NAK Team's System for Recognition Textual Entailment at the NTCIR-11 RITE-VAL task

Genki Teranaka
Keio University, Japan
teranaka@nak.ics.keio.ac.jp

Masahiko Sunohara
Keio University, Japan
haru@nak.ics.keio.ac.jp

Hiroaki Saito
Keio University, Japan
hxs@nak.ics.keio.ac.jp

ABSTRACT

The NAK team participated in the NTCIR-11 RITE-VAL task. This paper describes our textual entailment system and discusses the official results. Our system adopts statistical method: classification of the support vector machine (SVM). For Japanese SV subtask, our best result was 63.19 for macro-F1 score and 74.55 for accuracy. For Japanese FV subtask, our best result was 53.07 for macro-F1 score and 60.82 for accuracy.

Team Name

NAK

Subtasks

System Validation, Fact Validation (Japanese)

Keywords

Textual Entailment, Support Vector Machine, Word2vec

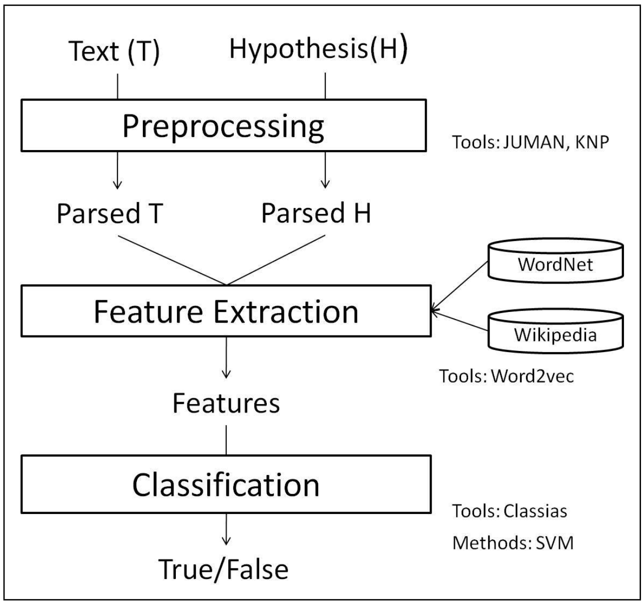

Figure 1: The overview of textual entailment system

1. INTRODUCTION

This paper describes our textual entailment recognition system for Japanese SV and FV subtasks in the NTCIR-11 RITE-VAL task[3] .

Recognition textual entailment (RTE) is focused on as a shared task to understand natural language. When a pair of text T(Text) and H(Hypothesis) is given, RTE task is to determine whether the T entails H or not. For example, following the pair of text is given:
T: Yasunari Kawabata won the Nobel Prize in Literature for his novel “Snow Country”
H: Yasunari Kawabata is the writer of “Snow Country”
Then T entails H if we can judge H is right from T.

One approach for the RTE task is the binary classification of machine learning. Support Vector Machine (SVM)[5] is one of the binary classifiers which can perform most efficiently. In the RTE task using SVM, the classification performance depends on the features extracted heuristically. Thus feature extraction plays an important role.

Word2vec[4] is a tool that can extract the feature as vector representations from words. The skip-gram model implemented in Word2vec is an efficient method for learning distributed vector representations. Distributed representation of words in vector space helps better performance in natural language processing tasks by grouping similar words. It is important for RTE task to consider a semantic similarity of words.

This paper is organized as follows. Section 2 describes our binary classification system. We show and discuss the result of the NTCIR-11 RITE-VAL task in Section 3. In the end, we conclude in Section 4.

2. SYSTEM DESCRIPTION

Our system consists of three steps. First step is preprocessing of morphological analysis and syntactic analysis mainly. Second step is feature extraction, and here we implemented 12 features. Third step is the SVM classification using features extracted by step 2. Figure 1 shows the overview of our system. In this section, we explain description of each step.

2.1 Preprocessing

In this step, we use two tools: JUMAN[1] and KNP[2]. JUMAN is a Japanese morphological analyzer and KNP is a Japanese dependency parser. KNP is implemented not

[1]http://nlp.ist.i.kyoto-u.ac.jp/index.php?JUMAN
[2]http://nlp.ist.i.kyoto-u.ac.jp/index.php?KNP

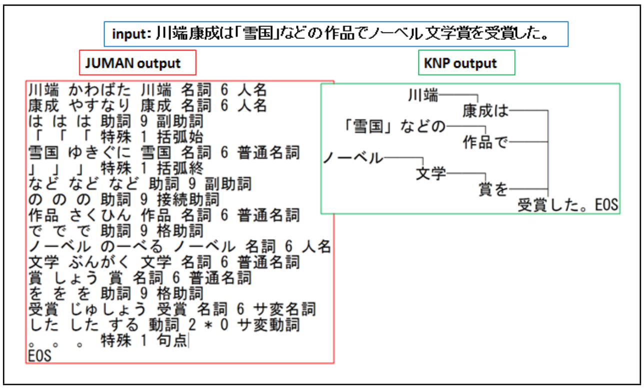

Figure 2: An example of JUMAN and KNP outputs

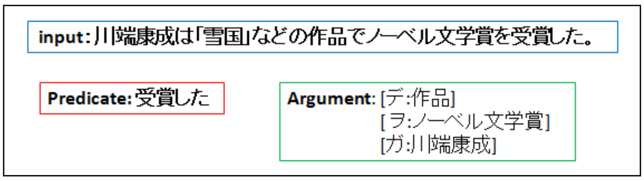

Figure 3: An example of construction predicate-argument structures

only as a dependency parser but also as reference resolution, predicate-argument structure analysis, named entity recognition (NER) and some other functions. Our system adopted the following functions in JUMAN and KNP.

Morphological Analysis and Dependency Parse
We can get the output like Figure 2 from JUMAN and KNP.

Predicate-Argument Structure
A predicate-argument structure identifies semantic relations between predicates and their related arguments[1]. Figure 3 shows an example.

Named Entity Recognition
KNP recognizes eight named entities as follows: ORGNIZATION, PERSON, LOCATION, ARTIFACT, DATE, TIME, MONEY, PERCENT

Subject Expression
We tagged the noun which is the subject of the statement.

Negative Expression
We tagged the negative expression.

Tense
We tagged the predicate with tense: PAST, PRESENT, FUTURE.

Wikipedia Entry
KNP detects some phrase which can be searched in Wikipedia.

2.2 Feature Extraction

In this step, we describe the details of features we implemented. In advance, we defined some notations as the following:

- t_1: T(Text)
- t_2: H(Hypothesis)
- $n(x)$: The number of x, x is a set.
- $|x|$: Length of x, x is a text.

We have implemented 12 features as follow. Token Overlap, Chunk Overlap, 4-gram of Token Overlap, Noun Overlap, and Jaro distance are referred to WSD team's approach for the NTCIR-10 RITE-2 task[1]. We mainly proposed Named Entity Matching, Wikipedia, and Word2vec Distance.

Token Overlap
We split t_1 and t_2 into morphemes in preprocessing. We defined the token overlap score between t_1 and t_2:

$$TO(t_1, t_2) = \frac{n(T_1 \cap T_2)}{n(T_2)} \tag{1}$$

where T_1 is the token set of t_1 and T_2 is the token set of t_2.

Chunk Overlap
We defined the chunk overlap score between t_1 and t_2:

$$CO(t_1, t_2) = \frac{n(C_1 \cap C_2)}{n(C_2)} \tag{2}$$

where C_1 is the chunk set of t_1 and C_2 is the chunk set of t_2.

4-gram of Token Overlap
We defined the 4-gram of token overlap score between t_1 and t_2:

$$\text{4-gram}TO(t_1, t_2) = \frac{n(G_1 \cap G_2)}{n(G_2)} \tag{3}$$

where G_1 is the 4-gram token set of t_1 and G_2 is the 4-gram token set of t_2.

Noun Overlap
We defined the noun overlap score between t_1 and t_2:

$$NO(t_1, t_2) = \frac{n(N_1 \cap N_2)}{n(N_2)} \tag{4}$$

where N_1 is the set of nouns contained in t_1 and N_2 is the set of nouns contained in t_2.

Jaro Distance
The Jaro distance is a measure that considers the number of matching characters in both strings being compared, and also the number of transpositions which is defined as the number of matching characters (in a different sequence order) divided by two[3]. The measure returns a score between 0 and 1. We defined the Jaro distance score between t_1 and t_2:

$$JD(t_1, t_2) = \begin{cases} 0 & \text{if } m = 0 \\ \frac{1}{3}\left(\frac{m}{|t_1|} + \frac{m}{|t_2|} + \frac{m-t}{m}\right) & \text{otherwise} \end{cases} \tag{5}$$

where m is the number of matching t_1 and t_2 characters and t is half the number of transpositions.

[3]http://nbviewer.ipython.org/gist/MajorGressingham/7691723

Modality

If t_1 and t_2 have predicates whose original is same, we extracted m_1 and m_2, modalities of t_1 and t_2. Kawada et al.[2] decided the relation of textual entailment between m_1 and m_2. If m_1 entails m_2, the modality score is 1 and if m_1 doesn't entail m_2, the modality score is -1. Otherwise the modality score is 0.

Named Entity Matching

The tag set of named entity has 8 factors, $\{tag_k|k = 1, 2, ..., 8\}$, and for each tag we can define the named entity matching score between t_1 and t_2 by equation:

$$NE_{tag_k}(t_1, t_2) = \max\{JD(i, j)|i \in n_1^{tag_k}, j \in n_2^{tag_k}\} \quad (6)$$

where $n_1^{tag_k}$ is the set of named entity tagged with tag_k in t_1, and $n_2^{tag_k}$ is the set of named entity tagged with tag_k in t_2.

Tense

The tense of the predicate has the relation of textual entailment defined in Table 1. In Table 1, 0 means t_1 does not entail t_2 and 1 means t_1 may entail t_2.

Table 1: the entailment relation of the tense between t_1 and t_2[2]

$t_1 \setminus t_2$	PRESENT	PAST	FUTURE
PRESENT	1	1	1
PAST	0	1	1
FUTURE	0	0	1

Negative Expression

In preprocessing, we tagged negative expression. If the sentence includes l number of negative expressions, then we defined the negative expression score between t_1 and t_2:

$$NEX(t_1, t_2) = -1^l \quad (7)$$

Synsets, Hypernyms

Here, we used Japanese WordNet. If a noun in t_2 is the synset or hypernym of a noun in t_1, both nouns are regarded overlapping. This score is calculated in the same manner as the noun overlap score in (4).

Wikipedia

Here, we used Wikipedia search. If t_2 includes expressions with Wikipedia entry tag, our system searches Wikipedia for that expression and extracts the definition D. The Wikipedia score was defined:

$$WS(t_1, D) = TO(t_1, D) \quad (8)$$

where TO is defined by (1).

Word2vec Distance

Word2vec[4] is a tool which uses the continuous bag-of-words and skip-gram architectures for computing vector representations of words. It can learn vector representation of each word from some corpora, and can calculate the semantic distance between two words. We trained Word2vec by Wikipedia corpus. A semantic distance of Word2vec is cosine similarity between vectors of two words and takes the value from -1 to 1. If the distance is close to 1, the meaning of the words are semantically close. We defined the Word2vec distance score between t_1 and t_2:

$$s_l = \max\{distance(i, m_l), i \in N_1\}$$

$$WD(t_1, t_2) = \min\{s_k|k = 1, 2, ..., L\} \quad (9)$$

where N_1 is the set of noun in t_1. $N_2 = \{m_k|k = 1, 2, ..., L\}$ where L is the size of N_2. The function $distance(x, y)$ returns the semantic distance in Word2vec between x and y.

[4]http://code.google.com/p/word2vec/

2.3 Classification

Our system adopted support vector machine for binary classification. We used L1-regularized L1-loss function with hinge loss function. As a tool, we used Classias[7] which implements machine learning for classification.

3. TASK RESULTS

In this section, we describe our results of each task and discuss them. Table 2 shows the result of the system validation subtask, and Table 3 shows the result of the fact validation subtask.

Table 2: Results of RITEVAL JA-SV

"System-Run" Name	Macro-F1	Accuracy
RITEVAL-NAK-JA-SV-01	62.02	73.89
RITEVAL-NAK-JA-SV-02	63.19	74.55
RITEVAL-NAK-JA-SV-03	54.14	72.23

Table 3: Results of RITEVAL JA-FV

"System-Run" Name	Macro-F1	Accuracy
RITEVAL-NAK-JA-FV-01	53.07	55.36
RITEVAL-NAK-JA-FV-02	51.12	60.82

3.1 System Validation

RITEVAL-NAK-JA-SV-01 omitted alignment features: Token Overlap, Chunk Overlap, 4-gram of Token Overlap, Noun Overlap, and Jaro Distance. RITEVAL-NAK-JA-SV-02 used all features in Section 2.2 RITEVAL-NAK-JA-SV-03 omitted semantic features without alignment, Modality, Named Entity Matching, Tense, Negative Expression, Synsets, Hypernyms, Word2vec Distance. The best performance of the three is RITEVAL-NAK-JA-SV-02, 63.19 for macro-F1 score and 74.55 for accuracy. RITEVAL-NAK-JA-SV-01 is better than RITEVAL-NAK-JA-SV-02 for macro-F1 and accuracy, it means that alignment features are more effective than semantic features for RTE task at present.

3.2 Fact Validation

RITEVAL-NAK-JA-FV-01 uses all features described in section 2.2 and RITEVAL-NAK-JA-FV-02 omits alignment features. Both systems use hinge loss SVM as a linear classifier. We employed TSUBAKI[8] search results provided by the organizers. This search results includes 5 result sets each query and result set has 5 candidate strings of t1. TSUBAKI gives probability score to each candidate string. We simply used candidate string that has highest score as t1. In Table 3 we show results of the Fact Validation Task. RITEVAL-NAK-JA-FV-02 performed more better than RITEVAL-NAK-JA-FV-02 in terms of accuracy. but RITEVAL-NAK-JA-FV-02 performed better in term of macro-F1.

3.3 Error Analysis

We show some examples of recognition textual entailment by our system from the test dataset.
(1)
T:かつて大日本帝国憲法時代に行なわれたことのある陪審制を日本国憲法下で行なうことは可能か。
H:日本では，陪審制はこれまで実施されたことはない。
—Recognition Result—
Correct Answer: N
RITEVAL-NAK-JA-SV-01: Y
RITEVAL-NAK-JA-SV-02: N
RITEVAL-NAK-JA-SV-03: N

From example (1), RITEVAL-NAK-JA-SV-01 mistook N with Y, because of using alignment features only. To recognize textual entailment such as example (1), we have to consider some semantic relations as follow:
「大日本帝国」≃ 「日本」
「ない」is negative
RITEVAL-NAK-JA-SV-02 and RITEVAL-NAK-JA-SV-03 selected the correct answer because of using semantic features, such as Word2vec Distance and Negative Expression.
(2)
T:なお、常任理事国（５大国）は手続事項と異なり、実質事項には拒否権を行使できる。
H:安保理の常任理事国は，手続事項以外の事項について，拒否権をもっている。
—Recognition Result—
Correct Answer: Y
RITEVAL-NAK-JA-SV-01: Y
RITEVAL-NAK-JA-SV-02: Y
RITEVAL-NAK-JA-SV-03: N

From example (2), RITEVAL-NAK-JA-SV-03 hypothesized incorrectly, because of using semantic features only. Thus, alignment features are also important for recognition textual entailment.
(3)
T:報道は表現の自由に基づく、報道の自由や知る権利に支えられている。
H:表現の自由には，報道機関の報道の自由も含まれる。
—Recognition Result—
Correct Answer: Y
RITEVAL-NAK-JA-SV-01: N
RITEVAL-NAK-JA-SV-02: N
RITEVAL-NAK-JA-SV-03: N

From example (3), all answers of our system selected wrong answers. In recognition entailment, one of the most difficult linguistic phenomena to recognize is scrambling. In example (3), the subject in T is "報道" but the subject in H is "表現の自由". In this case, not only word order but also the predicate and its argument in H differs from that in T whether entailment is true. As a solution to this problem, Natori et al.[6] proposed to construct datasets of scrambling text pairs. Some approach for scrambling is awaited.
(4)
T:宮廷で吟遊詩人が騎士の武勲や恋愛の物語を語り伝えた。
H:西欧世界の宮廷などで，吟遊詩人が武勲詩や騎士の恋愛詩を朗唱した。
—Recognition Result—
Correct Answer: Y
RITEVAL-NAK-JA-SV-01: N
RITEVAL-NAK-JA-SV-02: N
RITEVAL-NAK-JA-SV-03: N

Another one of the most difficult linguistic phenomena to recognize is the replacement of phrases. In example (4), "騎士の武勲や恋愛の物語を語り伝えた" in T replace "武勲詩や騎士の恋愛詩を朗唱した" in H. It's very difficult to determine that both of them have the same meaning. We cannot propose specific solutions to this problem at this point.

3.4 Dataset Analysis

In formal run, we use 'RITE-VAL-JA-test-systemval' as test data and use 'Combination of RITE2-JA-dev-bc and RITE2-JA-testlabel-bc'as the training data. Table 5 shows the precision, recall and F1 score in formal run result of RITEVAL-NAK-JA-SV-02, and it says the recall and F1 score of labled Y is very low. This is caused by imbalanced training data (Y:N = 469:725) from Table 4. In order to improve this, we built new dataset 'Combination of all training data' include 1893 number of data labeled Y and 1895 number of data labeled N. And we performed development run using this dataset.

Table 4: # of positive/negative samples in datasets

Dataset Name	Y	N
RITE-VAL-JA-test-systemval	339	1040
Combination of RITE2-JA-dev-bc and RITE2-JA-testlabel-bc	496	725
Combination of all training data	1893	1895

Table 5: Result of RITEVAL-NAK-JA-SV-02

Label	Precision	Recall	F1
Y	47.81	38.64	42.74
N	81.18	86.25	83.64

Table 6 shows the results of development run of each system and Table 7 shows the precision, recall and F1 score in development run result of RITEVAL-NAK-JA-SV-02. The recall of Y improved by 10.92 point and the F1 score of Y improved by 5.96 point. The macro-F1 score of RITEVAL-NAK-JA-SV-02 is 65.79, improved by 2.60. This is the third best score of all team (the sixth in formal run).

Table 6: Results of development run

"System-Run" Name	Macro-F1	Accuracy
RITEVAL-NAK-JA-SV-01	63.10	72.66
RITEVAL-NAK-JA-SV-02	65.79	74.33
RITEVAL-NAK-JA-SV-03	57.88	69.98

Table 7: Development run result of RITEVAL-NAK-JA-SV-02

Label	Precision	Recall	F1
Y	47.86	49.56	48.70
N	83.37	82.40	82.88

4. CONCLUSIONS

In this paper, we introduced the details of NAK team approaches for NTCIR-11 RITE-VAL task, and discussed the run results of our textual entailment system. The best result of our system was 63.19 for macro-F1 score and 74.55 for accuracy at the SV subtask, and 53.07 for macro-F1 score and 60.82 for accuracy at the FV subtask. We performed development run by improving training datasets at the SV task, and achieved 65.79 for macro-F1 score improved by 2.60. This score is the third best score of all teams that participated in Japanese SV subtask.

To make higher performance in RTE, we need to investigate better semantic approaches for RTE.

5. REFERENCES

[1] D. Ito, M. Tanaka, and H. Yamana. WSD Team's Approaches for Textual Entailment Recognition at the NTCIR10 (RITE2). In *Proceedings of the 10th NTCIR Conference*, pages 449–456, 2013.

[2] T. Kawada, K. Julien, and K. Torisawa. Generation of Entailment Pattern Pairs in Consideration of the Tense-Modality. In *Proceedings of the 20th Conference of the Association for Natural Language*, pages 562–565, 2014.

[3] S. Matsuyoshi, Y. Miyao, T. Shibata, C.-J. Lin, C.-W. Shih, Y. Watanabe, and T. Mitamura. Overview of the NTCIR-11 Recognizing Inference in TExt and Validation (RITE-VAL) Task. In *Proceedings of the 11th NTCIR Conference*, 2014.

[4] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed Representations of Words and Phrases and their Compositionality. *Clinical Orthopaedics and Related Research*, abs/1310.4546, 2013.

[5] N. Cristianini and J. Showe-Taylor. *An Introduction to Support Vector Machines.* The Press Syndicate of the University of Cambridge, 2000.

[6] F. Natori, S. Matsuyoshi, and F. Fukumoto. Construction of Scrambling Datasets on Recognition Textual Entailment Tasks. In *Proceedings of the 20th Conference of the Association for Natural Language*, pages 745–748, 2014.

[7] N. Okazaki. Classias: a collection of machine-learning algorithms for classification, 2009.

[8] K. Shinzato, T. Shibata, D. Kawahara, and S. Kurohashi. TSUBAKI: An Open Search Engine Infrastructure for Developing Information Access Methodology. *Journal of Information Processing*, 20(1):216–227, 2012.