A Named Entity Recognition Method using Rules Acquired from Unlabeled Data

Tomoya Iwakura
Fujitsu Laboratories Ltd.
1-1, Kamikodanaka 4-chome, Nakahara-ku, Kawasaki 211-8588, Japan
iwakura.tomoya@jp.fujitsu.com

Abstract

We propose a Named Entity (NE) recognition method using rules acquired from unlabeled data. Rules are acquired from automatically labeled data with an NE recognizer. These rules are used to identify NEs, the beginning of NEs, or the end of NEs. The application results of rules are used as features for machine learning based NE recognizers. In addition, we use word information acquired from unlabeled data as in a previous work. The word information includes the candidate NE classes of each word, the candidate NE classes of co-occurring words of each word, and so on. We evaluate our method with IREX data set for Japanese NE recognition and unlabeled data consisting of more than one billion words. The experimental results show that our method using rules and word information achieves the best accuracy on the GENERAL and ARREST tasks of IREX.

1 Introduction

Named Entity (NE) recognition aims to recognize proper nouns and numerical expressions in text, such as names of people, locations, organizations, dates, times, and so on. NE recognition is one of the basic technologies used in text processing such as Information Extraction and Question Answering.

To implement NE recognizers, semi-supervised-based methods have recently been widely applied. These methods use several different types of information obtained from unlabeled data, such as word clusters (Freitag, 2004; Miller et al., 2004), the clusters of multi-word nouns (Kazama and Torisawa, 2008), phrase clusters (Lin and Wu, 2009), hyponymy relations extracted from WikiPedia (Kazama and Torisawa, 2008), NE-related word information (Iwakura, 2010), and the outputs of classifiers or parsers created from unlabeled data (Ando and Zhang, 2005). These previous works have shown that features acquired from large sets of unlabeled data can contribute to improved accuracy. From the results of these previous works, we see that several types of features augmented with unlabeled data contribute to improved accuracy. Therefore, if we can incorporate new features augmented with unlabeled data, we expect more improved accuracy.

We propose a Named Entity recognition method using rules acquired from unlabeled data. Our method uses rules identifying not only whole NEs, but also the beginning of NEs or the end of NEs. Rules are acquired from automatically labeled data with an NE recognizer. The application results of rules are used as features for machine-learning based NE recognitions. Compared with previous works using rules identifying NEs acquired from manually labeled data (Isozaki, 2001), or lists of NEs acquired from unlabeled data (Talukdar et al., 2006), our method uses new features such as identification results of the beginning of NEs and the end of NEs. In addition, we use word information (Iwakura, 2010). The word information includes the candidate NE classes of each word, the candidate NE classes of co-occurring words of each word, and so on. The word information is also acquired from automatically labeled data with an NE recognizer.

We report experimental results with the IREX Japanese NE recognition data set (IREX, 1999). The experimental results show that our method using rules and word information achieves the best accuracy on the GENERAL and ARREST tasks. The experimental results also show that our method contributes to fast improvement of accuracy compared with only using manually labeled

Proceedings of Recent Advances in Natural Language Processing, pages 170–177,
Hissar, Bulgaria, 12-14 September 2011.

Table 1: Basic character types

Hiragana (Japanese syllabary characters), Katakana, Kanji (Chinese letter), Capital alphabet, Lower alphabet, number and Others

training data.

2 Japanese Named Entity Recognition

This section describes our NE recognition method that combines both word-based and character-based NE recognitions.

2.1 Chunk Representation

Each NE consists of one or more words. To recognize NEs, we have to identify word chunks with their NE classes. We use Start/End (SE) representation (Uchimoto et al., 2000) because an SE representation-based NE recognizer shows the best performance among previous works (Sasano and Kurohashi, 2008). SE representation uses five tags which are S, B, I, E and O, for representing chunks. S means that the current word is a chunk consisting of only one word. B means the start of a chunk consisting of more than one word. E means the end of a chunk consisting of more than one word. I means the inside of a chunk consisting of more than two words. O means the outside of any chunk. We use the IREX Japanese NE recognition task for our evaluation. The task is to recognize the eight NE classes. The SE based NE label set for IREX task has (8 × 4) + 1 = 33 labels such as B-PERSON, S-PERSON, and so on.

2.2 Word-based NE Recognition

We classify each word into one of the NE labels defined by the SE representation for recognizing NEs. Japanese has no word boundary marker. To segment words from Japanese texts, we use MeCab 0.98 with ipadic-2.7.0.[1]

Our NE recognizer uses features extracted from the current word, the preceding two words and the two succeeding words (5-word window). The basic features are the word surfaces, the last characters, the base-forms, the readings, the POS tags, and the character types of words within 5-word window size. The base-forms, the readings, and the POS tags are given by MeCab. Base-forms are representative expressions for conjugational words. If the base-form of each word is not equivalent to the word surface, we use the base-form as a feature. If a word consists of only one character, the character type is expressed by using the corresponding character types listed in Table 1. If a word consists of more than one character, the character type is expressed by a combination of the basic character types listed in Table 1, such as Kanji-Hiragana. MeCab uses the set of POS tags having at most four levels of subcategories. We use all the levels of POS tags as POS tag features.

We use outputs of rules to a current word and word information within 5-word window size as features. The rules and the word information are acquired from automatically labeled data with an NE recognizer. We describe rules in section 3. We use the following NE-related labels of words from unlabeled data as word information as in (Iwakura, 2010).

Candidate NE labels: We use NE labels assigned to each word more than or equal to 50 times as candidate NE labels of words.

Candidate co-occurring NE labels: We use NE labels assigned to co-occurring words of each word more than or equal to 50 times as candidate co-occurring NE labels of the word.

Frequency information of candidate NE labels and candidate co-occurring NE labels: These are the frequencies of the NE candidate labels of each word on the automatically labeled data. We categorize the frequencies of these NE-related labels by the frequency of each word n; $50 \leq n \leq 100$, $100 < n \leq 500$, $500 < n \leq 1000$, $1000 < n \leq 5000$, $5000 < n \leq 10000$, $10000 < n \leq 50000$, $50000 < n \leq 100000$, and $100000 < n$.

Ranking of candidate NE labels: This information is the ranking of candidate NE class labels for each word. Each ranking is decided according to the label frequencies.

For example, we obtain the following statistics from automatically labeled data with an NE recognizer for *Tanaka*: S-PERSON was assigned to *Tanaka* 10,000 times, B-PERSON was assigned to *Tanaka* 1,000 times, and I-PERSON was assigned to words appearing next to *Tanaka* 1,000 times. The following NE-related labels are acquired for *Tanaka*: Candidate NE labels are S-PERSON and B-ORGANIZATION. Frequency information of candidate NE labels are $5000 < n \leq 10000$ for S-PERSON, and $500 < n \leq 1000$ for B-ORGANIZATION. The ranking of candidate NE labels are the first for S-PERSON, and second for

[1] http://mecab.sourceforge.net/

B-ORGANIZATION. Candidate co-occurring NE labels at the next word position is I-PERSON. Frequency information of candidate co-occurring NE labels at the next word position is $500 < n \leq 1000$ for I-PERSON.

2.3 Character-based NE Recognition

Japanese NEs sometimes include partial words that form the beginning, the end of NE chunks or whole NEs.[2] To recognize Japanese NEs including partial words, we use a character-unit-chunking-based NE recognition algorithm (Asahara and Matsumoto, 2003; Nakano and Hirai, 2004) following word-based NE recognition as in (Iwakura, 2010).

Our character-based NE recognizer uses features extracted from the current character, the preceding two characters and the two succeeding characters (5-character window). The features extracted from each character within the window size are the followings; the character itself, the character type of the character listed in Table 1, and the NE labels of two preceding recognition results in the direction from the end to the beginning.

In addition, we use words including characters within the window size. The features of the words are the character types, the POS tags, and the NE labels assigned by a word-based NE recognizer.

As for words including characters, we extract features as follows. Let $W(c_i)$ be the word including the i-th character c_i and $P(c_i)$ be the identifier that indicates the position where c_i appears in $W(c_i)$. We combine $W(c_i)$ and $P(c_i)$ to create a feature. $P(c_i)$ is one of the followings: B for a character that is the beginning of a word, I for a character that is in the inside of a word, E for a character that is the end of a word, and S for a character that is a word. [3]

We use the POS tags of words including characters within 5-character window. Let $POS(W(c_i))$ be the POS tag of the word $W(c_i)$ including the i-th character c_i. We express these features with the position identifier $P(c_i)$ like $P(c_i)$-$POS(W(c_i))$. In addition, we use the character types of words including characters. To utilize outputs of a word-based NE recognizer, we use NE labels of words assigned by a word-unit NE recognizer. Each character is classified into one of the 33 NE labels provided by the SE representation.

2.4 Machine Learning Algorithm

We use a boosting-based learner that learns rules consisting of a feature, or rules represented by combinations of features consisting of more than one feature (Iwakura and Okamoto, 2008). The boosting algorithm achieves fast training speed by training a weak-learner that learns several rules from a small portion of candidate rules. Candidate rules are generated from a subset of features called bucket. The parameters for the boosting algorithm are as follows. We used the number of rules to be learned as R=100,000, the bucketing size for splitting features into subsets as $|B|$=1,000, the number of rules learned at each boosting iteration as ν =10, the number of candidate rules used to generate new combinations of features at each rule size as ω=10, and the maximum number of features in rules as ζ=2.

The boosting algorithm operates on binary classification problems. To extend the boosting to multi-class, we used the one-vs-the-rest method. To identify proper tag sequences, we use the Viterbi search. To apply the Viterbi search, we convert the confidence value of each classifier into the range of 0 to 1 with sigmoid function defined as $s(X) = 1/(1 + exp(-\beta X))$, where X is the output of a classifier to an input. We used β=1 in this experiment. Then we select a tag sequence which maximizes the sum of those log values.

To obtain a fast processing and training speed, we apply a technique to control the generation of combinations of features (Iwakura, 2009). This is because fast processing speed is required to obtain word information and rules from large unlabeled data. Using this technique, instead of manually specifying combinations of features to be used, features that are not used in combinations of features are specified as atomic features. The boosting algorithm learns rules consisting of more than one feature from the combinations of features generated from non-atomic features, and rules consisting of only a feature from the atomic and the non-atomic features. We can obtain faster training speed and processing speed because we can reduce the number of combinations of features

[2] For example, Japanese word ”*houbei*” (visit U.S.) does not match with LOCATION ”bei (U.S)”.

[3] If “*Gaimusyouha*”, is segmented as “*Gaimusyou (the Ministry of Foreign Affairs) / ha (particle)*”, then words including characters are follows; *W(Gai) = Gaimusyou*, *W(mu) = Gaimusyou*, *W(syou) = Gaimusyou*, and *W(ha)=ha*. The identifiers that indicate positions where characters appear are follows; *P(Gai) =B*, *P(mu) = I*, *P(syou) = E*, and *P*(*ha*)=S.

to be examined by specifying part of features as atomic. We specify features based on word information and rules acquired from unlabeled data as the atomic features.

3 Rules Acquired from Unlabeled Data

This section describes rules and a method to acquire rules.

3.1 Rule Types

Previous works such as Isozaki (Isozaki, 2001), Talukdar et al., (Talukdar et al., 2006), use rules or lists of NEs for only identifying NEs. In addition to rules identifying NEs, we propose to use rules for identifying the beginning of NEs or the end of NEs to capture context information. To acquire rules, an automatically labeled data with an NE recognizer is used. The following types of rules are acquired.

Word N-gram rules for identifying NEs (*NE-W-rules*, for short): These are word N-grams corresponding to candidate NEs.

Word trigram rules for identifying the beginning of NEs (*NEB-W-rules*): Each rule for identifying the beginning of NEs is represented as a word trigram consisting of the two words preceding the beginning of an NE and the beginning of the NE.

Word trigram rules for identifying the end of NEs (*NEE-W-rules*): Each rule for identifying the end of NEs is represented as a word trigram consisting of the two words succeeding the end of an NE and the end of the NE.

In addition to word N-gram rules, we acquire Word/POS N-gram rules for achieving higher rule coverage. Word/POS N-gram rules are acquired from N-gram rules by replacing some words in N-gram rules with POS tags. We call *NE-W-rules*, *NEB-W-rules* and *NEE-W-rules* converted to Word/POS N-gram rules *NE-WP-rules*, *NEB-WP-rules* and *NEE-WP-rules*, respectively. Word/POS N-gram rules also identify NEs the beginning of NEs and the end of NEs

To acquire Word/POS rules, we replace words having one of the following POS tags with their POS tags as rule constituents: proper noun words, unknown words, and number words. This is because words having these POS tags are usually low frequency words.

3.2 Acquiring Rules

This section describes the method to acquire the rules used in this paper. The rule acquisition consists of three main steps: First, we create automatically labeled data. Second, seed rules are acquired. Finally the outputs of rules are decided.

The first step prepares an automatically labeled data with an NE recognizer. The NE recognizer recognizes NEs from unlabeled data and generates the automatically labeled data by annotating characters recognized as NEs with the NE labels.

The second step acquires seed rules from the automatically labeled data. The following is an automatically labeled sentence.
[Tanaka/\$PN mission/\$N party/\$N $]_{ORG}$ went/ \$V to/\$P [U.K / \$PN$]_{LOC}$ ..." ,
where \$PN (Proper Noun), \$N, \$V, and \$P following / are POS tags, and words between "[and]" were identified as NEs. ORG and LOC after "]" indicate NE types.

The following seed rules are acquired from the above sentence by following the procedures described in previous sections:
NE-W-rules: {*Tanaka mission party* → ORG},
NEB-W-rules: {*went to U.K* → LW=B-LOC},
NEE-W-rules: {*party went to* → FW=E-ORG},
NE-WP-rules: {*\$PN mission party* → ORG},
NEB-WP-rules: {*went to \$PN* → LW=B-LOC},
NEE-WP-rules: {*\$PN mission party* → LW=B-ORG},
where FW, LW, B-LOC, and E-ORG indicate the first words of word sequences that a rule is applied to, the last words of word sequences that a rule is applied to, the beginning word of a LOCATION NE, and the end word of an ORGANIZATION NE, respectively. The left of each → is the *rule condition* to apply a rule, and the right of each → is the seed output of a rule. If the output of a rule is only an NE type, this means the rule identifies an NE. Rules with outputs including = indicate rules for identifying the beginning of NEs or the end of NEs. The left of = indicates the positions of words where the beginning of NEs or the end of NEs exist in the identified word sequences by rules. For example, LW=B-LOC means that LW is B-LOC.

The final step decides the outputs of each rule. We count the outputs of the rule condition of each seed rule, and the final outputs of each rule are decided by using the frequency of each output. We use outputs assigned to each seed rule

more than or equal to 50 times.[4] For example, if LW=B-LOC are obtained 10,000 times, and LW=B-ORG are obtained 1,000 times, as the outputs for $\{$*went to $PN*$\}$, the followings are acquired as final outputs:
LW=B-LOC_RANK1, LW=B-ORG_RANK2, LW=B-LOC_FREQ-5000 $< n \leq$ 10000, and LW=B-ORG_FREQ-500 $< n \leq$ 1000.

The LW=B-LOC_RANK1 and the LW=B-ORG_RANK2 are the ranking of the outputs of rules. LW=B-LOC is 1st ranked output, and LW=B-ORG is 2nd ranked output. Each ranking is decided by the frequency of each output of each rule condition. The most frequent output of each rule is ranked as first.

LW=B-LOC_FREQ-5000 $< n \leq$ 10000 and LW=B-ORG_FREQ-500 $< n \leq$ 1000 are frequency information. To express the frequency of each rule output as binary features, we categorize the frequency of each rule output by the frequency of each rule output n; $50 \leq n \leq 100$, $100 < n \leq 500$, $500 < n \leq 1000$, $1000 < n \leq 5000$, $5000 < n \leq 10000$, $10000 < n \leq 50000$, $50000 < n \leq 100000$, and $100000 < n$.

3.3 Rule Application

We define the rule application by following the method for using phrase clusters in NER (Lin and Wu, 2009). The application of rules is allowed to overlap with or be nested in one another. If a rule is applied at positions b to e, we add the features combined with the outputs of the rule and matching positions to each word; outputs with *B-* (beginning) to b-th word, outputs with *E-* (end) to b-th word, outputs with *I-* (inside) within $b + 1$-th to $e - 1$-th words, outputs with *P-* (previous) to $b - 1$-th word, and outputs with *F-* (following) to $e + 1$-th word.

If a rule having the condition $\{$*went to $PN*$\}$ is applied to {.... Ken/$PN went/$V to/$P Japan/ $PN for/$P ...}, the followings are captured as rule application results: b-th word is went, the word between b-th and e-th is to, e-th word is Japan, $b - 1$-th is Ken, and $e + 1$-th is for.

If the output of the rule is LW=B-LOC, the following features are added: B-LW=B-LOC for went, I-LW=B-LOC for to, E-LW=B-LOC for Japan, P-LW=B-LOC for Ken, and F-LW=B-LOC for for.

3.4 Repeatedly Acquisition

We also apply a method to acquire word information (Iwakura, 2010) to the rule acquisition repeatedly. This is because the previous work reported that better accuracy was obtained by repeating the acquisition of NE-related labels of words. The collection method is as follows.

(1) Create an NE recognizer from training data.

(2) Acquire word information and rules from unlabeled data with the current NE recognizer.

(3) Create a new NE recognizer with the training data, word information and rules acquired at step (2). This NE recognizer is used for acquiring new word information and rules at the next iteration.

(4) Go back to step (2) if the termination criterion is not satisfied. The process (2) to (4) is repeated 4 times in this experiment.

4 Experiments

4.1 Experimental settings

The following data prepared for IREX (IREX, 1999) were used in our experiment. We used the CRL data for the training. CRL data has 18,677 NEs on 1,174 stories from Mainichi Newspaper. In addition, to investigate the effectiveness of unlabeled data and labeled data, we prepared another labeled 7,000 news stories including 143,598 NEs from Mainichi Shinbun between 2007 and 2008 according to IREX definition. We have, in total, 8,174 news stories including 162,859 NEs that are about 8 times of CRL data. To create the additional labeled 7,000 news stories, about 509 hours were required. The average time for creating a labeled news story is 260 seconds, which means only 14 labeled news stories are created in an hour.

For evaluation, we used formal-run data of IREX: GENERAL task including 1,581 NEs, and ARREST task including 389 NEs.

We compared performance of NE recognizers by using the F-measure (FM) defined as follows with Recall (RE) and Precision (PR);
FM = 2 × RE × PR / (RE + PR),
where,
RE = NUM / (the number of correct NEs),
PR = NUM / (the number of NEs extracted by an NE recognizer),

[4] We conducted experiments using word information and rules obtained from training data with different frequency threshold parameters. The parameters are 1, 3, 5, 10, 20, 30, 40, and 50. We select 50 as the threshold because the parameter shows the best result among the results obtained with these parameters on a pilot study.

Table 2: Experimental Results: Each AV. indicates a micro average F-measure obtained with each NE recognizer. B., +W, +R, and +WR indicate the base line recognizer, using word information, using rules, and using word information and rules. Base indicates the base line NE recognizer not using word information and rules.

	B.	+ W	+ R	+WR
GENERAL	85.35	88.04	85.93	**88.43**
ARREST	85.64	89.35	87.39	**91.33**
AV.	85.40	88.56	86.22	89.00

and NUM is the number of NEs correctly identified by an NE recognizer.

The news stories from the Mainichi Shinbun between 1991 and 2008 and Japanese WikiPedia entries of July 13, 2009, were used as unlabeled data for acquiring word information and rules. The total number of words segmented by MeCab from these unlabeled data was 1,161,758,003, more than one billion words. [5]

4.2 Evaluation of Our Proposed Method

We evaluated the effectiveness of the combination of word information and rules. Table 2 shows experimental results obtained with an NE recognizer without any word information and rules (NER-BASE, for short), an NE recognizer using word information (NER-W for short), an NE recognizer using rules (NER-R, for short), and an NE recognizer using word information and rules (NER-WR, for short), which is based on our proposed method

We used word information and rules obtained with the NER-BASE, which was created from CRL data without word information and rules. We see that we obtain better accuracy by using word information and rules acquired from unlabeled data.

The NER-WR shows the best average F-measure (FM). The average FM of the NER-WR is 3.6 points higher than that of the NER-BASE. The average FM of the NER-WR is 0.44 points higher than that of NER-W, and 2.78 points higher than that of the NER-R. These results show that combination of word information and rules contributes to improved accuracy. We also evaluated the effectiveness of the combination of rules for identifying NEs, and rules for identifying beginning of NEs or end of NEs. The micro average FM values for an NE recognizer using rules for identifying NEs, an NE recognizer using rules for identifying beginning of NEs or end of NEs, and the NE recognizer using the both types of rules are 85.77, 84.19 and 86.22. This result shows using the two types of rules are effective.

Table 3: Experimental Results obtained with NE recognizers using word information and rules: G., A., and AV. indicate GENERAL, ARREST, and a micro average obtained with each NE recognizer at each iteration, respectively.

	1	2	3	4	5
G.	85.35	**88.43**	88.22	88.20	88.31
A.	85.64	91.33	91.52	91.49	**92.19**
AV.	85.40	89.00	88.88	88.85	**89.08**

Then we evaluate the effectiveness of the acquisition method described in section 3.4. Table 3 shows the accuracy obtained with each NE recognizer at each iteration. The results at iteration 1 is the results obtained with the base line NE recognizer not using word information and rules. We obtained the best average accuracy at iteration 5. The results obtained with the NE recognizer at iteration 5 shows 4.76 points higher average F-measure than that of the NE recognizer at iteration 1, and 0.37 points higher average F-measure than that of the NE recognizer at iteration 2.

Table 4 shows the results of the previous works using IREX Japanese NE recognition tasks. All the results were obtained with CRL data as manually labeled training data. Our results are F-measure values obtained with the NE recognizer at iteration 5 on Table 3.

We see that our NE recognizer shows the best F-measure values for GENERAL and ARREST. Compared with our method only using unlabeled data, most previous works use handcrafted resources, such as a set of NEs are used in (Uchimoto et al., 2000), and NTT GOI Taikei (Ikehara et al., 1999), which is a handcrafted thesaurus, is used in (Isozaki and Kazawa, 2002; Sasano and Kurohashi, 2008). These results indicate that word information and rules acquired from large unlabeled data are also useful as well as handcrafted resources. In addition, we see that our method with large labeled data show much better perfor-

[5] We used WikiPedia in addition to news stories because Suzuki and Isozaki (Suzuki and Isozaki, 2008) reported that the use of more unlabeled data in their learning algorithm can really lead to further improvements. We treated a successive numbers and alphabets as a word in this experiment.

Table 4: Comparison with previous works. GE and AR indicate GENERAL and ARREST.

	GE	AR
(Uchimoto et al., 2000)	80.17	85.75
(Takemoto et al., 2001)	83.86	-
(Utsuro et al., 2002)	84.07	-
(Isozaki and Kazawa, 2002)	85.77	-
(Sasano and Kurohashi, 2008)	87.72	-
(Iwakura, 2010)	87.34	91.95
This paper	**88.31**	**92.19**

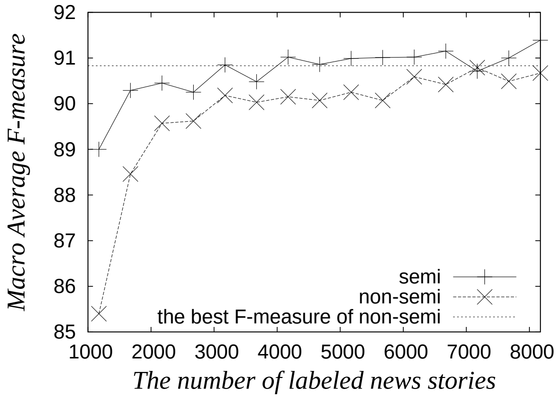

Figure 1: Experimental results obtained with different size of training data. Each point indicates the micro average F-measure of an NE recognizer.

mance than the other methods.

4.3 Evaluating Effectiveness of Our Method

This section describes the performances of NE recognizers trained with larger training data than CRL-data. Figure 1 shows the performance of each NE recognizer trained with different size of labeled training data. The leftmost points are the performance of the NE recognizers trained with CRL data (1,174 news stories). The other points are the performances of NE recognizers trained with training data larger than CRL data. The size of the additional training data is increased by 500 news stories.

We examined NE recognizers using our proposed method (semi), and NE recognizers not using our method (non-semi). In the following, semi-NER indicates NE recognizers using unlabeled data based on our method, and non-semi-NER indicates NE recognizers not using unlabeled data. Figure 1 shows that the semi-NER trained with CRL data shows competitive performance of the non-semi-NER trained with about 1.5 time larger training data consisting of CRL data and additional labeled 500 news stories. To create manually labeled 500 news stories, about 36 hours are required.[6] To achieve the competitive performance of the non-semi-NER trained with CRL data and the labeled 7,000 news stories, semi-NER requires only 2,000 news stories in addition to CRL data. This result shows that our proposed method significantly reduces the number of labeled data to achieve a competitive performance obtained with only using labeled data. Figure 1 also shows that our method contributes to improved accuracy when using the large labeled training data consisting of CRL data and 7,000 news stories. The accuracy is 90.47 for GENERAL, and 94.30 for ARREST. In contrast, when without word information and rules acquired from unlabeled data, the accuracy is 89.43 for GENERAL, and 93.44 for ARREST.

5 Related Work

To augment features, methods for using information obtained with clustering algorithms were proposed. These methods used word clusters (Freitag, 2004; Miller et al., 2004), the clusters of multi-word nouns (Kazama and Torisawa, 2008), or phrase clusters (Lin and Wu, 2009). In contrast, to collect rules, we use an automatically tagged data with an NE recognizer. Therefore, we expect to obtain more target-task-oriented information with our method than that of previous works. Although there are differences between our method and the previous works, our method and previous works are complementary .

To use rules in machine-learning-based NE recognitions, Isozaki proposed a Japanese NE recognition method based on a simple rule generator and decision tree learning. The method generates rules from supervised training data (Isozaki, 2001). Talukdar et al., proposed a method to use lists of NEs acquired from unlabeled data for NE recognition (Talukdar et al., 2006). Starting with a few NE seed examples, the method extends lists of NEs. These methods use rules or lists of NEs for identifying only NEs. Compared with these methods, our method uses rules for identifying the beginning of NEs and the end of NEs in addition

[6] We estimate the hours by using the average labeling time of a news story. The average time is 260 seconds per news story.

to rules identifying whole NEs. Therefore, our methods can use new features not used in previous works.

6 Conclusion

This paper proposed an NE recognition method using rules acquired from unlabeled data. Our method acquires rules for identifying NEs, the beginning of NEs, and the end of NEs from an automatically labeled data with an NE recognizer. In addition, we use word information including the candidate NE classes, and so on. We evaluated our method with IREX data set for Japanese NE recognition and unlabeled data consisting of more than one billion words. The experimental results showed that our method using rules and word information achieved the best accuracy on the GENERAL and ARREST tasks.

References

Rie Ando and Tong Zhang. 2005. A high-performance semi-supervised learning method for text chunking. In *Proc. of ACL 2005*, pages 1–9.

Masayuki Asahara and Yuji Matsumoto. 2003. Japanese named entity extraction with redundant morphological analysis. In *Proc. of HLT-NAACL 2003*, pages 8–15.

Dayne Freitag. 2004. Trained named entity recognition using distributional clusters. In *Proc. of EMNLP 2004*, pages 262–269.

Satoru Ikehara, Masahiro Miyazaki, Satoshi Shirai, Akio Yokoo, Hiromi Nakaiwa, Kentaro Ogura, Yoshifumi Ooyama, and Yoshihiki Hayashi. 1999. *Goi-Taikei -A Japanese Lexicon CDROM*. Iwanami Shoten.

Committee IREX. 1999. *Proc. of the IREX workshop*.

Hideki Isozaki and Hideto Kazawa. 2002. Speeding up named entity recognition based on Support Vector Machines (in Japanese). In *IPSJ SIG notes NL-149-1*, pages 1–8.

Hideki Isozaki. 2001. Japanese named entity recognition based on a simple rule generator and decision tree learning. In *Proc. of ACL 2001*, pages 314–321.

Tomoya Iwakura and Seishi Okamoto. 2008. A fast boosting-based learner for feature-rich tagging and chunking. In *Proc. of CoNLL 2008*, pages 17–24.

Tomoya Iwakura. 2009. Fast boosting-based part-of-speech tagging and text chunking with efficient rule representation for sequential labeling. In *Proc. of RANLP 2009*.

Tomoya Iwakura. 2010. A named entity extraction using word information repeatedly collected from unlabeled data. In *Proc. of CICLing 2010*, pages 212–223.

Jun'ichi Kazama and Kentaro Torisawa. 2008. Inducing gazetteers for named entity recognition by large-scale clustering of dependency relations. In *Proc. of ACL-08: HLT*, pages 407–415.

Dekang Lin and Xiaoyun Wu. 2009. Phrase clustering for discriminative learning. In *Proc. of ACL-IJCNLP 2009*, pages 1030–1038.

Scott Miller, Jethran Guinness, and Alex Zamanian. 2004. Name tagging with word clusters and discriminative training. In *Proc. of HLT-NAACL 2004*, pages 337–342.

Keigo Nakano and Yuzo Hirai. 2004. Japanese named entity extraction with bunsetsu features (in Japanese). In *IPSJ Journal, 45(3)*, pages 934–941.

Ryohei Sasano and Sadao Kurohashi. 2008. Japanese named entity recognition using structural natural language processing. In *Proc. of IJCNLP 2008*, pages 607–612.

Jun Suzuki and Hideki Isozaki. 2008. Semi-supervised sequential labeling and segmentation using giga-word scale unlabeled data. In *Proc. of ACL-08: HLT*, pages 665–673.

Yoshikazu Takemoto, Toshikazu Fukushima, and Hiroshi Yamada. 2001. A Japanese named entity extraction system based on building a large-scale and high quality dictionary and pattern-matching rules (in Japanese). 42(6):1580–1591.

Partha Pratim Talukdar, Thorsten Brants, Mark Liberman, and Fernando Pereira. 2006. A context pattern induction method for named entity extraction. In *Proc. of CoNLL 2006*, pages 141–148.

Kiyotaka Uchimoto, Qing Ma, Masaki Murata, Hiromi Ozaku, Masao Utiyama, and Hitoshi Isahara. 2000. Named entity extraction based on a maximum entropy model and transformati on rules. In *Proc. of the ACL 2000*, pages 326–335.

Takehito Utsuro, Manabu Sassano, and Kiyotaka Uchimoto. 2002. Combining outputs of multiple Japanese named entity chunkers by stacking. In *Proc. of EMNLP 2002*, pages 281–288.