# IKOMA at TAC2011: A Method for Recognizing Textual Entailment using Lexical-level and Sentence Structure-level features

**Masaaki Tsuchida** **Kai Ishikawa**
Information and Media Processing Laboratories, NEC Corporation, Nara, Japan.
{m-tsuchida@cq, k-ishikawa@dq}.jp.nec.com

## Abstract

This paper describes the Recognizing Textual Entailment (RTE) system that our teams developed for TAC 2011. Our system combines the entailment score calculated by lexical-level matching with the machine-learning-based filtering mechanism using various features obtained from lexical-level, chunk-level and predicate argument structure-level information. In the filtering mechanism, we try to discard the T-H pairs that have high entailment score and are actually not entailment. That is, for filtering false positive T-H pairs caused by our lexical-level manner, we use additional information like features from word chunks and predicate-argument structures.

## 1 Introduction

The Recognizing Textual Entailment (RTE) task is concerned with the question whether a given Text (T) entails a given Hypothesis (H). T entails H if, typically, a human reading T would infer that H is most likely true. For example, if "President Barack Obama visited Japan" is given as H and "President Obama met Japanese Prime Minister in Tokyo" is given as T, RTE systems should answer "*T entails H*".

In recent years, many research groups have participated in the PASCAL RTE challenges. Up to RTE-5, they developed sophisticated methods based on logical inference (Hickl and Bensley, 2007; Clark and Harrison, 2009), similarity between dependency parse trees (Bar-Haim et al., 2009) or similarity between syntactic graphs (Padò et al., 2009). Sammons et al. (2010) reports that such previous works have made significant progress (Sammons et al., 2010) beyond a *smart* lexical baseline (Do et al., 2009). This suggests that structured semantic content such as dependency parse trees and syntactic graph beyond lexical-level one is required for sophisticated RTE system to perform well. However, the top 3 systems (Jia et al., 2010; Majumdar and Bhattacharyya, 2010; Tateishi and Ishikawa, 2010) in RTE-6 were basically lexical-level matching approaches, which might be considered as *baseline* up to RTE-6. Therefore we think that lexical-level methods are still important, though we try to enhance our RTE system by structural information.

This paper reposts our RTE system that combines the entailment score calculated by lexical-level matching and the machine-learning-based filtering mechanism using various features obtained from lexical-level, chunk-level and predicate-argument-structure-level information. In the filtering mechanism, we try to discard the T-H pairs that have high entailment score and are actually not entailment. That is, false positive T-H pairs classified by our lexical-level manner are discarded by the filtering mechanism using various features including more than lexical-level one.

## 2 Description of Our System

Our system first calculates the entailment score of given T-H pairs and detects the entailment pair candidates by their scores with the threshold learned from the development data. Then the machine-learning-based filtering discards the entailment pair candidates that have high scores but seem *not entail*. For this filtering, we use various features from word

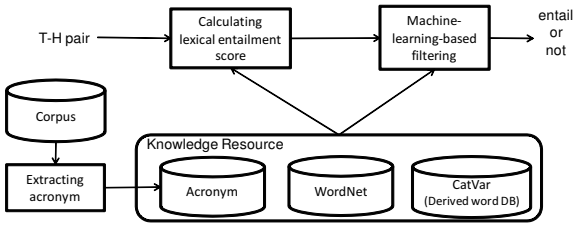


Figure 1: System Architecture

chunks and predicate-argument-structures, which are more than lexical-level information. In our system, we use SENNA [1] for analyzing part-of-speeaches of words, word chunks, named entities and predicate-argument-strucures in sentences. Figure 1 shows our system architecture. Our system uses the three knowledge resources, acronym extracted from the corpus, WordNet (Fellbaum, 1998) and CatVar (Habash and Dorr, 2003), which contains categorical variations of English lexemes. The acronyms are created for organizational names with more than three words, by selecting the first letter of each word.

In the following, we describe the entailment score and the machine-learning-based filtering mechanism in detail.

### 2.1 Lexical Entailment Score

In this score, we assume that T entails H when T has sufficient words common to or in entailment relation with the words in H. Thus the entailment score between T and H is defined as follows.

$$ent\_sc(\mathrm{T},\mathrm{H}) = \frac{\sum_{t_h \in \mathbf{H_t}} match(t_h, \mathbf{T_t}, \mathbf{R}) w(t_h)}{\sum_{t_h \in \mathbf{H_t}} w(t_h)} \quad (1)$$

$$w(t) = (\log \frac{N}{freq(t)})^{\alpha}$$

Here, each $\mathbf{T_t}$ and $\mathbf{H_t}$ denotes a set of words in each given text T and H. $w(t)$ is the weight of the word $t$, and $freq(t)$ is the frequency of the word $t$ in a corpus. $N$ is the number of the texts in the corpus. $R$ is a set of knowledge resources. In our current system, $R$ consists of WordNet and CarVar. $match(t, T_t, R)$ takes 1 if the word $t$ corresponds to a word in $\mathbf{T_t}$ whereas we also consider synonym and derived words in $\mathbf{R}$, otherwise $match(t, T_t, R)$ takes 0.

Before calculating the score, our system normalizes organization names to their acronyms, and then calculates the score. If the entailment score exceeds a threshold, then we consider "T entails H".

In both of main and novelty detection task, we used a threshold value in our system that maximizes the micro-average F-measure for the development set of RTE-7. We also set $\alpha$ in equation 1 to 1.8 in our system through experiments using the development set of RTE-7.

### 2.2 Machine Learning based Filtering

This filtering mechanism aims to discard false-positive T-H pairs caused by the lexical entailment score. For this purpose, we firstly train a model that classifies T-H pairs having high lexical entailment score into false-positive or true-positive. If the model predicts T-H pairs as *false-positive*, then we discards the T-H pairs from entailment T-H pair candidates.

#### 2.2.1 Training Model

The model for filtering is trained by LIBSVM (Chang and Lin, 2011), which is a famous support vector machine (Cortes and Vapnik, 1995) package, with various features obtained from lexical level, chunk level and predicate-argument-structure (PAS) level information. More precisely, we use the following sorts of features.

- Lexical level
  - Entailment score $ent\_sc$
  - Cosine similarity
  - Entailment score $ent\_sc$, where we compare only words with the same part-of-speech tag.
- Chunk level
  - Matching ratios for each chunk types (e.g., NP and VP) in all corresponding chunk pairs
- PAS level
  - Matching ratios for each argument type (Xavier and Lluís, 2005) (e.g.,

[1] http://ml.nec-labs.com/senna/

A0, A1) in all corresponding PAS pairs for each the semantic relation of two predicates.

- The number of negation mismatch in all corresponding PAS pairs for each the semantic relation of two predicates.
- The number of modal verb mismatch in all corresponding PAS pairs for each the semantic relation of two predicates.

In our system, we use *same-expression*, *synonym*, *antonym*, *entailment* and *no-relation*, which any relation are not found, as semantic relations of two predicates. *Synonym*, *antonym* and *entail* relations can be obtained from the WordNet.

For acquiring the above features in chunk and PAS level, we need to detect corresponding pairs that should be checked for testing whether the pairs have entailment. We also need to detect whether such corresponding pairs are in entailment relation. In the following, we omit an explanation in case of chunk level, because the method can be apply to chunk level by straightforward extension.

For the former problem, we propose a simple alignment method for detecting such corresponding pairs. We firstly transform all words contained in PAS into a word vector using bag of words representation, and then calculate the cosine similarity for all PAS pairs that are generated by combining PAS from each T and H. Finally we regard the most similar PAS from T for each PAS from H as *corresponding pairs*. We expect that the method is robust for paraphrased pairs, because the method ignores structure information in sentences.

For the latter problem, for each corresponding pair, we calculate our lexical entailment score between the words of each argument type of the PAS from H (as H in equation 1) and the words of the same argument type of the PAS from T (as T in equation 1). We then regard the argument type in the pair as *matching* if the score exceeds the pre-defined threshold $T_{arg}$. By this way, we count the matching number of each argument type from all pairs distinguished by each relation type of two predicates, and then calculate matching ratios for those. We empirically set $T_{arg}$ to 0.70 in our system.

#### 2.2.2 Filtering by the trained model

The filtering mechanism conservatively modifies the results of T-H pairs detected by the lexical entailment score. That is, we discard such pairs if the model predicts false-positive pairs caused by our lexical entailment scores with high confidence, we discard such pairs, because we found that the lexical entailment score is more reliable than our trained model.

In our filtering mechanism, we first detect a threshold of the model for predicting false-positive pairs with the pre-defined precision $T_{prec}$ by using a development set. Our filtering then discards pairs if the values predicted by the model for pairs exceeds the threshold. That is, the higher $T_{prec}$ is, the more conservative our filter becomes. We empirically set $T_{prec}$ to 80% in our system.

## 3 Evaluation

For main and novelty detection task in RTE-7, we submitted the results of following systems.

**IKOMA1** Lexical entailment score + Filtering with the threshold set by proposed method.

**IKOMA2** Lexical entailment score + Filtering with the default threshold (i.e., 0).

**IKOMA3** Lexical entailment score, only.

Table 1 and Table 3 show the evaluation results for main and novelty detection task in terms of *micro-average*, respectively. Table 1 and 3 show that our filtering can slightly improve precisions and F-measures in some case. For instance, IKOMA1, which is our best system for main task, and IKOMA3, our best system for novelty detection task used the filtering mechanism. However, we could think that the dominant part of the performance is obtained from lexical entailment score, because we cannot find major improvements by the filtering from the F-measures by the our lexical entailment score. We suspect that the feature space of our current filtering model is not suitable enough for our purpose, which is used to detect false-positive pairs caused by the lexical entailment score.

Next, we show the results of the ablation tests using test and development sets in Table 5. We used IKOMA3, which use lexical entailment score only,

Table 5: The ablation test (micro-average) for the test and the development sets for RTE7. We used IKOMA3 as the base system.

| | TEST | | | DEV | | |
|---|---|---|---|---|---|---|
| | Precision | Recall | F-measure | Precision | Recall | F-measure |
| IKOMA3 | 46.51 | 49.46 | 47.94 | 57.72 | 54.31 | 55.96 |
| No Acronym | 47.01(+0.50) | 49.24(+0.16) | 48.10(+0.16) | 56.57(-1.15) | 52.73(-1.58) | 54.63 (-1.33) |
| No CatVar | 47.41(+0.90) | 46.79(-2.67) | 47.10(-0.84) | 58.88(+1.16) | 52.55(-1.76) | 55.53(-0.43) |
| No WordNet | 47.16(+0.65) | 47.02(-2.44) | 47.09(-0.85) | 59.82(+2.1) | 52.82(-1.49) | 56.10(+0.14) |

Table 1: The main task results (micro-average) in RTE7.

| RunID | Precision | Recall | F-measure |
|---|---|---|---|
| IKOMA1 | 46.96 | 49.08 | **48.00** |
| IKOMA2 | 58.48 | 30.05 | 39.70 |
| IKOMA3 | 46.51 | 49.46 | 47.94 |

Table 2: The main task results (macro-average) in RTE7.

| RunID | Precision | Recall | F-measure |
|---|---|---|---|
| IKOMA1 | 48.94 | 50.22 | **49.58** |
| IKOMA2 | 58.87 | 31.95 | 41.42 |
| IKOMA3 | 48.37 | 50.53 | 49.43 |

Table 3: The novelty detection task results (micro-average) in RTE7.

| RunID | Precision | Recall | F-measure |
|---|---|---|---|
| IKOMA1 | 88.73 | 92.82 | 90.73 |
| IKOMA2 | 86.92 | 95.38 | **90.95** |
| IKOMA3 | 88.73 | 92.82 | 90.73 |

Table 4: The novelty detection task results (macro-average) in RTE7.

| RunID | Precision | Recall | F-measure |
|---|---|---|---|
| IKOMA1 | 88.38 | 92.54 | 90.41 |
| IKOMA2 | 86.57 | 95.17 | **90.66** |
| IKOMA3 | 88.38 | 92.54 | 90.41 |

as the base system for the ablation test. In Table 5, we found that knowledge resources tend to improve recall and degrade precision. However, tendency of the contributions of the knowledge resources seems different between the two data sets. Thus we consider that we need more data set and experiments for evaluating contributions of knowledge resources.

## 4 Conclusion

This paper described our system that combines the entailment score calculated by lexical-level matching with the machine-learning-based filtering mechanism using various features obtained from lexical-level, chunk-level and predicate-argument-structure-level information. In our lexical entailment score, assuming that if T adequately have same or entailment words for the words in H, then T entails H, we defined the score reflecting this assumption. In our filtering mechanism, we try to discard false-positive T-H pairs caused by the lexical entailment score. For this filtering, we use additional information obtained from word chunks and predicate-argument-structures, which are beyond lexical-level information.

Our evaluation results show that our filtering mechanism can slightly improve precisions and F-measures in some case. However, we could think that the dominant part of the performance is obtained from lexical entailment score, because we cannot find major improvements by the filtering from the F-measures by the our lexical entailment score. From ablation test using the test set and development set, we showed that knowledge resources tend to improve a recall and degrade a precision, and showed that the contributions of knowledge resources have the different tendency between the two data set.

Through participating in RTE-7, we strengthen our belief that lexical similarity alone is not enough for recognizing textual entailment from the experience that we enhance the lexical-matching-based method. We suggest that breakthrough in RTE is obtained by handling inference and paraphrasing beyond lexical-level.

## References


Roy Bar-Haim, Jonathan Berant, Ido Dagan, Iddo Greental, Shachar Mirkin, Eyal Shnarch, and Idan Szpektor. 2009. Efficient semantic deduction and approximate matching over compact parse forests. In *Proc. of the Text Analysis Conference 2008*.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Peter Clark and Phil Harrison. 2009. An inference-based approach to recognizing entailment. In *Proc. of Text Analysis Conference 2009*.

Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. In *Machine Learning*, pages 273–297.

Quang Do, Dan Roth, Mark Sammons, Yuancheng Tu, and V. G. Vinod Vydiswaran. 2009. Robust, light-weight approaches to compute lexical similarity. Technical report.

Christiane D. Fellbaum. 1998. Wordnet: An electronic lexical database. MIT Press.

Nizar Habash and Bonnie Dorr. 2003. A categorial variation database for english. In *Proc. of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 17–23.

Andrew Hickl and Jeremy Bensley. 2007. A discourse commitment-based framework for recognizing textual entailment. In *Proc. of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 171–176.

Houping Jia, Xiaojiang Huang, Tengfei Ma, Xiaojun Wan, and Jianguo Xiao. 2010. Pkutm participation at tac 2010 rte and summarization track. In *Proc. of Text Analysis Conference 2010*.

Debarghya Majumdar and Pushpak Bhattacharyya. 2010. Lexical based text entailment system for main task of rte6. In *Proc. of Text Analysis Conference 2010*.

Sebastian Padò, Marie-Catherine de Marneffe, and Bill MacCartney. 2009. Deciding entailment and contradiction with stochastic and edit distance-based alignment. In *Proc. of the Text Analysis Conference 2008*.

Mark Sammons, V.G. Vinod Vydiswaran, and Dan Roth. 2010. Ask not what textual entailment can do for you... In *Proc. of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1199–1208.

Kenji Tateishi and Kai Ishikawa. 2010. Ikoma at tac2010: Textual entailment system using local-novelty detection. In *Proc. of Text Analysis Conference 2010*.

Carreras Xavier and Màrquez Lluís. 2005. Introduction to the conll-2005 shared task: semantic role labeling. In *Proc. of the 9th Conference on Computational Natural Language Learning*, pages 152–164.