i < n$. Using this information, we can acquire a segmented word string W

$W =$ 大分 は 今日 は 快晴 で す.

Next, each word is annotated with possible pronunciations using the corpus and dictionaries

> 大分/{o o i ta, da i bu} は/{ha} 今日/{kyo u, ko n ni chi} は/{ha} 快晴/{} で/{de} す/{su}.

Words with only one possible pronunciation (e.g. は/{ha}) are annotated with that pronunciation. For words with several possible pronunciations (e.g. 大分/{o o i ta, da i bu}), an estimator is trained to decide between the multiple possible pronunciations. The pronunciation of unknown words is estimated using a character-based n-gram model similar to one described in Section 2.1. This process is described in more detail in Figure 1.

This process of WS followed by PE results in a string of words, each annotated with a single pronunciation

> 大分/o o i ta は/ha 今日/kyo u は/ha 快晴/ka i se i で/de す/su.

3.3 Features/Implementation for WS

In order to test the partial annotation strategy, we developed a point-wise estimator for WS and PE that is able to be trained on partially annotated data[2]. For WS, the following features are used:

1. **Character n-grams:** Character n-grams surrounding the character boundary to be estimated. A window width of w characters is

[2] Source code and documentation are available at `http://www.phontron.com/kytea`

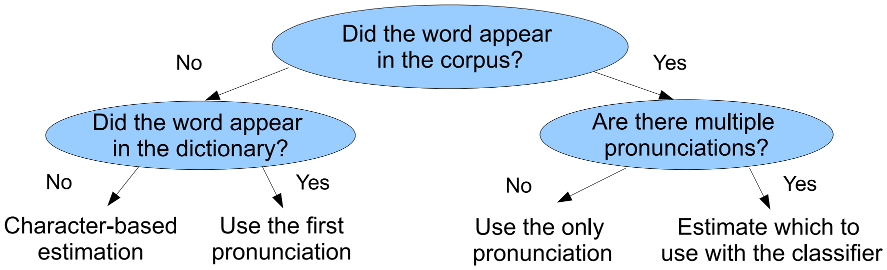
Did the word appear in the corpus?
No
Yes
Did the word appear in the dictionary?
Are there multiple pronunciations?
No
Yes
No
Yes
Character-based estimation
Use the first pronunciation
Use the only pronunciation
Estimate which to use with the classifier

Figure 1: The pronunciation estimation process.

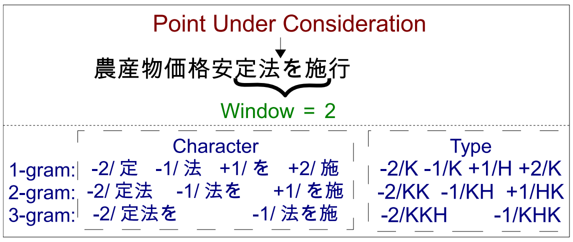
Point Under Consideration
農産物価格安定法を施行
Window = 2
Character
Type
1-gram: -2/ 定 -1/ 法 +1/ を +2/ 施
-2/K -1/K +1/H +2/K
2-gram: -2/ 定法 -1/ 法を +1/ を施
-2/KK -1/KH +1/HK
3-gram: -2/ 定法を -1/ 法を施
-2/KKH -1/KHK

Figure 2: Character and character type 3-gram features with a window width of 2.

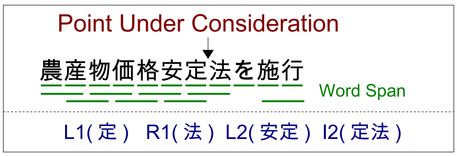
Point Under Consideration
農産物価格安定法を施行
Word Span
L1(定) R1(法) L2(安定) I2(定法)

Figure 3: Dictionary word features.

selected, and only characters within this window are used in analysis, allowing for the limiting of the parameter space (see Figure 2.)

2. **Character type n-grams:** Japanese characters can be broadly grouped into 6 types, Chinese ideographs, *katakana* (used in phonetic spelling of foreign names), *hiragana* (used in conjugation and Japanese origin words), roman characters, digits, and other characters. As changes in character type often indicate word boundaries, this information is useful to incorporate in a word segmentation system. Character type information was incorporated by adding features similar to the previously mentioned character n-grams, only replacing the character itself with the character type.

3. **Dictionary Word Features:** In order to allow for the use of a dictionary in the segmentation process, three types of features describing the presence of words in the dictionary are included in the feature set. First, when judging about whether a word boundary exists between characters x_i and x_{i+1}, if a word starts at character x_{i+1} a feature "R" (right) indicating a word exists to the right of the point is added. Likewise, if a word ends at character x_i a feature "L" (left) is added. Finally, if a word spans both x_i and x_{i+1}, a feature "I" (included) is added. All of these features are annotated with the length of the word under consideration, so if a word of three characters ends at character x_i, a feature "L3" will be included in the estimation process. An example of these features is shown in figure 3.

These features are calculated for each potential word boundary between x_i and x_{i+1}, and a classifier is used to determine the presence or absence of the word boundary. In the evaluation presented in Section 4, the linear support vector machine (SVM) implementation provided by the LIBLINEAR software package (Fan et al., 2008) was used to solve this classifying task.

SVMs are well suited to this task, as focusing on the annotation of difficult or rare cases tends to increase the number of outliers that are included in the training data. Many machine learning methods are heavily influenced by these outliers, resulting in a reduction of accuracy on more common phenomena. SVM, on the other hand, are relatively robust to the inclusion of rare instances, resulting in little reduction of accuracy, even when these rare cases are included.

3.4 Features/Implementation for PE

Words that have multiple readings in the corpus are estimated using a classifier trained on the instances in the corpus. This classifier uses the character n-grams and character type n-grams described in the previous section. Words that only have a single pronunciation in the corpus, or only occur in the dictionary are given that reading. Unknown words are estimated using a character-based language model annotated with character/pronunciation pairs.

4 Evaluation

We evaluated the effectiveness of the word-based partial annotation strategy on a Japanese PE domain-adaptation task.

4.1 Conditions of the Experiments

The general-domain resources used are as follows:

- **Balanced Corpus of Contemporary Japanese (BCCWJ):** A 898k word general-domain corpus fully annotated with word boundaries and pronunciations (Maekawa, 2008).
- **UniDic (Version 1.3.12):** An 212k word dictionary of Japanese words and proper names with an average of 1.05 pronunciations per word (Den et al., 2008).
- **Number Dictionary:** A dictionary of two digit and four digit numbers that often appear as years.

For domain adaptation and testing, we created a corpus of fully annotated sentences from the Nikkei newspaper with word boundaries and pronunciations. Nikkei is a business newspaper, and differs significantly in writing style from the general domain corpus. 10% of the corpus (29k words) was used as testing data, and the other 90% (263k words) was used as training data to be fully or partially annotated.

For full annotation, the training corpus was shuffled in random order, and sentences were chosen from the head of the shuffled corpus. For partial annotation, the following annotation strategy was used:

1. Frequencies of all character bi-grams in the target domain corpus are counted.
2. Bi-grams that occur at least once in the general domain corpus are removed.
3. In descending order of frequency, one instance of each remaining character bi-gram is annotated with word boundaries and pronunciations until a certain set number of bi-grams has been annotated.

This strategy has the advantage of being simple, while still covering the great majority of mis-segmented and unknown words.

4.2 Evaluation Criterion

WS accuracy is measured using point-wise accuracy. For each point between two characters in the corpus, WS judges whether a word boundary exists or not. The WS accuracy is measured by the number of times this judgment is correct, divided by the total number of points between characters in the corpus.

PE is evaluated using mora F-measure. The longest common subsequence (LCS) of mora is found between the correct answer and system output, and given precision (P) and recall (R),

$$P = \frac{|LCS|}{|Sys\text{-}out|} \qquad R = \frac{|LCS|}{|Correct|}.$$

F-measure can be found given the harmonic mean of precision and recall:

$$F = 2 \times P \times R/(P + R).$$

Table 1: Accuracy of the point-wise and tri-gram Models (%).

Training Data	Point-wise		Tri-gram	
	WS	PE	WS	PE
BCCWJ Only	98.82	98.31	98.73	98.16
BCCWJ+NKN	99.41	99.26	99.34	99.15

4.3 Strategies

We tested the performance of three different strategies:

Strategy $\mathcal{S}$: A pair-based tri-gram model using full annotation.

Strategy $\mathcal{F}$: A point-wise estimator using full annotation.

Strategy $\mathcal{P}$: A point-wise estimator using partial annotation.

4.4 Evaluation

We compared the accuracies of the three models over various data sizes. The accuracies using none and all of the target-domain data are shown in Table 1 as a baseline and gold standard respectively.

Figure 4 and Figure 5 show the WS and PE accuracies for various data sizes and strategies. Strategies $\mathcal{S}$ and $\mathcal{F}$ both perform similarly on PE when using fully annotated corpora. Strategy $\mathcal{P}$ performs significantly better than both $\mathcal{S}$ and $\mathcal{F}$, particularly when few words have been annotated.

In the two figures, it seems as though the difference between $\mathcal{P}$ and $\mathcal{F}$ is saturating at higher data sizes. This is because the corpus that we are using in the target domain is relatively small, at 263k words. However, in reality a far greater amount of unannotated target-domain data will be available, and performance will likely continue to increase, even after larger numbers of words are annotated.

5 Conclusion

We presented a strategy for word-based partial annotation that is able to reduce the amount of annotation necessary to achieve increases in accuracy. In order to take advantage of partially annotated

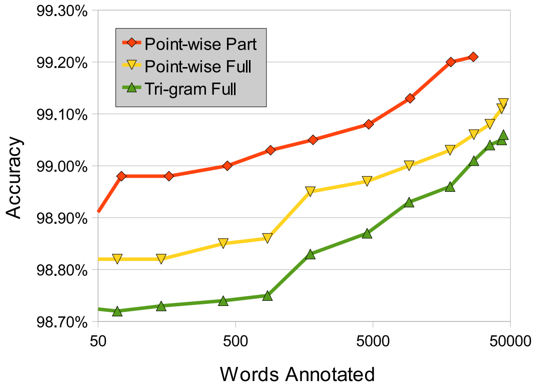

Figure 4: Accuracy of various models on word segmentation.

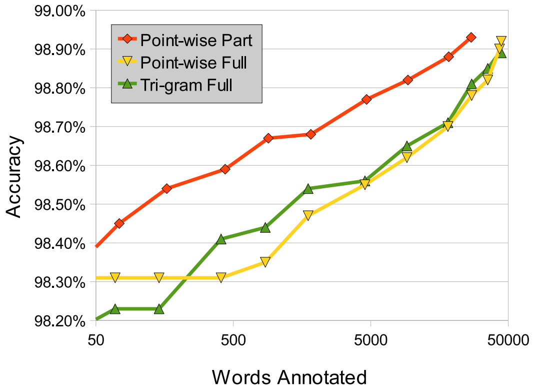

Figure 5: Accuracy of various models on pronunciation estimation.

data, we decomposed the problem of Japanese pronunciation estimation into two parts, and developed a point-wise estimator that can be trained on incomplete annotations. This annotation strategy proved effective in reducing the amount of annotation necessary for equal gains in accuracy. From these results we can conclude that when we prepare language resources for NLP applications that it is important to consider both the annotation strategy and the machine learning techniques, as well as the compatibility between the two.

A promising future research direction in this area is the development of a more intelligent method to select words to annotate, such as active learning. Another possible direction would be the application of this technique to other domains such as Japanese morphological analysis or POS tagging in English or other languages.

References

Yasuharu Den, Junpei Nakamura, Toshinobu Ogiso, and Hideki Ogura. 2008. A proper approach to Japanese morphological analysis: Dictionary, model, and evaluation. In *Proc. LREC2008*.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874.

Taku Kudo. 2009. MeCab: yet another part-of-speech and morphological analyzer. http://mecab.sourceforge.net.

Kikuo Maekawa. 2008. Balanced corpus of contemporary written Japanese. In *Proceedings of the 6th Workshop on Asian Language Resources*, pages 101–102.

Tohru Nagano, Shinsuke Mori, and Masafumi Nishimura. 2005. A stochastic approach to phoneme and accent estimation. In *Proc. InterSpeech2005*, pages 3293–3296.

Eric Ringger, Peter Mcclanahan, Robbie Haertel, George Busby, Marc Carmen, James Carroll, and Deryle Lonsdale. 2007. Active learning for part-of-speech tagging: Accelerating corpus annotation. In *Proc. LAW2007*, pages 101–108.

Yuta Tsuboi, Hisashi Kashima, Hiroki Oda, Shinsuke Mori, and Yuji Matsumoto. 2008. Training conditional random fields using incomplete annotations. In *Proc. COLING08*, pages 897–904.