FLL: Local Alignments based Approach for NTCIR-10 RITE-2

Takuya Makino, Seiji Okajima, and Tomoya Iwakura

FUJITSU LABORATORIES LTD.
4-1-1 Kamikodanaka, Nakahara-ku, Kawasaki, Japan
{makino.takuya, okajima.seiji, iwakura.tomoya}@jp.fujitsu.com

ABSTRACT

This paper describes the textual entailment system of FLL for RITE-2 task in NTCIR-10. Our system is based on a set of local alignments conducted on different linguistic units, such as word, Japanese base phrase, numerical expression, Named Entity, and sentence. Our system uses features obtained from local alignments' results. We applied our system to Japanese BC task and Japanese MC task at formal run, and Japanese UnitTest task at unofficial run. The performance of our system for BC task and MC task outperformed baseline, and the result of UnitTest achieved the best performance.

Team Name

FLL

Subtasks

Japanse BC, Japanese MC and Japanese UnitTest

Keywords

NTCIR, RITE, machine learning, local alignment

1. INTRODUCTION

Recognizing textual entailment is, given two texts t_1 and t_2, to recognize whether t_1 entails t_2 [4], or recognizing one or more relations between given two texts [16]. Recognizing textual entailment is a common task across many natural language processing applications, such as Question Answering, Multi-Document Summarization and Information Retrieval.

This paper describes our textual entailment system for NTCIR-10 RITE2 task. To recognize relations between two texts, we developed a system that uses local alignments' results as features. Our system first parses given Japanese texts for obtaining linguistic units, such as words, Japanese base phrases, numerical expressions, Named Entity (NE), dependency tree, and predicate-argument structure. Then, relation recognition results on each linguistic unit are used to generate features to recognize a relation between the two texts. We applied our system to Japanese BC task and Japanese MC task at formal run, and Japanese UnitTest task at unofficial run. The performance of our system for the BC and the MC outperformed baseline, and the result of the UnitTest achieved the best performance.

2. BASIC ANALYZERS AND RESOURCES

This section first describes the basic analysis results of texts and the resources used in our system.

2.1 Basic Language Analyzers

Let t_1 and t_2 be a given pair of texts. To recognize a relation between t_1 and t_2, we recognize the following information.

We first recognize words from t_i for $i \in \{1, 2\}$. We denote the word sequence for t_i as $W_i = \langle w_{i(1)}...w_{i(|W_i|)}\rangle$ where $w_{i(j)}$ is the j-th word of t_i, and $|W_i|$ is the number of words in t_i. To recognize words, MeCab[1] was used.

Then, from each text t_i, we recognize numerical expressions $N_i = \{n_{i(1)}, ..n_{i(|N_i|)}\}$, NEs $NE_i = \{ne_{i(1)}, ..ne_{i(|NE_i|)}\}$, and Japanese base phrase called bunsetsu $B_i = \{b_{i(1)}, ..b_{i(|B_i|)}\}$, where $|N_i|$, $|NE_i|$, and $|B_i|$ are the number of recognized numerical expressions, the number of recognized NEs, and the number of recognized bunsetsu for t_i respectively. $n_{i(j)}$ is a numerical expression. Numerical expressions were recognized with normalizeNumexp.[2] $ne_{i(j)}$ is an NE along with the NE type. Our NE recognizer [10] was used to recognize NEs. $b_{i(j)}$ is a bunsetsu that consists of one or more words. To recognize bunsetsu, we used CaboCha [11].

CaboCha was also used to recognize the dependency relation between each pair of bunsetsu in B_i. The set of dependency relations for t_i is denoted as E_i. $e_{i(j,k)}$ in E_i indicates the dependency relation between a modifier $b_{i(j)}$ and $b_{i(j)}$ $(1 \le j < k \le |B_i|)$. If $b_{i(j)}$ modifies $b_{i(k)}$, $e_{i(j,k)}$ includes the bunsetsu pair of $b_{i(j)}$ and $b_{i(k)}$ otherwise no bunsetsu pair is included.

Predicate-argument structure (PAS) relations were also used. Let $p_{i(j,k)}$ be the PAS relation between $b_{i(j)} \in B_i$ and $b_{i(j)} \in B_i$ $(1 \le j < k \le |B_i|)$. If there exists a PAS relation between $b_{i(j)}$ and $b_{i(k)}$, $p_{i(j,k)}$ retains the relation type. If not, $p_{i(j,k)}$ has no value. SynCha [7] was used to obtain predicate-argument structure. Words that POS tags are verb, adjective or noun are predicates, and case components whose case makers are *ga* (nominative), *wo* (accusative), or *ni* (dative) are an argument in SynCha. SynCha originally returns the predicate-argument relation between words, however, to utilize bunsetsu information that words belong to, we retain predicate-argument information at bunsetsu units. If there is a relation between a word in $b_{i(j)}$ and a word $b_{i(k)}$, $p_{i(j,k)}$ retains the relation of the two

[1]http://mecab.googlecode.com/svn/trunk/mecab/doc/index.html
[2]http://www.cl.ecei.tohoku.ac.jp/~katsuma/software/normalizeNumexp/

words as the relation of $b_{i(j)}$ and $b_{i(k)}$.

As sentence level, locations mentioned by a given text was also annotated. We used our location identifier that identifies 47 prefectures in Japan that a give text mentions, and the set of locations mentioned by t_i is denoted as L_i.

2.2 Resources

We used the antonym dictionary in the verb entailment database [6] provided by the ALAGIN forum to calculate the semantic similarity of the verb pair. In addition, we used WordNet [12, 9] to extract hypernym and synonym which are used to calculate the semantic similarity of the content word pair and the chunk pair. We also prepared an in-house Japanese antonym set that was created by translating antonym pairs in English WordNet into Japanese. To translate the pairs, we used the Japanese-English dictionary of Eijiro.[3]

3. FEATURES

This section describes features used in our system.

3.1 Surface Features

The following surface based similarities between t_1 and t_2 are used.

- *Cosine similarity of Words*: Let CW_1 be the set of content words in t_1 and CW_2 be the set of content words in t_2. The value of cosine measure is defined as $|\mathrm{CW}_1 \cap \mathrm{CW}_2|/|\mathrm{CW}_1||\mathrm{CW}_2|$, where $|CW_1|$, $|CW_2|$, and $|CW_1 \cap CW_2|$ are the number of content words in the sets.

- *Cosine similarity of Characters*: In addition to word-based one, we also calculate character-based cosine similarity.

- *Jaccard coefficient*: Let CW_1 be the set of content words in t_1 and CW_2 be the set of content words in t_2, respectively. The value of Jaccard coefficient is defined as $|\mathrm{CW}_1 \cap \mathrm{CW}_2|/|\mathrm{CW}_1 \cup \mathrm{CW}_2|$, where $|CW_1 \cap CW_2|$ is the number of content words that occur both in t_1 and t_2, and $|CW_1 \cup CW_2|$ is the number of content words that occur in t_1 or $t2$,

- *Longest common subsequence*: Longest common subsequence is a common subsequence of string t_1 and t_2 of maximum length. We used the length of the longest common subsequence that is normalized by the length of t_2.

3.2 Bunsetsu Entailment-based Features

In order to recognize textual inference between sentences, we used bunsetsu entailment features. Bunsetsu entailment features are defined based on the entailment types of bunsetsu. The entailment types between a bunsetsu b_1 in t_1 and a bunsetsu b_2 in t_2 are defined as follows.

B: Bidirectional entailment
: b_1 entails b_2 AND b_2 entails b_1.

F: Forward entailment
: b_1 entails b_2 AND b_2 does not entail b_1.

R: Backward entailment
: b_2 entails b_1 AND b_1 does not entail b_2.

C: Contradiction
: b_1 and b_2 contradicts, or cannot be true at the same time.

I: Independence
: Otherwise.

Let $REL(b_1, b_2)$ be the relation between a bunsetsu b_1 in t_1 and b_2 in t_2. The value of $REL(b_1, b_2)$ is one of B, F, R, C and I. Our bunsetsu relation recognizer recognizes a relation $REL(b_1, b_2)$ for each pair of $b_1 \in B_1$ and $b_2 \in B_2$ with the following features.

- Surface features: the length of b_1, the edit distance of b_1 and b_2 and the length of longest common subsequence of b_1 and b_2. These features are normalized by divided by the length of b_2.

- Relations between a word w_1 in b_1 and a word w_2 in b_2: The relations are synonym, hyponym, hypernym and antonym that obtained from Japanese WordNet, ALAGIN and the antonym set of English WordNet.

We employ a supervised learning approach to predict a relation of given two bunsetsu. Labeled bunsetsu pairs for training were manually made from text pairs in the RITE1 BC task and RITE1 MC task and the classifier was trained with AROW [3]. We used CaboCha to recognize bunsetsu.

Finally, we used the following features derived from the relations of bunsetsu.

- The proportion of B, F, R, C, and I between two sentences. The value of each label is the number of the label assigned to bunsetsu divided by the total number of the bunsetsu pairs. This feature is used to capture an alighment of a bunsetsu in t_1 and a bunsetsu in t_2.

- The proportion of each entailment label B, F, R, and C of $REL({b'}_1, {b'}_2)$, where the ${b'}_1$ is modified by b_1 and the ${b'}_2$ is modified by b_2. We generate these features from pairs of bunsetsu (b_1, b_2) that labels are B. This feature is used to capture an alighment of (b_1, b'_1) and (b_2, b'_2).

3.3 Numerical Expression-based Features

We used the relations between numerical expressions in t_1 and t_2 as one of features. Numerical expressions, such as temporal expressions and quantitative expressions, are extracted with *normalizeNumexp*. *normalizeNumexp* also extracts the range of time or quantity values of the expressions. The features of numerical expressions are as follows.

- Whether all the numerical expressions N_2 of t_2 are exactly included in the numerical expressions N_1 in t_1. If there exist numerical expressions that have ranges, the ranges should be the same for this feature.

- Whether all the numerical expressions in N_2 are partially included in N_1. This feature is used when some values in N_2 are included in the ranges that numerical expressions in N_1 and the numerical expressions expressed by values in N_1 are exactly included in N_1.

[3]http://www.alc.co.jp/

- Whether all the numerical expressions N_1 are included in N_2
- Whether there exist one or more numerical expression in N_2 that do not match with the numerical expressions in N_1.

These features are only defined when both t_1 and t_2 have numerical expressions.

3.4 ILP-based Alignment Features

This section describes our approach for an unsupervised textual alignment. We assume that a pair of a text t_1 and a hypothesis t_2 that t_1 entails t_2 (entailed pair) has better local alignments than the other pairs in which the text of each of the other pairs does not entail the hypothesis (non-entailed pair).

In this paper, a local alignment in t_1 and t_2 means a content word alignment, a bunsetsu alignment, or an edge alignment. The goodness of the alignment of t_1 and t_2 is defined as the sum of the scores of local alignments. For local alignments, we define the score of the alignment between two words w and w', two bunsetsu b and b', and two edges e and e' as $s_{ww'}$, $s_{bb'}$ and $s_{ee'}$ respectively. An edge e means a pair of bunsetsu b_m and b_h that b_m modifies b_h.

However, these local alignments have some constraints. For example, in order to choose a bunsetsu alignment between two bunsetsu, the alignment of the words in the one of the two bunsetsu must be chosen from the words of the rest of the two bunsetsu. This is because a content word is a part of a bunsetsu.

Inspired by [14], to select the local alignments that maximize the scores, we solved this problem with an Integer Linear Programming (ILP) solver.

Our formalization is as follows:

$$\begin{aligned}
&\max. \\
&\quad \sum_{w\in W_1, w'\in W_2} s_{ww'}a_{ww'} + \sum_{b\in B_1, b'\in B_2} s_{bb'}a_{bb'} \\
&\quad + \sum_{e\in E_1, e'\in E_2} s_{ee'}a_{ee'}, \qquad (1)\\
&\text{s.t.} \\
&\quad \forall w' \in W_2 \sum_{w\in W_1} a_{ww'} \geq 1; \forall b' \in B_2 \sum_{b\in B_1} a_{bb'} \geq 1; \\
&\quad \forall e' \in E_2 \sum_{e\in E_1} a_{ee'} \geq 1; \\
&\quad \forall b \in B_1, \forall b' \in B_2 \qquad \sum_{w\in W(b), w'\in W(b')} a_{ww'} - a_{bb'} \geq 0; \\
&\quad \forall e \in E_1, \forall e' \in E_2, \{b_m b_h\} \in e, \{b'_m b'_h\} \in e' \\
&\quad a_{b_m b'_m} + a_{b_h b'_h} - a_{ee'} \geq 0; \\
&\quad \forall w \in W_1, \forall w' \in W_2 \quad a_{ww'} \in \{0,1\}; \\
&\quad \forall b \in B_1, \forall b' \in B_2 \quad a_{bb'} \in \{0,1\}; \\
&\quad \forall e \in E_1, \forall e' \in E_2 \quad a_{ee'} \in \{0,1\}.
\end{aligned}$$

Our model is going to maximize Equation (1) under some constraints. Here, let $a_{ww'}$ denote 1 if our model choose alignment of w and w', otherwise 0. W_1 denotes the set of the content words in t_1 and W_2 denotes the set of the content words in t_2. $s_{ww'}$ denotes the score of the alignment of w and w'. $a_{bb'}$ denotes 1 if our model chooses the alignment of b and b', otherwise 0. B_1 denotes the set of the bunsetsu in t_1, B_2 denote the set of the bunsetsu in t_2. $a_{ee'}$ denotes 1 if our model chooses the alignment of e and e', otherwise 0. E_1 denotes the set of the dependency relations in t_1 and E_2 denotes the set of the dependency relations in t_2.

All content words, bunsetsu and dependency relations in t_1 must be aligned with at least a content word, a bunsetsu and a dependency in t_2. We regard the bunsetsu alignment alignment $a_{bb'}$ can be selected when at least one alignment between content words in these bunsetsu b and b' is selected. $W(\cdot)$ is the function that returns the set of content words. In addition, we regard the edge alignment alignment $a_{ee'}$ is selected when alignment between modifier bunsetsu $a_{b_m b'_m}$ or alignment between head bunsetsu $a_{b_h b'_h}$ is selected.

We defined scores of alignments $s_{ww'}$, $s_{bb'}$ and $s_{ee'}$ below. The words, numerical expressions, bunsetsu, dependency relations, and predicate-argument structure of given texts are recognized as described in section 2.1.

- *The score of a word alignment* ($s_{ww'}$): The following four similarities are used for word alignments: edit distance similarity, the longest common subsequence between two words, a Japanese WordNet-based similarity with hypernym relations, and an English WordNet-based similarity with antonym relations. The Japanese WordNet-based similarity is the distance from least common hypernym of two words to the root and normalized by the maximum value of the Japanese WordNet-based similarity. To measure the English WordNet-based similarity, we translated Japanese given two words into their corresponding English words with a Japanese-English dictionary. Then, if the translations of the given words have antonym relation in the English WordNet, the value is set to -1. We regarded the score of the word alignment $s_{ww'}$ as the sum of these similarities divided by the number of measures.

- *The score of a bunsetsu alignment* ($s_{bb'}$): For given two bunsetsu b and b', we used the Jaccard coefficient, the edit distance similarity, the longest common subsequence, and the edit distance similarity between the bunsetsu modified by given bunsetsu. If the both b and b' are labeled as predicate on PAS, we used the edit distance similarity of the arguments of the predicates of b and b'. We also used a WordNet-based similarity and numerical expressions. The WordNet-based similarity is the average score of the words in these bunsetsu. The score of each word is measured with Japanese WordNet as in the score of a word alignment. If the both bunsetsu have numerical expressions, we determine the relation of these two numerical expressions based on handcrafted rules, like the range of a numerical expression of b includes the value of a numerical expression of b'. We also used the output of a bunsetsu level entailment relation recognition analyzer described in section 3.2. If the relation of a two bunsetsu is reverse entailment or contradiction, the score of the bunsetsu entailment relation is -1. If the relation is independent, the score 0, otherwise 1. The score of each bunsetsu alignment $s_{bb'}$ is the sum of these measures divided by the number of measures.

- *The score of an edge alignment* ($s_{ee'}$): We defined the score of the edge alignment $s_{ee'}$ between two edges e

and e' as follows:
$s_{ee'} = 2 \cdot s_{b_m b'_m} \cdot s_{b_h b'_h} / \left(s_{b_m b'_m} + s_{b_h b'_h}\right)$,
where $\{b_m, b_h\} \in e$ and $\{b'_m, b'_h\} \in e'$. The scores are measured based $s_{b_m b'_m}$ and $s_{b_h b'_h}$ that are the score of bunsetsu alignments.

3.5 Location Features

We used the feature of locations that are mentioned by t_1 and t_2. This feature assumes that if t_1 entails t_2, the locations mentioned by t_1 and t_2 are the same. To estimate locations of text, we used a location identifier that identifies 47 prefectures in Japan based on bag-of-words of the text. A location feature is whether each location mentioned by t_2 are also mentioned by t_1 or not.

3.6 Named Entity Features

If there exist NEs in t_2 that are not included in the NEs in t_1, it is an evidence that t_1 does not entail t_2. Therefore, we used features that indicate whether NEs in t_2 are included or not in t_1. To recognize NEs, we used an NE recognizer [10] that recognizes NEs defined by IREX [8]. Among the outputs of the recognizer, PERSON, LOCATION, ORGANIZATION, and ARTIFACT were used for generating the following features.

- Whether all the NEs NE_2 in t_2 are included in the NEs NE_1 in t_1 or not.
- Whether there exists an NE in NE_2 at least that is not included in NE_1. We checked this condition for each NE type.
- The cosine similarity between NE_1 and NE_2.

3.7 Latent Topics Features

If t_1 and t_2 indicate the same topics, it is an evidence that t_1 entails t_2. To identify topics of sentences, we used Latent Dirichlet Allocation (LDA) [1] based on Gibbs sampling [5]. The following features were used.

- Whether the topic that has the highest probability for t_1 is equivalent to that of t_2.
- The cosine similarity between topics of t_1 and that of t_2. We used topics that have probabilities more than the default probability of each topic.[4]

To identify topics of t_1 and t_2, we used a model trained on the sentences in the first paragraph of each news article of Mainichi Shimbun 2001 to 2005 in advance. The features are words except auxiliary verb, postposition, and attached words for verb or adjective. We decided the number of latent topics as 500 because the number of topics showed the best performance on the development data of JA BC task.

4. SYSTEM DEVELOPMENT

Each of three members developed a system with the features described in section 3. Each system was trained with libSVM [2] using RBF kernel. In total, we have developed the following three base systems.

- S1: ILP-based Features, Surface Features except word-based and character-based cosine similarity
- S2: Bunsetsu Alignment Features, Numerical-Expression Features, Location Features, Named Entity Features, Surface Features except word-based character-based cosine similarity
- S3: Latent Topics Features, Named Entity Features, and Surface Features of character-based cosine similarity.

Then, we examined all the combinations of systems. We assume features selected by some members would be important. Therefore, if systems used the same features, we used them as different features given by different systems. When systems were combined, we first merged features that used in the systems, and trained a model with the merged features. Systems with the combinations of the base systems were also trained with libSVM and some additional features. The model of each task is trained from the development data of the task. We selected the soft margin parameter for each task that showed the best accuracy of 10 fold cross-validation on the development of the task.

The following the submitted systems.

- BC
  - FLL-JA-BC-01: S1 + S3
  - FLL-JA-BC-02: S2 + S3
  - FLL-JA-BC-03: S1 + S2 + S3
  - FLL-JA-BC-04: S3
  - FLL-JA-BC-05: S2
  - FLL-JA-BC-06: S2 + S3 + the cosine similarity of words
- MC
  - FLL-JA-MC-01: S2 + S3
  - FLL-JA-MC-02: S2 + S3 + the cosine similarity of words
  - FLL-JA-MC-03: S3 + the cosine similarity of words
  - FLL-JA-MC-04: S2
- UnitTest
  - FLL-JA-UnitTest-01: S2 + S3 + the cosine similarity of words
  - FLL-JA-UnitTest-02: S2 + the cosine similarity of words
  - FLL-JA-UnitTest-03: S3 + the cosine similarity of words

5. RESULTS

The results of our system for BC task, MC task and UnitTest are shown in Table 1, Talbe 2 and Table 3. Each system name with FLL in the tables indicates one of our systems, and † indicates that the results were submitted at unofficial run. The number in parentheses after each system name means the ranking if the system is in top three for each task. Our best system for each task outperformed the baseline, and the best system for MC task showed a high accuracy. On UnitTest, our system showed the best accuracy.

[4]We used the fixed hyper-parameters: $\alpha = 0.1$ and $\beta = 0.01$.

System	Macro F1	Accuracy
DCUMT-JA-BC-01 (1st)	80.49	81.64
WSD-JA-BC-03 (2nd)	80.08	80.66
SKL-JA-BC-02 (3rd)	79.46	79.84
FLL-JA-BC-03	67.99	70.00
FLL-JA-BC-01	63.06	68.36
FLL-JA-BC-05†	61.05	63.28
baseline	62.53	63.93
FLL-JA-BC-02	59.73	64.10
FLL-JA-BC-06†	55.69	57.70
FLL-JA-BC-04†	52.58	55.08

Table 1: The results of our runs for BC task

System	Macro F1	Accuracy
SKL-JA-MC-01 (1st)	59.96	69.53
SKL-JA-MC-02 (2nd)	58.25	68.61
SKL-JA-MC-03 (3rd)	55.45	68.07
FLL-JA-MC-01	53.67	64.96
FLL-JA-MC-04†	51.27	64.23
FLL-JA-MC-02†	35.12	44.71
baseline	26.61	45.44
FLL-JA-MC-03†	22.47	34.49

Table 2: The results of our runs for MC task

System	Macro F1	Accuracy
FLL-JA-UnitTest-01 (1st) †	77.77	90.87
FLL-JA-UnitTest-03 (2nd) †	76.98	91.29
JAIST-JA-UnitTest-02 (3rd)	74.52	89.21
baseline	51.70	86.31
FLL-JA-UnitTest-02†	51.35	77.59

Table 3: The results of our runs for UnitTest task

The UnitTest data set includes several sentence pairs are created for each sample so that only one linguistic phenomenon appears in each pair. Compared with the other systems that participated in UnitTest, our system showed higher precision for text pairs categorized as synonym:phrase and entailment:phrase [16].

We think one of the reasons is features. For example, the best system of RITE1 JA BC task [13] used translation results of given sentences for realizing matching of different structures and words via translation results. The best system of RITE2 JA MC task [15] used tree edit distance, word overlap rations, dictionary-based matching, and so on. Compared with these RITE1 systems, our system introduced the following features for aiming at capturing local differences: bunsetsu alignment features, ILP-based features, Named Entity-based features, location estimation-based features, and features-based on the topics of sentences.

To examine the effectiveness of each feature, we measured the accuracy of the best system of UnitTest obtained by removing each feature. The influential features were character similarity, word similarity, Named Entity (NE) and bunsetsu relation based ones. We think features based on character similarity and word similarity, such as longest common subsequence, levenshtein distance, and word similarity, worked well because each pair of texts in the UnitTest data set includes only one different linguistic phenomenon. On the other hand, NE and bunsetsu relation based ones captured differences of semantics. For example, NE-based features captured the differences of NEs between two sentences such as the first sentence does not include a location NE that is included in the second sentence. The bunsetsu relation based ones captured linguistic units that have same meaning but different surface expressions. For example, bunsetsu relation recognizer aligned expressions such as a partial address expression like “Hyogo Prefecture” and “Ibo Gun, Hyogo Prefecture”, a verb expression like “first introduce” and “initiate”, a noun synonym like “a popular name” and “colloquial term”, and a partial list expression like “such as jungle gym, swing, climbing bar and soccer goal ” and “gym and swing”. Therefore, we think these features contributed to the high accuracy on the UnitTest. Features that showed adverse effect were latent topic features. Most of text pairs in UnitTest have the entailment relation. Such text pairs should have the same latent topics. However, LDA-based latent topic estimation often assigned different latent topics to each text in a pair.

6. CONCLUSION

This paper has described the textual entailment system of FLL for RITE-2 task in NTCIR-10. Our system is based on the set of local alignments conducted on different linguistic unit levels, such as word, Japanese base phrase, numerical expression, Named Entity, and sentence. Our system used features obtained from local alignments' results. The performance of our system for the BC and the MC outperformed baseline, and the result of the UnitTest achieved the best performance.

7. REFERENCES

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[2] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):27:1–27:27, May 2011.

[3] K. Crammer, A. Kulesza, and M. Dredze. Adaptive regularization of weight vectors. In *NIPS*, pages 414–422, 2009.

[4] I. Dagan and O. Glickman. The pascal recognising textual entailment challenge. In *Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment*, 2005.

[5] T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences of the United States of America*, 101(Suppl 1):5228–5235, 2004.

[6] C. Hashimoto, K. Torisawa, D. S. Stijn, J. Kazama, and S. Kurohashi. Extracting paraphrases from definition sentences on the web. In *Proc. of ACL'11*, pages 1087–1097, 2011.

[7] R. Iida and M. Poesio. A cross-lingual ilp solution to zero anaphora resolution. In *Proc. of ACL'11*, pages 804–813, 2011.

[8] IREX Committee. *Proc. of the IREX workshop*. 1999.

[9] H. Isahara, F. Bond, K. Uchimoto, M. Utiyama, and K. Kanzaki. Development of the japanese wordnet. In

Proc. of LREC'08, 2008.

[10] T. Iwakura. A named entity recognition method using rules acquired from unlabeled data. In *Proc. of RANLP'11*, pages 170–177, 2011.

[11] T. Kudo and Y. Matsumoto. Japanese dependency analysis using cascaded chunking. In *Proc. of CoNLL'02*, pages 1–7, 2002.

[12] G. A. Miller. Wordnet: A lexical database for english. *Communications of the ACM*, 38:39–41, 1995.

[13] L. M. N. Quang Nhat Minh Pham and A. Shimazu. A machine learning based textual entailment recognition system of jaist team for ntcir9 rite. In *NTCIR RITE1*, 2011.

[14] M. Sammons, V. Vydiswaran, T. Vieira, N. Johri, M.-W. Chang, D. Goldwasser, V. Srikumar, G. Kundu, Y. Tu, K. Small, J. Rule, Q. Do, and D. Roth. Relation alignment for textual entailment recognition. In *Text Analysis Conference (TAC)*, 2009.

[15] Y. Tsuboi, H. Kanayama, M. Ohno, and Y. Unno. Syntactic difference based approach for ntcir-9 rite task. In *NTCIR RITE1*, 2011.

[16] Y. Watanabe, Y. Miyao, J. Mizuno, T. Shibata, H. Kanayama, C.-W. Lee, C.-J. Lin, S. Shi, T. Mitamura, N. Kando, H. Shima, and K. Takeda. Overview of the Recognizing Inference in Text (RITE-2) at NTCIR-10. In *Proceedings of the 10th NTCIR Conference*, 2013.