Semantic Relation Classification using Physical Sizes

Eiji ARAMAKI **Takeshi IMAI** **Kengo MIYO** **Kazuhiko OHE**
The University of Tokyo Hospital department
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan
aramaki@hcc.h.u-tokyo.ac.jp

Abstract

Although researchers have shown increasing interest in extracting/classifying semantic relations, most previous studies have basically relied on lexical patterns between terms. This paper proposes a novel way to accomplish the task: a system that captures a physical size of an entity. Experimental results revealed that our proposed method is feasible and prevents the problems inherent in other methods.

1 Introduction

Classification of semantic relations is important to NLP as it would benefit many NLP applications, such as machine translation and information retrieval.

Researchers have already proposed various schemes. For example, Hearst (1992) manually designed lexico-syntactic patterns for extracting is-a relations. Berland and Charniak (1999) proposed a similar method for part-whole relations. Brin (1998) employed a bootstrapping algorithm for more specific relations (author-book relations). Kim and Baldwin (2006) and Moldovan et al.(2004) focused on nominal relations in compound nouns. Turney (2005) measured relation similarity between two words. While these methods differ, they all utilize lexical patterns between two entities.

Within this context, our goal was to utilize information specific to an entity. Although entities contain many types of information, we focused on the **physical size** of an entity. Here, **physical size** refers to the typical width/height of an entity. For example, we consider *book* to have a physical size of 20×25 cm, and *book* to have a size of 10×10 m, etc.

We chose to use physical size for the following reasons:

1. Most entities (except abstract entities) have a physical size.

2. Several semantic relations are sensitive to physical size. For example, a content-container relation ($e1$ content-container $e2$) naturally means that $e1$ has a smaller size than $e2$. A *book* is also smaller than its container, *library*. A part-whole relation has a similar constraint.

Our next problem was how to determine physical sizes. First, we used Google to conduct Web searches using queries such as "*book (*cm x*cm)*" and "*library (*m x*m)*". Next, we extracted numeric expressions from the search results and used the average value as the physical size.

Experimental results revealed that our proposed approach is feasible and prevents the problems inherent in other methods.

2 Corpus

We used a corpus provided by SemEval2007 Task #4 training set. This corpus consisted of 980 annotated sentences (140 sentences×7 relations). Table 1 presents an example.

Although the corpus contained a large quantity of information such as WordNet sense keys, comments, etc., we used only the most pertinent information: entity1 ($e1$), entity2 ($e2$), and its relation (true/false)

```
The <e1>library</e1> contained <e2>books
</e2> of guidance on the processes.
WordNet(e1) = "library\%1:14:00::",
WordNet(e2) = "book\%1:10:00::",
Content-Container(e2, e1) = "true",
Query = "the * contained books"
```

Table 1: An Example of Task#4 Corpus.

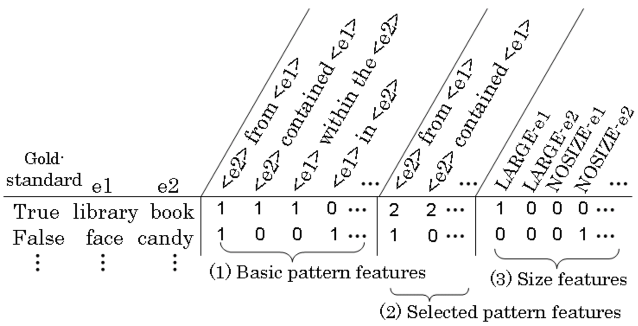

Figure 1: Three types of Features.

[1]. For example, we extracted a triple example (*library*, *book*, *true* from Table 1.

3 Method

We applied support vector machine (SVM)-based learning (Vapnik, 1999) using three types of features: (1) basic pattern features (Section 3.1), (2) selected pattern features (Section 3.2), and (3) physical size features (Section 3.3). Figure 1 presents some examples of these features.

3.1 Basic Pattern Features

First, the system finds lexical patterns that co-occur with semantic relations between two entities ($e1$ and $e2$). It does so by conducting searches using two queries "*e1 * e2*" and "*e2 * e1*". For example, two queries, "*library * book*" and "*book * library*", are generated from Table 1.

Then, the system extracts the word (or word sequences) between two entities from the snippets in the top 1,000 search results. We considered the extracted word sequences to be basic patterns. For example, given "*...library contains the book...*", the basic pattern is "*(e1) contains the (e2)*" [2].

We gathered basic patterns for each relation, and identified if each pattern had been obtained as a SVM feature or not (1 or 0). We refer to these features as **basic pattern features**.

3.2 Selected Pattern Features

Because basic pattern features are generated only from snippets, precise co-occurrence statistics are not available. Therefore, the system searches again with more specific queries, such as "*library contains the book*". However, this second search is a heavy burden for a search engine, requiring huge numbers of queries (# of samples $\times$ # of basic patterns).

We thus selected the most informative n patterns (STEP1) and conducted specific searches (# of samples $\times$ n basic patterns)(STEP2) as follows:

STEP1: To select the most informative patterns, we applied a decision tree (C4.5)(Quinlan, 1987) and selected the basic patterns located in the top n branches [3].

STEP2: Then, the system searched again using the selected patterns. We considered log weighted hits ($\log_{10} |hits|$) to be selected pattern features. For example, if "*library contains the book*" produced 120,000 hits in Google, it yields the value $\log_{10}(12,000) = 5$.

3.3 Physical Size Features

As noted in Section 1, we theorized that an entity's size could be a strong clue for some semantic relations.

We estimated entity size using the following queries:

1. "$< entity >$ (* cm x * cm)",
2. "$< entity >$ (* x * cm)",
3. "$< entity >$ (* m x * m)",
4. "$< entity >$ (* x * m)".

In these queries, $< entity >$ indicates a slot for each entity, such as "*book*", "*library*", etc. Then, the system examines the search results for the numerous expressions located in "*" and considers the average value to be the size.

[1]Our system is classified as an A4 system, and therefore does not use WordNet or Query.

[2]This operation does not handle any stop-words. Therefore, "*(e1) contains THE (e2)*" and "*(e1) contains (e2)*" are different patterns.

[3]In the experiments in Section 4, we set $n = 10$.

	Precision	Recall	$F_{\beta=1}$
PROPOSED	**0.57 (=284/497)**	**0.60 (=284/471)**	**0.58**
+SEL	0.56 (=281/496)	0.59 (=281/471)	0.57
+SIZE	0.53 (=269/507)	0.57 (=269/471)	0.54
BASELINE	0.53 (=259/487)	0.54 (=259/471)	0.53

Table 2: Results.

When results of size expressions were insufficient (numbers < 10), we considered the entity to be non-physical, i.e., to have no size.

By applying the obtained sizes, the system generates a size feature, consisting of six flags:

1. **LARGE-e1**: ($e1$'s X $>$ $e2$'s X) and ($e1$'s Y $>$ $e2$'s Y)
2. **LARGE-e2**: ($e1$'s X $<$ $e2$'s X) and ($e1$'s Y $<$ $e2$'s Y)
3. **NOSIZE-e1**: only $e1$ has no size.
4. **NOSIZE-e2**: only $e2$ has no size.
5. **NOSIZE-BOTH**: Both $e1$ and $e2$ have no size.
6. **OTHER**: Other.

4 Experiments

4.1 Experimental Set-up

To evaluate the performance of our system, we used a SemEval-Task No#4 training set. We compared the following methods using a ten-fold cross-validation test:

1. **BASELINE**: with only basic pattern features.
2. **+SIZE**: BASELINE with size features.
3. **+SEL**: BASELINE with selected pattern features.
4. **PROPOSED**: BASELINE with both size and selected pattern features.

For SVM learning, we used TinySVM with a linear kernel[4].

4.2 Results

Table 2 presents the results. PROPOSED was the most accurate, demonstrating the basic feasibility of our approach.

Table 3 presents more detailed results. +SIZE made a contribution to some relations (REL2 and REL4). Particularly for REL4, +SIZE significantly boosted accuracy (using McNemar tests (Gillick and Cox, 1989); $p = 0.05$). However, contrary to our expectations, size features were disappointing for part-whole relations (REL6) and content-container relations (REL7).

[4]http://chasen.org/ taku/software/TinySVM/

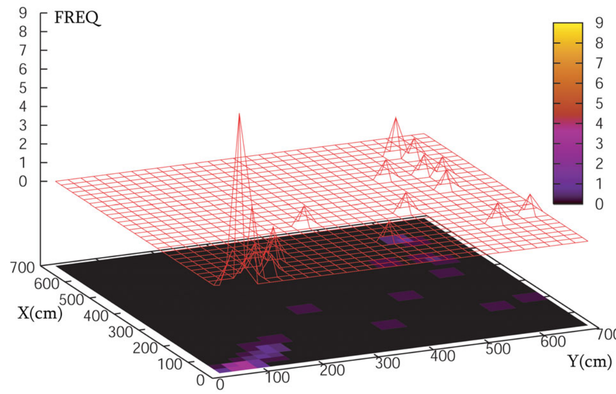

Figure 2: The Size of a "Car".

The reason for this was mainly the difficulty in estimating size. Table 4 lists the sizes of several entities, revealing some strange results, such as a *library* sized 12.1 $\times$ 8.4 cm, a *house* sized 53 $\times$ 38 cm, and a *car* sized 39 $\times$ 25 cm. These sizes are unusually small for the following reasons:

1. Some entities (e.g."*car*") rarely appear with their size,
2. In contrast, entities such as "*toy car*" or "*mini car*" frequently appear with a size.

Figure 2 presents the size distribution of "*car*." Few instances appeared of real cars sized approximately 500 $\times$ 400 cm, while very small cars smaller than 100 $\times$ 100 cm appeared frequently. Our current method of calculating average size is ineffective under this type of situation.

In the future, using physical size as a clue for determining a semantic relation will require resolving this problem.

5 Conclusion

We briefly presented a method for obtaining the size of an entity and proposed a method for classifying semantic relations using entity size. Experimental results revealed that the proposed approach yielded slightly higher performance than a baseline, demonstrating its feasibility. If we are able to estimate en-

Relation		PROPOSED	+SEL	+SIZE	BASELINE
	Precision	**0.60 (=50/83)**	0.56 (=53/93)	0.54 (=53/98)	0.50 (=53/106)
REL1	Recall	0.68 (=50/73)	**0.72 (=53/73)**	**0.72 (=53/73)**	**0.72 (=53/73)**
(Cause-Effect)	$F_{\beta=1}$	**0.64**	0.63	0.59	0.61
	Precision	0.59 (=43/72)	**0.60 (=44/73)**	0.56 (=45/79)	0.55 (=44/79)
REL2	Recall	0.60 (=43/71)	0.61 (=44/71)	**0.63 (=45/71)**	0.61 (=44/71)
(Instrument-Agency)	$F_{\beta=1}$	0.60	**0.61**	0.59	0.58
	Precision	0.70 (=56/80)	**0.73 (=55/75)**	0.65 (=54/82)	0.68 (=51/74)
REL3	Recall	**0.65 (=56/85)**	0.64 (=55/85)	0.63 (=54/85)	0.60 (=51/85)
(Product-Producer)	$F_{\beta=1}$	0.67	**0.68**	0.64	0.64
	Precision	0.41 (=23/56)	0.35 (=18/51)	0.48 (=24/49)	**0.52 (=13/25)**
REL4	Recall	0.42 (=23/54)	0.33 (=18/54)	**0.44 (=24/54)**	0.24 (=13/54)
(Origin-Entity)	$F_{\beta=1}$	0.41	0.34	**0.46**	0.32
	Precision	**0.62 (=40/64)**	0.61 (=40/65)	0.56 (=28/50)	0.56 (=29/51)
REL5	Recall	**0.68 (=40/58)**	**0.68 (=40/58)**	0.48 (=28/58)	0.50 (=29/58)
(Theme-Tool)	$F_{\beta=1}$	**0.65**	**0.65**	0.51	0.53
	Precision	0.45 (=46/101)	**0.46 (=46/100)**	0.41 (=49/118)	0.43 (=53/123)
REL6	Recall	0.70 (=46/65)	0.70 (=46/65)	0.75 (=49/65)	**0.81 (=53/65)**
(Part-Whole)	$F_{\beta=1}$	0.55	0.55	0.53	**0.56**
	Precision	0.63 (26/41)	**0.64 (=25/39)**	0.51 (=16/31)	0.55 (=16/29)
REL7	Recall	**0.40 (26/65)**	0.38 (=25/65)	0.24 (=16/65)	0.24 (=16/65)
(Content-Container)	$F_{\beta=1}$	**0.49**	0.48	0.33	0.34

Table 3: Detailed Results.

entity	#	size
library	51	12.1×8.4 m
room	204	5.4×3.5 m
man	75	1.5×0.5 m
benches	33	93×42 cm
granite	68	76×48 cm
sink	34	57×25 cm
house	86	53×38 cm
books	50	46×24 cm
car	91	39×25 cm
turtles	15	38×23 cm
food	38	35×26 cm
oats	16	24×13 cm
tumor shrinkage	6	-
habitat degradation	5	-

Table 4: Some Examples of Entity Sizes.

"#" indicates the number of obtained size expressions.

"-" indicates a "NO-SIZE" entity.

tity sizes more precisely in the future, the system will become much more accurate.

References

Matthew Berland and Eugene Charniak. 1999. Finding parts in very large corpora. In *Proceedings of the Annual Conference of the Association for Computational Linguistics (ACL1999)*, pages 57–64.

Sergey Brin. 1998. Extracting patterns and relations from the world wide web. In *WebDB Workshop at 6th International Conference on Extending Database Technology, EDBT'98*, pages 172–183.

L. Gillick and SJ Cox. 1989. Some statistical issues in the comparison of speech recognition algorithms. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 532–535.

M. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of International Conference on Computational Linguistics (COLING1992)*, pages 539–545.

Su Nam Kim and Timothy Baldwin. 2006. Interpreting semantic relations in noun compounds via verb semantics. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 491–498.

D. Moldovan, A. Badulescu, M. Tatu, D. Antohe, and R. Girju. 2004. Models for the semantic classification of noun phrases. *Proceedings of HLT/NAACL-2004 Workshop on Computational Lexical Semantics*.

J.R. Quinlan. 1987. Simplifying decision trees. *International Journal of Man-Machine Studies*, 27(1):221–234.

Peter D. Turney. 2005. Measuring semantic similarity by latent relational analysis. In *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI-05)*, pages 1136–1141.

Vladimir Vapnik. 1999. *The Nature of Statistical Learning Theory*. Springer-Verlag.