A Surface-Similarity Based Two-Step Classifier for RITE-VAL

Shohei Hattori Satoshi Sato
Graduate School of Engineering, Nagoya University
Furo-cho, Chikusa-ku, Nagoya, 464-8603, JAPAN
{syohei_h,ssato}@nuee.nagoya-u.ac.jp

ABSTRACT

This paper describes the system of the team SKL in the NTCIR-11 RITE-VAL workshop. The system consists of two modules: RTE module and text-search module. The RTE module, which is a modified version of our previous system for the binary classification in the RITE-2 workshop, takes two-step classification strategy. The first step classifies a given text pair into positive or negative entailment class based on an overlap measure. If the pair is classified into positive class, the second step examines whether the assigned class should be flipped or not by using heuristic rules that detect the mismatch of named entities and numbers. The Fact Validation subtask in this workshop is to determine whether a given hypothesis is true or not based on a given document. For this subtask, we introduce the text-search module that extracts the text segment from the document; the RTE module produces the final output from the extracted text segment and the hypothesis.

Team Name

SKL

Subtasks

Fact Validation, System Validation (Japanese)

Keywords

two-step classification strategy, overlap ratio, overriding rules

1. INTRODUCTION

RITE-VAL [3] is the third workshop that concerns textual entailment in the NTCIR workshop series. RITE-VAL introduces a new subtask called *Fact Validation*, which is to determine whether a given hypothesis is true or not based on a given document. This subtask is more practical than the binary classification (BC) task in the previous RITE2 workshop, where a text segment (usually a sentence) is explicitly given to determine the truth of the hypothesis [5].

For the RITE2 workshop, we implemented a surface-similarity based system (SKL-01) [1, 2], which takes two-step classification strategy. This system achieves high performance in the RITE2 formal run: the system ranked 7th among 42 systems in the BC subtask, and the MC subsystem ranked first among 21 systems in the MC subtask [5].

For the RITE-VAL workshop, we have implemented a new system with two modules. The RTE (Recognizing Textual Entailment) module is a modified version of the previous system for RITE2; a new text-search module is introduced to identify the text segment that is used by the RTE module.

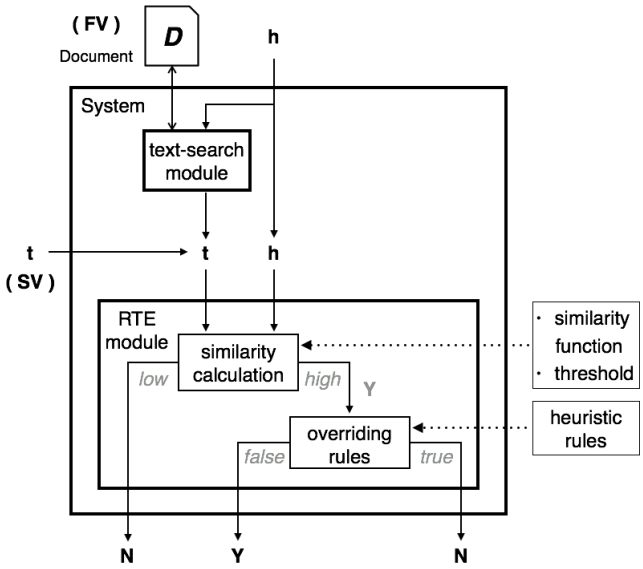

Figure 1: System overview

The rest of this paper is organized as follows. Section 2 overviews our new system. Section 3 and 4 describe the RTE module and the text-search module, respectively. Section 5 reports our experimental results in the development stage and the formal run.

2. SYSTEM OVERVIEW

There are two subtasks in RITE-VAL: Fact Validation (FV) and System Validation (SV). The FV subtask is, given a document set (**D**) and a hypothesis (**h**), to determine whether **D** entails **h** or not. The SV subtask is, given a text (**t**) and a hypothesis (**h**), to determine whether **t** entails **h** or not.

Figure 1 shows the configuration of the system for RITE-VAL. The system consists of two modules: the RTE module and the text-search module. The RTE module solves the SV subtask. The combination of two modules solves the FV subtask, where the text-search module extracts a text (**t**) from the document (**D**).

3. RTE MODULE

The RTE module is a modified version of our previous system for RITE2 [1]. The system recognizes textual entailment in two steps. The first step assigns a default class to a given text pair (**t** and **h**) based on the surface similarity between two texts, where a given similarity function and a threshold are used. The second step examines the necessity of overriding the default class by applying a given set of heuristic rules.

3.1 Similarity Functions

For the similarity calculation, we use the overlap ratio shown in Equation (1), which has been introduced in [1].

$$\text{overlap\_ratio}(\boldsymbol{E};\mathbf{t},\mathbf{h}) = \frac{\sum_{x\in\boldsymbol{E}} \min(\text{f}(x,\mathbf{t}),\text{f}(x,\mathbf{h}))}{\sum_{x\in\boldsymbol{E}} \text{f}(x,\mathbf{h})} \quad (1)$$

In this formula, $\boldsymbol{E}$ is a set of entities such as characters or words, and $\text{f}(x,\mathbf{t})$ is a function that calculates the frequency of an entity x in a text **t**.

In this paper, we use four different similarity functions in total, which are calculated from four different entity sets: characters ($\boldsymbol{C}$), Kanji and Katakana characters ($\boldsymbol{K}$), words ($\boldsymbol{W}$), and nouns ($\boldsymbol{N}$).

$$\text{cor}(\mathbf{t},\mathbf{h}) = \text{overlap\_ratio}(\boldsymbol{C};\mathbf{t},\mathbf{h}) \quad (2)$$
$$\text{kor}(\mathbf{t},\mathbf{h}) = \text{overlap\_ratio}(\boldsymbol{K};\mathbf{t},\mathbf{h}) \quad (3)$$
$$\text{tor}(\mathbf{t},\mathbf{h}) = \text{overlap\_ratio}(\boldsymbol{W};\mathbf{t},\mathbf{h}) \quad (4)$$
$$\text{nor}(\mathbf{t},\mathbf{h}) = \text{overlap\_ratio}(\boldsymbol{N};\mathbf{t},\mathbf{h}) \quad (5)$$

3.2 Overriding Rules

An overriding rule is to examine whether the default class should be flipped or not in the second step. The action part of a rule is always "flip positive into negative". The condition part is a binary function that examines semantic dissimilarity. In the previous system, we have implemented two functions: NE_mismatch and Num_mismatch [1]. In addition, we introduce Year_mismatch.

3.2.1 Mismatch of Named Entities or Numbers

In general, a mismatch of named entities or numbers causes semantic dissimilarity, even if two texts are highly similar on the surface. The functions NE_mismatch and Num_mismatch return true when **h** includes a named entity or number that does not appear in **t**. For the named entity detection, we use JUMAN, a Japanese morphological analyzer. For the number detection, we use the character type (Arabic numeral).

3.2.2 Match of Years

From the analysis of our previous formal-run result in RITE2, we have noticed that the simple exact-match judgment sometimes fails. A typical case is years with a *range* expression, such as "1940's" and "from 1943 to 1948" in the following text pair.

Table 1: Year expressions

type	cue phrase	normalized expression
single-year	n 年 (in n)	n
century	n 世紀 (in the n century)	$[n_{start} - n_{end}]$
range	n_1 年から n_2 年 (from n_1 to n_2)	$[n_1$-$n_2]$
	n 年代 (in the n's)	$[n_{start} - n_{end}]$
	n_1 世紀から n_2 世紀 (from n_1 century to n_2 century)	$[n_{start} - n_{end}]$

t：ジョン・ケネス・ガルブレイスは 1943 年から 1948 年にかけて「フォーチュン」誌の編集者を務めた。
(John Kenneth Galbraith served as the editor of "Fortune" magazine from 1943 to 1948.)
h：フォーチュンは，1940 年代，ジョン・ケネス・ガルブレイスを編集員として起用した．
(In the 1940's, "Fortune" appointed John Kenneth Galbraith as editional committee member.)

For this text pair, the function Num_mismatch returns true because **t** does not have "1940". This is incorrect because **t** entails **h** in this case.

Shibata et al. [4] reported that approximately 30% of text pairs in the development data of the RITE2 ExamSearch task [5] contain time (year) expressions. This development data is reused for the development data of the RITE-VAL FV subtask.

In order to save this type of failure, we introduce the new function Year_match. This function first extracts the *year expressions* from each text (**t** and **h**) by using cue phrases shown in Table 1. When an extracted year expression has the range (i.e., not a single-year), it is normalized into the pair of the start year and the end year. For example, "1940 年代 (1940's)" is normalized into "1940–1949"; "1943 年から 1948 年 (from 1943 to 1948)" is normalized into "1943–1948".

When **h** contains a single-year type expression, the function return true if **t** has the same year. When **h** contains a year expression with a range, the function returns true if the range of years in **h** covers that in **t**. For example, the function returns true for the example text pair, because "1940–1949" covers "1943–1948". Note that the function Year_mismatch is the negation of the function Year_match.

4. TEXT-SEARCH MODULE

In the FV subtask, a document (textbook) **D** is given instead of a text (**t**). The text-search module is responsible for extracting a text segment from the document **D**, which is used as **t** by the RTE module.

First, we assume that the document is a sequence of text segments. Under this assumption, the problem is simplified into the selection of the best segment from the sequence. This selection can be executed in the following two steps.

1. Decomposing the document into the sequence of text segments.

2. Selecting the best text segment from the sequence.

```
<textbook>
  <page>
    <title>〈1〉コンビニから現代をみると</title>
    <id>JA-1</id>
    <text>
      <sectionTitle>1.　グローバル化の時代</sectionTitle>
      <pageNumber>p.18〜19</pageNumber>
      <sectionTitle>〈1〉コンビニから現代をみると</sectionTitle>
      <topic>コンビニの成長</topic>
      <p>　私たちはなぜコンビニエンスストア (コンビニ) を利用するのだろうか。それは，コンビニが文字通りコンビニエンス (便利) だからである。立ち寄りやすい場所にある，欲しいものがそろっている，いつでも開いている，公共料金の支払いや宅配便の受付ができる，などさまざまな便利さをあげることができる。</p>
      <p>　コンビニの歴史は新しい。日本のコンビニは 1969(昭和 44) 年にはじまり，74 年にアメリカの企業と提携した大規模チェーン店が本格的なコンビニを出店した。その後，さまざまな企業が参入し，1980 年代に全国で 1 万店を数えた。そして，90 年代にめざましく増えて，4 万店をこえるまでになった。つまり，この教科書をつかっているみなさんが成長するのと時を同じくして，コンビニは増え，広がってきたのである。</p>
      …
```

Figure 2: An example of the textbook data

Table 2: Performance of overlap ratios

segment	similarity function	M-F1	th.
sentence	cor	55.7	0.90
sentence 2-gram		57.7	0.83
sentence 3-gram		57.2	0.89
paragraph	cor	55.3	0.90
	kor	58.0	0.88
	tor	55.0	0.86
	nor	**59.7**	0.69

For the first *decomposition* step, we use a pre-defined unit in the document. The document (textbook) provided for the FV subtask is a XML document, where several types of unit, such as section, topic, and paragraph, have been marked up, as shown in Figure 2. We use one of which because of no extra processing.

For the second *selection* step, we take a brute force strategy; we calculate scores of all segments in the document and select the segment with the highest score. Because the selected segment is used as $\mathbf{t}$ by the RTE module, the score should be the same as the similarity score in the first step of the RTE module.

There are several choices for the segment unit and the similarity function. We have experimentally explored the best combination, as shown in Table 2. Based on the result, we have decided to use *paragraph* for the text segment and noun overlap ratio (nor) for the similarity function.

5. EXPERIMENTS

5.1 FV subtask

5.1.1 Setting

Table 3 shows the configuration of the five runs that we submitted. All runs use *paragraph* as the text segment ($\mathbf{t}$), and the noun overlap ratio (nor) as the similarity function except FV-05. Each threshold value has been determined so that M-F1 score is the highest for the development data.

For the FV subtask, the RITE-VAL organizers provided two textbooks for the document $\mathbf{D}$.

Table 3: Configuration of five runs in FV subtask

run	textbook	similarity function	th.	overriding rules
FV-01	$\mathbf{D}_1 + \mathbf{D}_2$	nor$(p, \mathbf{h})$	0.69 ($\mathbf{D}_1$) 0.69 ($\mathbf{D}_2$)	NE, Year
FV-02	$\mathbf{D}_1$	nor$(p, \mathbf{h}')$	0.72	
FV-03	$\mathbf{D}_2$	nor$(p, \mathbf{h}')$	0.56	
FV-04	$\mathbf{D}_1 + \mathbf{D}_2$	nor$(p, \mathbf{h}')$	0.72 ($\mathbf{D}_1$) 0.56 ($\mathbf{D}_2$)	
FV-05	$\mathbf{D}_1 + \mathbf{D}_2$	wnor$(p, \mathbf{h}')$	0.83 ($\mathbf{D}_1$) 0.79 ($\mathbf{D}_2$)	

Table 4: Decision by using two documents

Decision by using $\mathbf{D}_1$	Decision by using $\mathbf{D}_2$	Final decision
Y	*	**Y**
*	**Y**	**Y**
N	**N**	**N**

- ntcir10_rite2_rite2-ja-textbook.xml ($\mathbf{D}_1$)
- riteval-ja-textbook2.xml ($\mathbf{D}_2$)

FV-02 uses $\mathbf{D}_1$; FV-03 uses $\mathbf{D}_2$; and the other runs use both $\mathbf{D}_1$ and $\mathbf{D}_2$. In case the system uses two documents, it first calculates the result (i.e., Y or N) by using each document and then determines the final result according to Table 4.

In all runs except FV-01, we have added one or two twists to our system aiming to improve the system performance.

The first is a weighting of nouns, which is intended to reflect the importance of words in the similarity calculation. We estimate the weight of the noun x based on the frequency in the document $\mathbf{D}$.

$$w(x, \mathbf{D}) \quad = \quad \frac{1}{\log_2\{\mathrm{f}(x, \mathbf{D}) + 1\}} \tag{6}$$

In this formula, the function $\mathrm{f}(\mathrm{x}, \mathbf{D})$ is the same with that in Equation (1). By using this weight, we define weighted noun overlap ratio (wnor) as follows.

```
def SV-01(t,h)
  if ((cor(t, h) ≥0.73) or
      (kor(t, h) > cor(t, h) ≥ 0.69) or
      ((0.69 > cor(t, h) > 0.65) and
       (kor(t, h)− 0.1 > cor(t, h))
     if (NE_mismatch or Num_mismatch or Year_mismatch)
      return N
     else
      return Y
     end
  else
   return N
  end
end
```

Figure 5: Pseudo-code of SV-01

$$\mathrm{wnor}(\mathbf{t},\mathbf{h},\mathbf{D}) = \frac{\sum_{x\in\boldsymbol{N}} \min(\mathrm{f}(x,\mathbf{t}),\mathrm{f}(x,\mathbf{h}))\cdot w(x,\mathbf{D})}{\sum_{x\in\boldsymbol{N}} f(x,\mathbf{h})\cdot w(x,\mathbf{D})} \quad (7)$$

where the set $\boldsymbol{N}$ is a set of nouns. FV-05 uses wnor as the similarity function.

The second is a pre-process of hypothesis. An examination of the development data reveals that year expressions in **h** tend to decrease the surface similarity because the extracted segment (**t**) has no year expression in many cases. Therefore, we decided to introduce the pre-process that removes a year expression from the text segment before similarity calculation, as follows.

h 1990 年代の後半には, 日本版ビッグバンと呼ばれる金融制度の改革が行われた。

h' 日本版ビッグバンと呼ばれる金融制度の改革が行われた。

All runs except FV-01 employ this pre-process ($\mathbf{h}'$ in Table 3 means this).

5.1.2 Result and Discussion

Table 5 shows the results of the FV subtask in the development stage and the formal run. FV-02 achieves the highest M-F1 score for the development set; FV-01 achieves the highest for the formal run.

The effects of the pre-process and the weighting of nouns are limited. Table 6 shows the results of ablation test, where both textbooks $\mathbf{D_1}$ and $\mathbf{D_2}$ are used. In the development stage, the combination of two twists slightly improves M-F1 by comparing FV-01 with FV-05. By contrast, it worsens M-F1 in the formal run.

5.2 SV subtask

For the SV subtask, we submitted only one run (SV-01). The configuration of SV-01 is the same as SKL-01 [1] in RITE2 except the set of overriding rules. Figure 5 shows the pseudo-code of SV-01. The similarity function of this system is a combined function of two similarity functions, cor and kor; the latter is employed to improve the classification accuracy around the boundary ($0.73 > \mathrm{cor} > 0.65$). Note that the similarity function for SV subtask differs from that for FV subtask. The set of overriding rules for the second step is slightly revised; Year_mismatch is newly introduced in addition to two rules (NE_mismatch and Num_mismatch).

Table 7 shows the result of the SV subtask in the formal run. This result shows that the system does not work well in recognizing **Y** pairs compared with **N** pairs.

Figure 3 and 4 show the histograms of the character overlap ratio (cor) for the RITE2 data set and the RITE-VAL data set, respectively. From these tables, we can see that the distributions of text pairs are quite different in the two data sets. In case of RITE2 data set (Figure 3), many **Y** pairs are distributed in the area with high cor value ($\mathrm{cor} \leq 0.69$). By contrast, in case of RITE-VAL (Figure 4), many **Y** pairs are distributed in the area with low cor value ($\mathrm{cor} < 0.69$). This difference causes the poor performance in recognizing **Y** pairs.

6. REFERENCES

[1] S. Hattori and S. Sato. Team SKL's Strategy and Experience in RITE2. In *Proceedings of the 10th NTCIR Conference*, pages 435–442, 2013.

[2] S. Hattori, S. Sato, and K. Komatani. Surface-Similarity Based Textual Entailment Recognition for Japanese Text. In *Journal of the Japanese Society for Artificial Intelligence*, vol.29, No.4, pages 416–426, 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. Shibata, S. Kurohashi, S. Kohama, and A. Yamamoto. Predicate-argument Structure based Textual Entailment Recognition System of KYOTO Team for NTCIR-10 RITE-2. In *Proceedings of the 10th NTCIR Conference*, pages 537–544, 2013.

[5] 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 the NTCIR-10 Workshop. In *Proceedings of the 10th NTCIR Conference*, pages 385–404, 2013.

Table 5: Results in the FV subtask

	run	Label	Precision	Recall	F1	Acc.	M-F1
Development Y:210 N:300	FV-01	Y	51.72 (120 / 232)	57.14 (120 / 210)	54.30	60.39	59.68
		N	67.63 (188 / 278)	62.67 (188 / 300)	65.05		
	FV-02	Y	56.22 (113 / 201)	53.81 (113 / 210)	54.99	**63.73**	**62.31**
		N	68.61 (212 / 309)	70.67 (212 / 300)	69.62		
	FV-03	Y	52.60 (101 / 192)	48.10 (101 / 210)	50.25	60.78	58.94
		N	65.72 (209 / 318)	69.67 (209 / 300)	67.64		
	FV-04	Y	50.94 (136 / 267)	64.76 (136 / 210)	57.02	59.80	59.63
		N	69.55 (169 / 243)	56.33 (169 / 300)	62.25		
	FV-05	Y	55.87 (100 / 179)	47.62 (100 / 210)	51.41	62.94	60.73
		N	66.77 (221 / 331)	73.67 (221 / 300)	70.05		
Formal run Y:208 N:306	FV-01	Y	50.00 (94 / 188)	45.19 (94 / 208)	47.47	59.53	**57.28**
		N	65.03 (212 / 326)	69.28 (212 / 306)	67.09		
	FV-02	Y	50.75 (68 / 134)	32.69 (68 / 208)	39.77	59.92	54.87
		N	63.16 (240 / 380)	78.43 (240 / 306)	69.97		
	FV-03	Y	45.36 (88 / 194)	42.31 (88 / 208)	43.78	56.03	53.84
		N	62.50 (200 / 320)	65.36 (200 / 306)	63.90		
	FV-04	Y	46.55 (108 / 232)	51.92 (108 / 208)	49.09	56.42	55.50
		N	64.54 (182 / 282)	59.48 (182 / 306)	61.90		
	FV-05	Y	51.91 (68 / 131)	32.69 (68 / 208)	40.12	**60.51**	55.33
		N	63.45 (243 / 383)	79.41 (243 / 306)	70.54		

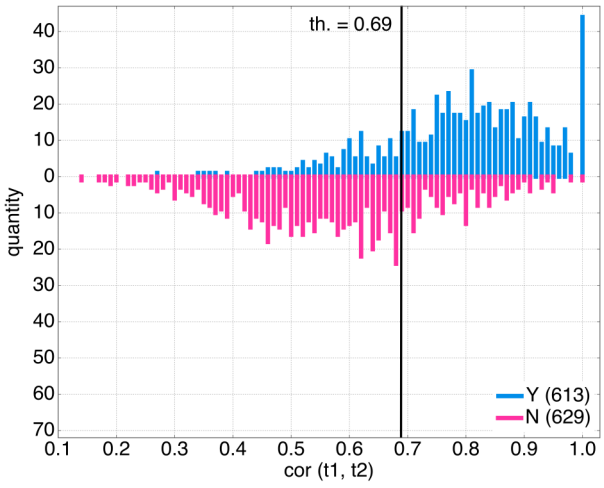

Figure 3: Histogram of cor (RITE2 data set)

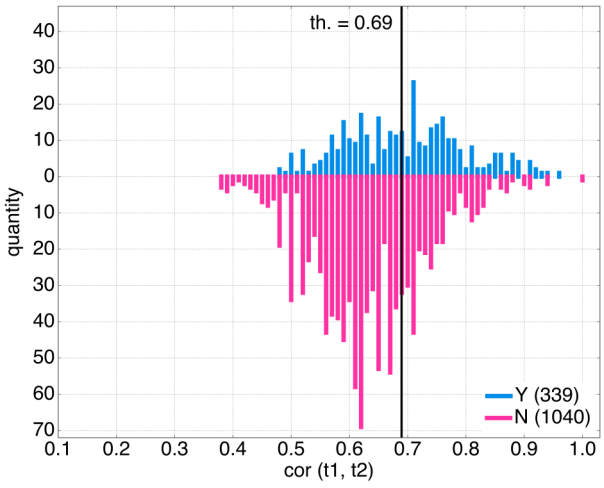

Figure 4: Histogram of cor (RITE-VAL data set)

Table 6: Ablation test

		pre-process	weighting	Acc.	M-F1	Y-F1	N-F1
Development	(FV-01)			60.39	59.68	54.30	65.05
	(FV-04)	✓		59.80	59.63	57.02	62.25
			✓	60.78	59.34	51.69	67.00
	(FV-05)	✓	✓	62.94	60.73	51.41	70.05
Formal run	(FV-01)			59.53	57.28	47.47	67.09
	(FV-04)	✓		56.42	55.50	49.09	61.90
			✓	58.37	55.13	43.09	67.18
	(FV-05)	✓	✓	60.51	55.33	40.12	70.54

Table 7: Result in the SV subtask

run	Label	Precision	Recall	F1	Acc.	M-F1
SV-01	Y (339)	47.59 (148 / 311)	43.66 (148 / 339)	45.54	74.33	64.37
	N (1040)	82.12 (877 / 1068)	84.33 (877 / 1040)	83.21		