TKDDI group at NTCIR10-RITE2: Recognizing Texual Entailment Based on Dependency Structure Alignment

Junta Mizuno Kentaro Inui
Tohoku University, Japan
{junta-m, inui}@ecei.tohoku.ac.jp

Asuka Sumida Gen Hattori Chihiro Ono
KDDI R&D Laboratories, Inc., Japan
{as-sumida, gen, ono}@kddilabs.jp

ABSTRACT

This paper discribes the TKDDI system which participated in NTCIR10-RITE2. We propose a method for recognizing textual entailment by not using only word alignment, but also using syntactic dependency structure alignment. Entailment can then be recognized by the overlap of the dependency structures. Our system achieved a macro f1 of 63.83 on JA-BC, 49.08 on JA-ExamBC and 74.00 on JA-UnitTest.

Team Name

TKDDI
Tohoku University and KDDI R&D Laboratories, Inc.

Subtasks

BC subtask, ExamBC subtask, and UnitTest (Japanese)

Keywords

textual entailment, alignment, dependency structure

1. INTRODUCTION

For two given sentences t1 and t2, recognizing textual entailment (RTE) is a task of recognizing whether t1 is infered by t2 [4]. Many researchers address this task by measuring word overlap [6], matching transformed expressions [15], using logic-based approaches [19]. Other methods for addressing this task, as exemplified in NTCIR-9 RITE [13], include taking a machine learning-based approach. In addition, most of the teams in NTCIR-9 RITE also integrated a word alignment feature. Word alignment measures a similarity of two words between t1 and t2 based on their surface similarity, lexical similarity etc.

In regards to word alignment, there are some word pairs which are similar but when aligned do not keep the original meaning of the sentences. For example, given the following sentences, all words in t2 can be aligned to t1 based on surface similarity, but t2 is not entailed by t1.

(1) t1 John$_a$ bought$_b$ four$_c$ books and three pencils$_d$.

t2 John$_a$ bought$_b$ four$_c$ pencils$_d$.

Sammons *et al.* [12] aligned a and b as word-level alignment, and aligned phrases "four books and three pencils" and "four pencils" as phrase-level alignment. After the alignment phase, they consider local decision as to whether "four pencils" matches "four books and three pencils".

We propose a method for recognizing textual entailment by not using only word alignment, but also using syntactic dependency structure alignment. Entailment can then be recognized by the overlap of the dependency structures. Chang *et al.* [3] used a similar strategy as constraints for Integer Linear Programming. They achieved the best performance using the data on RTE5 [1]. We confirm an effect of the approach on Japanese RTE data.

In this paper, we describe our system developed for NTCIR10-RITE2 [18]. We participate in JA-BC, JA-ExamBC, and JA-UnitTest subtasks. In particular, because our system is based on simple overlaps, JA-UnitTest is the important subtask for our system. As our system consists of glass box methods, the target of the system is transparent. For example, the system cannot address the pairs that need higher level inferences such as "implicit_relations". Other categories such as synonymy, scrambling and modifier are targets of our system. We confirmed that in experiments.

2. DEPENDENCY STRUCTURE ALIGNMENT-BASED APPROACH

In this section, we describe our approach for recognizing textual entailment between two sentences t1 and t2. We developed a system that recognizes entailment by aligning linguistic units and their syntactic dependency structures. For example, in Figure 1, all *chunks* in t2 are aligned to t1 (dotted lines), and all dependency structures in t2 are aligned to t1 (curved lines). The pair is then classified into Y(entailment).

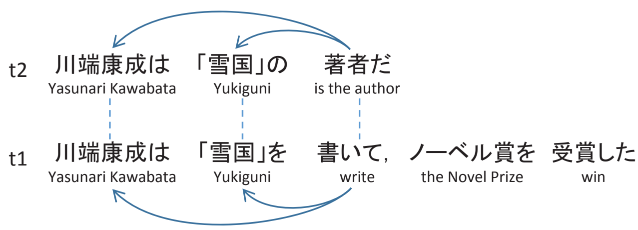

Figure 1: An alignment example

In the example, t1 is "Yasunari Kawabata wrote Yukiguni and won the Novel Prize"' and t2 is "Yasunari Kawabata is the author of Yukiguni".

In order to recognize entailment between t1 and t2, we detect alignments between their linguistic units. While in English, an alignment is usually done at the word level, whereas in Japanese, a popular unit of alignment is the phrase-like

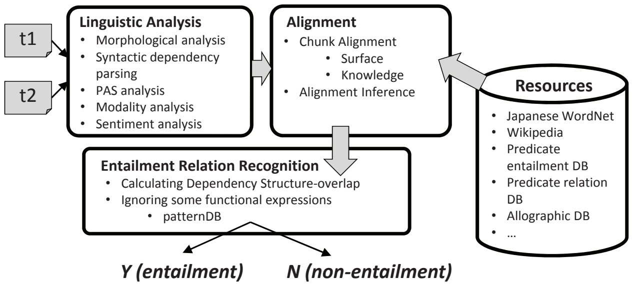

Figure 2: A system overview

unit called a *chunk*. A *chunk* consists of one or more content words and following zero or more functional words. Some of the functional words in *chunk* represent syntactic roles which is useful for recognizing semantic relations between sentences. In this paper, we use a *chunk* as the unit of alignment.

Figure 2 shows an overview of our system. Our system first conducts various forms of linguistic analysis: morphological analysis using MeCab [9], syntactic parsing using CaboCha [8] and predicate-argument structure analysis [17] to provide a basis for alignment.

In the Alignment Phase, *chunks* in t1 and t2 are aligned. Our system performs alignment by utilizing several external resources for linguistic and world knowledge. To overcome the coverage limitations of existing resources, our system also applies *alignment inference*, a method we developed to infer the presence of alignments between *chunks* in t1 and t2 without using external resources by analyzing the similarity of syntactic and semantic dependencies between chunks.

Finally, the system recognizes an entailment relation between a pair of texts. We participated in JA-BC, JA-ExamBC, and JA-UnitTest, so our system only classifies pairs Y(entailment) or N(non-entailment). When a ratio between the number of aligned dependency structures and the number of the dependency structures in t2 is higher than a threshold value, the pair is classified into Y.

As follows, we describe the details of our alignment approach in Section 3, the external resources used for alignment in Section 4, and our rule-based entailment relation recognizer in Section 5.

3. ALIGNMENT

In order to recognize textual entailment between a pair of texts, t1 and t2, it is necessary to identify which parts of t1 and t2 are semantically related. This is done through *alignment*.

First, we align *chunks* and label their local semantic relations. The decision to align a pair of *chunks* in t1 and t2 is made by analyzing the content words in the *chunks*.

We identify alignments based on lexical similarity of without using external resources in the *surface-based alignment* phase and based on semantic relatedness using external resources in the *knowledge-based phrase*.

3.1 Surface-based Alignment

In surface-based alignment, content words' head in the pair of *chunks*, $t1_a$ and $t2_a$, are converted into their dictionary forms, and the *chunks* are aligned if the content words' head in $t2_a$ are found in $t1_a$

As the following example shows that ⟨ ディズニーランドに "Disney Land"⟩ in t2 and ⟨ 東京ディズニーランドへ "to Tokyo Disney Land"⟩ in t1 are aligned because their head words are both ⟨ ディズニーランド "Disney Land"⟩. In other words, the functional word ⟨ へ "to"⟩ and the modifier word ⟨ 東京 "Tokyo"⟩ are ignored.

(2) t2 彼は ディズニーランドに$_a$ 行った
He went to Disney Land$_a$

t1 彼は 東京ディズニーランドへ$_a$ 行った
He went to Tokyo Disney Land$_a$

The reason of using only head words in the surface-based alignment because is to put emphasis on alignment recall. By this rough approach, we can flexibly align orthographic variants such as named entities like ⟨ ディズニーランド "Disney Land"⟩ and ⟨ 東京ディズニーランド "Tokyo Disney Land"⟩.

3.2 Knowledge-based Alignment

We use various resources for knowledge as described in Section 4 to determine semantic similarity. Each entry in the knowledge consists of two expressions and a semantic relation. The semantic relation is used to label the resulting alignment.

During this alignment phase, a pair of *chunks*, c_{t1} and c_{t2}, is aligned if a semantic relation between an expression including the headword of c_{t2} and a headword in c_{t1} is found in one of the knowledge sources. Along with the surface-based alignment, we use only head words. Each words in the *chunks* are matched against the entries in the resources using a character-level bi-gram cosine-based similarity measure [11]. If the average of the cosine similarity between the *chunk* and the entry in the resource is higher than a particular threshold, then the pair is aligned. In our implementation, the threshold of cosine value is set to 0.7. In the following example, ⟨ 生態系を "the ecosystem"⟩ and ⟨ 環境を "the environment"⟩ are aligned because the semantic relation, "ecosystem - (synonym) - environment", was found and the cosine-based similarity of each pair of expressions is higher than the threshold.

(3) t2 ブラックバスは 生態系を 破壊する
Black bass destroy the ecosystem

t1 ブラックバスは 環境を 破壊する
Black bass destroy the environment

3.3 Alignment Inference

In spite of using massive amounts of linguistic knowledge, there are many uncovered words. In particular, the coverage of domain-specific expressions is poor. For instance, consider ⟨ 畑で農薬を使用する "Agricultural chemicals are used in the field."⟩ and ⟨ 畑に農薬を散布する "Agricultural chemicals are sprayed on the field."⟩, ⟨ 使用する "used"⟩ entails ⟨ 散布する "sprayed"⟩ but none of our existing knowledge bases contain this relation.

To infer alignments between infrequent expressions, we develop a heuristic that relies on syntactic and semantic dependency similarity measurements between t1 and t2 that can be obtained from dependency parsing and predicate argument structure analysis. Our intuition is that predicates which have similar argument structures are likely to be alignments, so we align two *chunks* c_{t1} and c_{t2} if they are both predicates and at least two arguments are lexically aligned.

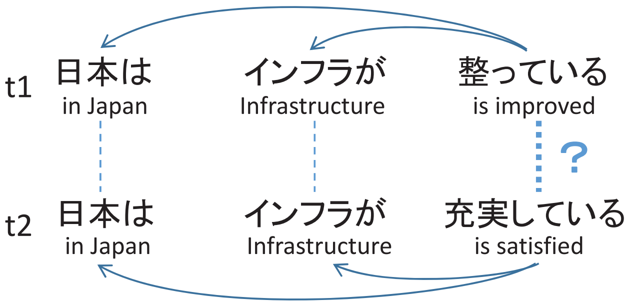

Figure 3: Alignment inference detects an alignment between two predicates that are not covered by any knowledge sources

Figure 3 shows an example of alignment inference. First, the *chunk* pair of ⟨日本は "in Japan"⟩ in t2 and ⟨日本は "in Japan"⟩ in t1 and the pair of ⟨インフラが "Infrastructure"⟩ in t2 and ⟨インフラが "Infrastructure"⟩ in t1. Note that both pairs of arguments are aligned in Surface-based Alignment. Finally, ⟨充実している "is satisfied"⟩ in t2 and ⟨整っている "is improved"⟩ in t1 are judged to be semantically similar due to their aligned argument structures.

The above case illustrated the alignment of predicates when many of their arguments were aligned. However, there are cases in where predicates are aligned even if they only share one aligned argument, as detailed below.

(a) **predicates indicate existence or non-existence**

When both predicates indicate existence or non-existence such as ⟨ある "exist"⟩ or ⟨少ない "few"⟩, the predicates are aligned even if they share only one argument. We manually constructed a list of predicates which indicate existence or non-existence.

(b) **predicates have sentiment polarity**

When both predicates have same sentiment polarity values, the predicates are aligned even if they share only one argument.

4. RESOURCES

Since large-scale lexical knowledge is absolutely essential to recognizing the semantic relation between words, we use various large-scale lexical resources. These resources are used for knowledge-based alignment as described in Section 3. This section describes how these resources are used. Table 1 gives an overview of the resources applied by our system.

4.1 Ontologies

We use the Japanese WordNet [2] to check whether the two words in t1 and t2 have hypernym or synonym relations. E.g. ⟨効果 "good effect" - 作用 "effect"⟩.

In addition, we use Wikipedia as a linguistic resource. Wikipedia has a massive amount of information on diverse topics such as sports, history and so on. However the entries in Wikipedia are unstructured. Extracting linguistic knowledge from Wikipedia is an important task for NLP. We use hypernym-hyponym relations [16] and synonym relations extracted from Wikipedia. Synonym relations can be extracted automatically from redirect database. Some words are hyper-linked to another word as "redirect". These words can be considered as synonyms or paraphrases [14].

Table 1: Resources

Resource	# of Entries
Allographic DB [7]	61,555
Japanese WordNet [2]	1,437,672
Predicate Entailment DB [5]	121,508
Predicate Relation DB [10]	32,314
Wikipedia Hyponyms [16]	3,893,452
Wikipedia Synonyms [14]	494,292

4.2 Predicate Relation Databases

To check whether the two predicates are semantically related, we use a database of relations between predicates [10] and a database of predicate entailments [5].

The database provided by [5] includes not only predicate entailment relations but also preconditions (e.g. ⟨酔っ払う "get drunk"⟩ - ⟨飲む "drink"⟩), action-reaction relations (e.g. ⟨借りる "borrow" -貸す "lend"⟩), antonyms, estimated events (e.g. ⟨紅葉する "leaves are colored"⟩ - ⟨落葉する "leaves fall"⟩), semantically related events (e.g. ⟨成立 "organize,form"⟩ - ⟨誕生 "birth"⟩). The database contains 52,689 predicate pairs which have entailment relations, and 121,508 pairs in total.

Also, the database of [10] includes the following relations: antonym, cause, effect, goal, hyponym, meronym and near_synonym, and contains 32,314 predicate pairs in total.

4.3 Allographic Knowledge

The Japanese Allographic Database [7] is a database used to check allographic ambiguity between two expressions. E.g. ⟨排気ガス "exhaust gas"⟩ and ⟨排ガス "exhaust gas"⟩. The *chunks* in t1 and t2 are aligned if the content words are allographic variants of each other.

5. RECOGNIZING ENTAILMENT

In the entailment recognition phase, a pair of texts is classified as either Y(entailment) or N(non-entailment). The entailment is recognized based on word-overlap and dependency structure-overlap between t1 and t2. For example, in Figure 3, all chunks are aligned in the alignment phase. Afterwards, we focus on their dependency structure. In t2, there are two dependency structures which both correspond to t1. The system classifies a pair of texts into Y when a ratio between the number of corresponding dependency structures and the number of the dependency structures in t2 is higher than a threshold value. The threshold value is set experimentally to 0.7. This result is submit as **TKDDI-01**.

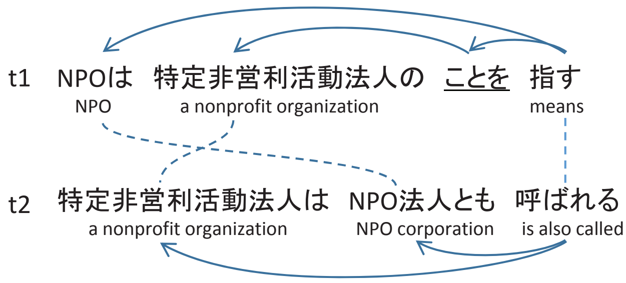

Figure 4: An example of functional expressions

However, some functional expressions prevent matching

Table 2: Results on the BC subtask (JA)

submission	Macro F1	Accuracy	F1(Y)	Prec.(Y)	Rec.(Y)	F1(N)	Prec.(N)	Rec.(N)
baseline	74.85	75.25	78.02	80.48	75.71	71.67	68.95	74.61
TKDDI-01	63.45	68.69	49.60	76.42	36.72	77.29	66.74	91.81
TKDDI-02	63.55	68.69	49.87	76.00	37.11	77.23	66.80	91.53
TKDDI-03*	63.83	69.02	50.13	77.24	37.11	77.53	66.94	92.09

Table 3: Results on the ExamBC subtask (JA)

submission	Macro F1	Accuracy	F1(Y)	Prec.(Y)	Rec.(Y)	F1(N)	Prec.(N)	Rec.(N)
baseline	58.75	62.28	46.69	51.39	42.77	70.81	67.43	74.55
TKDDI-01	48.62	62.28	22.12	54.55	13.87	75.11	63.12	92.73
TKDDI-02	48.62	62.28	22.12	54.55	13.87	75.11	63.12	92.73
TKDDI-03*	49.08	62.50	22.94	55.56	14.45	75.22	63.28	92.73

Table 4: Results on the UnitTest subtask (JA)

submission	Macro F1	Accuracy	F1(Y)	Prec.(Y)	Rec.(Y)	F1(N)	Prec.(N)	Rec.(N)
baseline	72.89	89.21	93.93	93.06	94.81	51.85	56.00	48.28
TKDDI-01	73.51	85.48	91.32	96.34	86.79	55.70	44.00	75.86
TKDDI-02	73.51	85.48	91.32	96.34	86.79	55.70	44.00	75.86
TKDDI-03*	74.00	85.89	91.58	96.35	87.26	56.41	44.90	75.86

Marked results are unofficial submits.

dependency structures. In Figure 4, a functional chunk ⟨ ことを (functional word in Japanese)⟩ in t1 (underlined) is an example. When considering only directly dependent structure, a dependency structure between ⟨ 特定非営利活動法人は "a nonprofit organization"⟩ and ⟨ 呼ばれる "is also called"⟩ in t2 does not match ⟨ 特定非営利活動法人の "a nonprofit organization"⟩ and ⟨ 指す "means"⟩ in t1. Therefore, this pair is classified incorrectly as N.

For this problem, we used pattern paraphrase database[1] for matching ⟨A は B のことを指す "A means B"⟩ and ⟨B は A とも呼ばれる "B is called A"⟩. In order to disregard small differences such as particles between a pattern and part of text, bi-gram cosine measure is used as a similarity measure. In order to search similar patterns fast, we used SimString [11]. This result is submit as **TKDDI-03**.

Our system also integrates another approach in order to disregard functional expressions. This approach is based on a number of documents containing such expressions. For each verb phrase, the system compares a number of Web documents[2] to the original expression and a shortened expressions. The shortened expressions are generated automatically by removing the last chunk. For example, when the expression ⟨ 分類される ことが 多い "they are often classified"⟩ is given, the last chunk ⟨ 多い "often"⟩ is removed and compared to a number of documents containing ⟨ 分類される ことが 多い "they are often classified"⟩ and a number of documents containing the shortened expression ⟨ 分類される "they are classified"⟩. When the latter number is larger than the former number, the system decides whether or not the expression can be shortened. Note that when counting a number of documents containing the shortened expressions, the original expression is not concluded. In other words, the queries to compare are "+original_expression" and "+shortened_expression AND -original_expression". This comparison iterates that a number of documents containing an original expression is larger than a shortened one or until the expression is just one chunk. This result is submit as **TKDDI-02**.

6. RESULTS

For our experiments, we evaluate our system in JA-BC, JA-ExamBC and JA-UnitTest. The baseline system recognizes entailment by word overlap. All alignment strategies are used in the baseline method and a pair of sentences are classified as Y(entailment) when a ratio of a number of aligned words to a number of words in t2 is higher than 0.7; otherwise, the sentences are classified as N(non-entailment).

The experimental results are shown in Table 2 through 4. Focusing on an overall accuracy and a macro F1, the proposed methods do not outperform the baseline. As all aligned *chunks* in the proposed methods are also aligned in the baseline method, the recall of N is low and the recall of Y is high. This result shows that the constraint of dependency structure alignment is too strict which reduces correct alignments. In contrast to JA-BC and JA-ExamBC, the proposed method is outpeformed by the baseline method in JA-UnitTest.

The results of TKDDI-03 indicate that the alignment considering functional expressions effects the performance. On the other hand, the results of TKDDI-02 indicate that the Web counting-based method does not effect the performance of RTE. The reason for this appears to due to the fact that the corpus size is too small.

7. DISCUSSION

Figures 5, 7 and 9 show the entailment threshold-overall accuracy curves. Figures 6, 8 and 10 show entailment threshold-macro f1 curves. For each graph, the curves show the difference of the threshold value of bi-gram cosine similarity used in the alignment phase.

All results, aside from the JA-UnitTest, accuracy have almost same trend. According to an imbalanced number of Y and N in JA-UnitTest (212 Y pairs and 29 N pairs), if entailment threshold is low, all pairs are classified into Y and the overall accuracy does not become lower when macro f1 becomes lower.

[1]The database is available on `https://alaginrc.nict.go.jp/images/documents/pattern_ALAGIN_v1_README.pdf`.
[2]We used Wikipedia documents written in Japanese as the corpus.

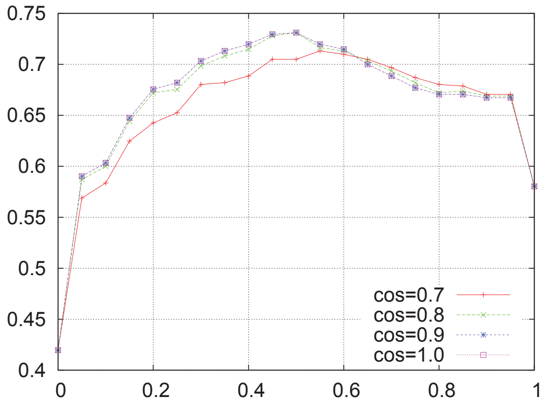

Figure 5: A result of JA-BC test (Accuracy)

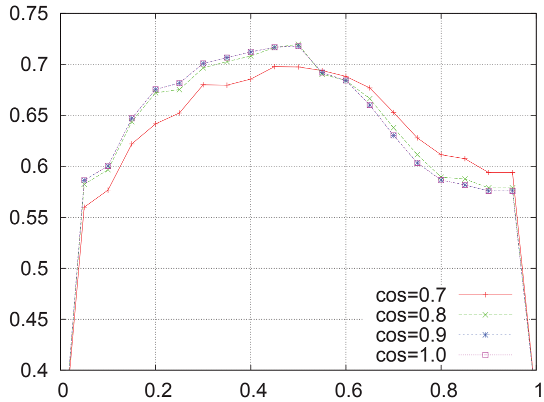

Figure 6: A result of JA-BC test (Macro F1)

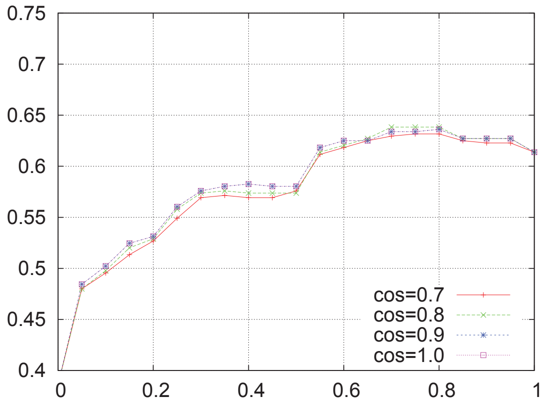

Figure 7: A result of JA-ExamBC test (Accuracy)

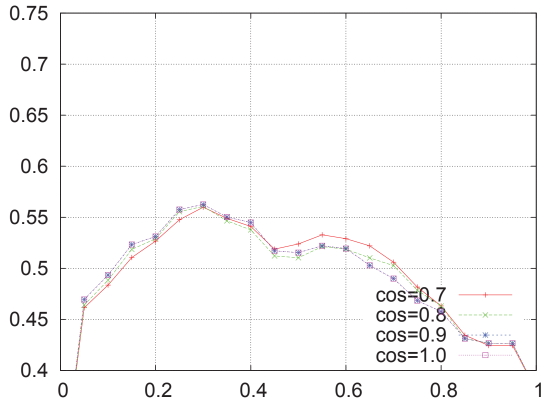

Figure 8: A result of JA-ExamBC test (Macro F1)

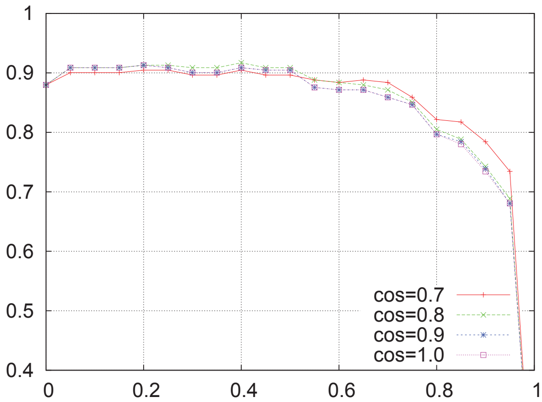

Figure 9: A result of JA-UnitTest test (Accuracy)

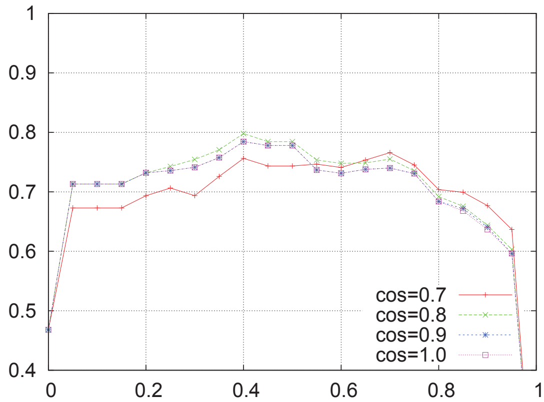

Figure 10: A result of JA-UnitTest test (Macro F1)

The horizontal axis shows threshold value to decide entailment and the vertical axis shows accuracy or macro f1.

Table 5: Detailed results on UnitTest data for each category

Category	Prec.(Y)	Rec.(Y)	F1(Y)	Prec.(N)	Rec.(N)	F1(N)
case_alternation	100.00 (7/ 7)	100.00 (7/ 7)	100.00	0.00 (0/ 0)	0.00 (0/ 0)	0.00
inference	0.00 (0/ 0)	0.00 (0/ 2)	0.00	0.00 (0/ 2)	0.00 (0/ 0)	0.00
spatial	100.00 (1/ 1)	100.00 (1/ 1)	100.00	0.00 (0/ 0)	0.00 (0/ 0)	0.00
implicit_relation	100.00 (17/ 17)	94.44 (17/ 18)	97.14	0.00 (0/ 1)	0.00 (0/ 0)	0.00
list	100.00 (3/ 3)	100.00 (3/ 3)	100.00	0.00 (0/ 0)	0.00 (0/ 0)	0.00
disagree:lex	0.00 (0/ 0)	0.00 (0/ 0)	0.00	100.00 (2/ 2)	100.00 (2/ 2)	100.00
synonymy:phrase	100.00 (32/ 32)	91.43 (32/ 35)	95.52	0.00 (0/ 3)	0.00 (0/ 0)	0.00
meronymy:lex	100.00 (1/ 1)	100.00 (1/ 1)	100.00	0.00 (0/ 0)	0.00 (0/ 0)	0.00
apposition	100.00 (1/ 1)	100.00 (1/ 1)	100.00	0.00 (0/ 0)	0.00 (0/ 0)	0.00
modifier	100.00 (42/ 42)	100.00 (42/ 42)	100.00	0.00 (0/ 0)	0.00 (0/ 0)	0.00
transparent_head	100.00 (1/ 1)	100.00 (1/ 1)	100.00	0.00 (0/ 0)	0.00 (0/ 0)	0.00
synonymy:lex	100.00 (9/ 9)	90.00 (9/ 10)	94.74	0.00 (0/ 1)	0.00 (0/ 0)	0.00
nominalization	100.00 (1/ 1)	100.00 (1/ 1)	100.00	0.00 (0/ 0)	0.00 (0/ 0)	0.00
coreference	100.00 (4/ 4)	100.00 (4/ 4)	100.00	0.00 (0/ 0)	0.00 (0/ 0)	0.00
disagree:phrase	0.00 (0/ 6)	0.00 (0/ 0)	0.00	100.00 (19/ 19)	76.00 (19/ 25)	86.36
temporal	100.00 (1/ 1)	100.00 (1/ 1)	100.00	0.00 (0/ 0)	0.00 (0/ 0)	0.00
disagree:modality	0.00 (0/ 1)	0.00 (0/ 0)	0.00	0.00 (0/ 0)	0.00 (0/ 1)	0.00
entailment:phrase	100.00 (34/ 34)	75.56 (34/ 45)	86.08	0.00 (0/ 11)	0.00 (0/ 0)	0.00
disagree:temporal	0.00 (0/ 1)	0.00 (0/ 0)	0.00	0.00 (0/ 0)	0.00 (0/ 1)	0.00
hypernymy:lex	100.00 (3/ 3)	100.00 (3/ 3)	100.00	0.00 (0/ 0)	0.00 (0/ 0)	0.00
scrambling	100.00 (13/ 13)	86.67 (13/ 15)	92.86	0.00 (0/ 2)	0.00 (0/ 0)	0.00
clause	100.00 (14/ 14)	100.00 (14/ 14)	100.00	0.00 (0/ 0)	0.00 (0/ 0)	0.00
relative_clause	100.00 (8/ 8)	100.00 (8/ 8)	100.00	0.00 (0/ 0)	0.00 (0/ 0)	0.00

Table 5 shows the detailed results of JA-UnitTest using TKDDI-03 settings. Focusing on Y, there is no true positive for the "inference" category. Considering the following example below, t2 contains temporal information (underlined), but t1 does not. Our system cannot find corresponding information in t1; therefore it is classified incorrectlyas N.

(4) t1 満州事変は塘沽協定で停戦している
The Manchurian Incident was ceased fire by Tanggu Truce

t2 満州事変は 1933 年の 塘沽協定で停戦している
The Manchurian Incident was ceased fire by Tanggu Truce in 1933

Focusing on N, there is no true positive for "disagree:temporal". In this category, there are some temporal contradictions between t1 and t2. The t2 of the example contains temporal information 〈 数ヶ月に渡って "across some months"〉 but there is no corresponding information in t1. However, because other parts of the sentence are the same, it is classified incorrectly as Y.

Both "disagree:phrase" and "entailment:phrase" depend on lexical knowledge in our strategy. In particular, some pairs contain a presupposition relation are not classified correctly. As seen from the example below, a presupposition of "use" is "exist". However, there is no such knowledge in resources described in Section 4.

(5) t1 土鍋は寄せ鍋をはじめとして、多くの鍋料理に対して 用いられる
An earthen pot is used in many one-pot meals such as chowder.

t2 寄せ鍋をはじめとして、多くの鍋料理が存在する
Many kinds of one-pot meals such as chowder exist.

8. CONCLUSION

In this paper we described the TKDDI system for the NTCIR10-RITE2. In the experiments of JA-UnitTest, many sentence pairs are classified correctly. However, the system classified incorrectly for "disagree:phrase", "entailment:phrase" and "inference". We are planning to address these problems by improving alingmnet inference.

9. REFERENCES

[1] L. Bentivogli, I. Dagan, H. Dang, D. Giampiccolo, and B. Magnini. The fifth pascal recognizing textual entailment challenge. *Proceedings of TAC*, 9:14–24, 2009.

[2] F. Bond, H. Isahara, S. Fujita, K. Uchimoto, T. Kuribayashi, and K. Kanzaki. Enhancing the Japanese WordNet. In *Proceedings of the 7th Workshop on Asian Language Resources*, pages 1–8, 2009.

[3] M. Chang, D. Goldwasser, D. Roth, and V. Srikumar. Discriminative learning over constrained latent representations. In *Proc. of NAACL*, pages 429–437, 2010.

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

[5] C. Hashimoto, K. Torisawa, K. Kuroda, M. Murata, and J. Kazama. Large-scale verb entailment acquisition from the web. In *Conference on Empirical Methods in Natural Language Processing (EMNLP2009)*, pages 1172–1181, 2009.

[6] V. Jijkoun and M. de Rijke. Recognizing textual entailment using lexical similarity. In *Proc. of the PASCAL Challenges Workshop on RTE*, 2005.

[7] M. Kojima, M. Murata, J. Kazama, K. Kuroda, A. Fujita, E. Aramaki, M. Tsuchida, Y. Watanabe,

and K. Torisawa. Using Various Features in Machine Learning to Obtain High Levels of Performance for Recognition of Japanese Notational Variants. In *Proceedings of The 24th Pacific Asia Conference on Language, Information and Compuation (PACLIC 24)*, pages 653–660, 2010.

[8] T. Kudo and Y. Matsumoto. Japanese dependency analysis using cascaded chunking. In *Proc. of CoNLL*, pages 63–69, 2002.

[9] T. Kudo, K. Yamamoto, and Y. Matsumoto. Appliying conditional random fields to japanese morphological analysis. In *In Proc. of EMNLP*, 2004.

[10] S. Matsuyoshi, K. Murakami, Y. Matsumoto, , and K. Inui. A database of relations between predicate argument structures for recognizing textual entailment and contradiction. In *Proc. of the 2nd International Symposium on Universal Communication (ISUC2008)*, pages 366–373, 2008.

[11] N. Okazaki and J. Tsujii. Simple and efficient algorithm for approximate dictionary matching. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 851–859, Beijing, China, August 2010.

[12] M. Sammons, V. Vydiswaran, T. Vieira, N. Johri, M. Chang, D. Goldwasser, V. Srikumar, G. Kundu, Y. Tu, and K. Small. Relation alignment for textual entailment recognition. *Proceedings of Recognizing Textual Entailment 2009*, 2009.

[13] H. Shima, H. Kanayama, C.-W. Lee, C.-J. Lin, T. Mitamura, Y. Miyao, S. Shi, and K. Takeda. Overview of NTCIR-9 RITE: Recognizing Inference in TExt. In *NTCIR-9 Proceedings*, 2011.

[14] H. Shima, Y. Li, N. Orii, and T. Mitamura. LTI's Textual Entailment Recognizer System. *Proceedings of NTCIR-9 Workshop Meeting*, pages 1–8, Nov. 2011.

[15] A. Stern, E. Shnarch, A. Lotan, S. Mirkin, L. Kotlerman, N. Zeichner, J. Berant, and I. Dagan. Rule Chaining and Approximate Match in textual inference. *Proceedings of the Third Text Analysis Conference (TAC 2010) November*, pages 15–16, 2010.

[16] A. Sumida, N. Yoshinaga, and K. Torisawa. Boosting Precision and Recall of Hyponymy Relation Acquisition from Hierarchical Layouts in Wikipedia. In *Proc. of the 6th International Language Resources and Evaluation (LREC'08)*, pages 2462–2469, 2008.

[17] Y. Watanabe, M. Asahara, and Y. Matsumoto. A structured model for joint learning of argument roles and predicate senses. In *Proceedings of the 48th Annual Meeting of the Association of Computational Linguistics*, pages 98–102, 2010.

[18] 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.

[19] Y. Watanabe, J. Mizuno, E. Nichols, N. Okazaki, and K. Inui. A latent discriminative model for compositional entailment relation recognition using natural logic. In *Proceedings of COLING 2012*, pages 2805–2820, 2012.