

Identifying Constant and Unique Relations by using Time-Series Text

Yohei Takaku*

Toyo Keizai Inc.
Chuo-ku, Tokyo 103-8345, Japan
takaku.yohei@gmail.com

Nobuhiro Kaji and Naoki Yoshinaga

Institute of Industrial Science,
University of Tokyo
Meguro-ku, Tokyo 153-8505, Japan
{kaji, ynaga}@tkl.iis.u-tokyo.ac.jp

Masashi Toyoda

Institute of Industrial Science,
University of Tokyo
Meguro-ku, Tokyo 153-8505, Japan
toyoda@tkl.iis.u-tokyo.ac.jp

Abstract

Because the real world evolves over time, numerous relations between entities written in presently available texts are already obsolete or will potentially evolve in the future. This study aims at resolving the intricacy in consistently compiling relations extracted from text, and presents a method for identifying constancy and uniqueness of the relations in the context of supervised learning. We exploit massive time-series web texts to induce features on the basis of time-series frequency and linguistic cues. Experimental results confirmed that the time-series frequency distributions contributed much to the recall of constancy identification and the precision of the uniqueness identification.

1 Introduction

We have witnessed a number of success stories in acquiring semantic relations between entities from ever-increasing text on the web (Pantel and Pennacchiotti, 2006; Banko et al., 2007; Suchanek et al., 2007; Wu et al., 2008; Zhu et al., 2009; Mintz et al., 2009; Wu and Weld, 2010). These studies have successfully revealed to us millions of relations between real-world entities, which have been proven to be beneficial in solving knowledge-rich problems such as question answering and textual entailment (Ferrucci et al., 2010).

*This work was conducted while the first author was a graduate student at University of Tokyo.

There exists, however, a great challenge to compile consistently relations extracted from text by these methods, because they assume a simplifying assumption that relations are time-invariant. In other words, they implicitly disregard the fact that statements in texts actually reflect the state of the world at the time when they were written, which follows that relations extracted from such texts eventually become outdated as the real world evolves over time.

Let us consider that relations are extracted from the following sentences:¹

- (1) a. 1Q84 *is written by* Haruki Murakami.
- b. Moselle river *flows through* Germany.
- c. U.S.'s *president is* George Bush.
- d. Pentax *sells* K-5, a digital SLR.

Here, italicized predicates represent the relations, while underlined entities are their arguments. The relations in statements 1a and 1b are true across time, so we can simply accumulate all the relation instances. The relations in 1c and 1d in contrast evolve over time. The relation written in 1c becomes outdated when the other person takes the position, so we need to supersede it when a new relation is extracted from text (e.g., U.S.'s president is Barack Obama). For the relation in 1d, we do not always need to supersede it with a new relation.

This study is motivated from the above consider-

¹Since our task settings are language-independent, we hereafter employ English examples as much as possible to widen the potential readership of the paper, although we conducted experiments with relations between entities in Japanese.

ations and proposes a method for identifying constancy and uniqueness of relations in order to select an appropriate strategy to maintain relation instances extracted from text. For example, the relations written in statements 1a and 1b are constant, while those in 1c and 1d are non-constant; the relation in 1c is unique,² whereas the relation in 1d is non-unique. With these properties of relations in mind, we can accumulate constant relations while appropriately superseding non-constant, unique relations with newly acquired relations.

We locate each identification task in the context of supervised classification. The key challenge in solving these classification tasks is how to induce an effective feature that identifies unique, non-constant relations (statement 1c) that seemingly appear as non-unique relations on text (statement 1b). We exploit massive time-series web text to observe actual evolutions of relation instances and induce features from the relation instances taken from a time sliding window and linguistic cues modifying the predicate and arguments of the target relation.

We evaluated our method on 1000 relations extracted from 6-year’s worth of Japanese blog posts with 2.3-billion sentences. We have thereby confirmed that the features induced from this time-series text contributed much to improve the classification accuracy.

The main contributions of this paper are twofold:

- We have introduced a novel task for identifying constancy relations. Since most of the existing studies assume that relations are time-invariant as discussed by Weikum et al. (2011), non-constant relations prevalent in their outcome incur a serious problem in maintaining the acquired relations. The notion of constancy is meant to resolve this stalemate.
- We have for the first time demonstrated the usefulness of a time-series text in relation acquisition and confirmed its impact in the two relation classification tasks. The features induced from the time-series text have greatly contributed to the accuracy of the classification based on uniqueness as well as the recall of the classification based on constancy.

²This kind of relation is referred to as functional relation in the literature (Ritter et al., 2008; Lin et al., 2010).

Constant	Non-constant
arg1 was born in arg2	arg1 ’s president is arg2
arg1 is a father of arg2	arg1 belongs to arg2
arg1 is written by arg2	arg1 lives in arg2

Table 1: Examples of constant, non-constant relations.

The remainder of this paper is structured as follows. Section 2 introduces the two properties of relations (constancy and uniqueness) and then defines the task setting of this study. Sections 3 and 4 describe the features induced from time-series text for constancy and uniqueness classification, respectively. Section 5 reports experimental results. Section 6 addresses work related to this study. Section 7 concludes this study and mentions future work.

2 Classification of Relations based on Constancy and Uniqueness

2.1 Constancy and uniqueness

We introduce two properties of relations: *constancy* and *uniqueness*.

A relation is *constant* if, for most values of *arg1*, the value of *arg2* is independent of time (Table 1). For example, $\langle \mathbf{arg1} \text{ was born in } \mathbf{arg2} \rangle$ is a constant relation since one’s birthplace never changes. On the other hand, $\langle \mathbf{arg1} \text{ ’s president is } \mathbf{arg2} \rangle$ is an example of non-constant relations. This can be checked by noting that, for example, the president of the United States was Barack Obama in 2011 but was previously George Bush and Bill Clinton before him.

A relation is *unique* if, for most values of *arg1*, there exists, at any given point in time, only one value of *arg2* that satisfies the relation (Table 2). For example, $\langle \mathbf{arg1} \text{ was born in } \mathbf{arg2} \rangle$ is obviously a unique relation. The relation $\langle \mathbf{arg1} \text{ is headquartered in } \mathbf{arg2} \rangle$ is also unique, while it is non-constant. Notice that there is usually only one headquarters at any point in time, although the location of a headquarters can change. In contrast, the relation $\langle \mathbf{arg1} \text{ is funded by } \mathbf{arg2} \rangle$ is a non-unique relation since it is likely that there exist more than one funder.

2.2 Discussion

Both constancy and uniqueness are properties that usually, not always, hold for most, not all, of the *arg1*’s values. To see this, let us examine the relation $\langle \mathbf{arg1} \text{ ’s president is } \mathbf{arg2} \rangle$. Although this relation is

Unique	Non-unique
arg1 was born in arg2	arg1 is funded by arg2
arg1 is headquartered in arg2	arg1 consists of arg2
arg1 's president is arg2	arg1 borders on arg2

Table 2: Examples of unique and non-unique relations.

non-constant and unique (Table 1 and 2), it is still possible to find exceptional cases. For example, a country might exist in which the president has never changed; a country might have more than one president at the same time during civil war. However, since such situations are rare, the relation $\langle \mathbf{arg1}$'s president is $\mathbf{arg2} \rangle$ is considered as neither constant nor non-unique.

The above discussion implies that the constancy and uniqueness of relations can not be determined completely objectively. We, nevertheless, claim that these properties of relations are intuitively acceptable and thus they can be identified with moderate agreement by different people (see section 5).

2.3 Task and our approach

This paper explores classifying given relations on the basis of constancy and uniqueness. We treat the problem as two independent binary classification tasks, and train supervised classifiers.

The technical challenge we address in this paper is how to design features for the two tasks. Section 3 presents features based on time-series frequency and linguistic cues for classifying constant and non-constant relations. Similarly, section 4 presents analogous features for classifying unique and non-unique relations.

3 Features for Constancy Classification

3.1 Time-series frequency

It is intuitive to identify constant relations by comparing frequency distributions over $\mathbf{arg2}$ in different time periods. This idea leads us to use frequency estimates from time-series text as features.

Time-series text For a time-series text, we used Japanese blog posts that had been gathered from Feb. 2006 to Sep. 2011 (68 months). These data include 2.3 billions of sentences. These posts were aggregated on a monthly basis by using time stamps attached with them, i.e., the unit of time is one month

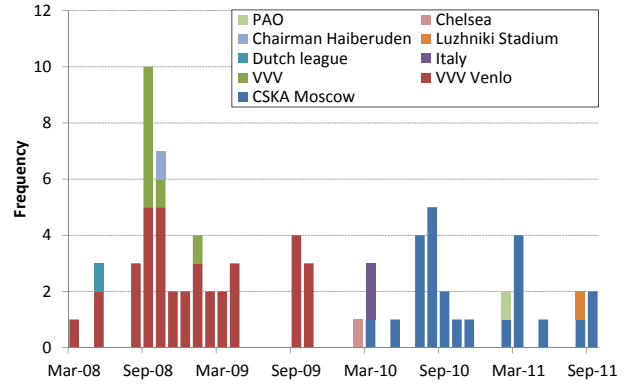


Figure 1: Time-series frequency distribution of $\langle \mathbf{arg1} \text{ belongs to } \mathbf{arg2} \rangle$ when $\mathbf{arg1}$ takes Keisuke Honda.

in our corpus.

Basic idea For constant relations (e.g., $\langle \mathbf{arg1} \text{ was born in } \mathbf{arg2} \rangle$), we can expect that the frequency distributions over $\mathbf{arg2}$ for a given $\mathbf{arg1}$ (e.g., *Mozart*) are similar to each other irrespective of the time windows that are used to estimate frequency.

In the case of non-constant relations (e.g., $\langle \mathbf{arg1} \text{ belongs to } \mathbf{arg2} \rangle$), on the other hand, the frequency distributions over $\mathbf{arg2}$ for a given $\mathbf{arg1}$ significantly differ depending on the time window. For example, Figure 1 illustrates the frequency distributions of $\mathbf{arg2}$ s for $\langle \mathbf{arg1} \text{ belongs to } \mathbf{arg2} \rangle$ in which $\mathbf{arg1}$ takes Keisuke Honda, a famous football player. We can clearly observe that due to Keisuke Honda being sold from VVV Venlo to CSKA Moscow, the distributions differ greatly between 2008 and 2010.

As is evident from the above discussions, the stability/change in the distribution over $\mathbf{arg2}$ is a good indicator of constant/non-constant relations. The following subsection addresses how to encode such information as features.

Feature computation Let us examine using as features the cosine similarity between frequency distributions over $\mathbf{arg2}$. Averaging such similarities over representative values of $\mathbf{arg1}$, we have

$$\frac{1}{N} \sum_{e \in \mathcal{E}_N(r)} \cos(F_{w_1}(r, e), F_{w_2}(r, e)),$$

where r is a relation (e.g., $\langle \mathbf{arg1}$'s president is $\mathbf{arg2} \rangle$), e is a named entity (e.g., *United States*) appearing in $\mathbf{arg1}$, and $F_w(r, e)$ is the frequency distribution over $\mathbf{arg2}$ when $\mathbf{arg1}$ takes e . The subscripts

w_1 and w_2 denote the time window (e.g., from Jan. 2011 to Feb. 2011) used to estimate the frequency distribution. $\mathcal{E}_N(r)$ denotes a set of top N frequent entities appearing in arg1 . We use the entire time-series text to obtain $\mathcal{E}_N(r)$.

Unfortunately, this idea is not suitable for our purpose. The problem is that it is not clear how to determine the two time windows, w_1 and w_2 . To identify non-constant relations, arg2 must have different values in the two time periods. Such time windows are, however, impossible to know of in advance.

We propose avoiding this difficulty by using average, maximum and minimum similarity over all possible time windows:

$$\begin{aligned} & \frac{1}{N} \sum_{e \in \mathcal{E}_N(r)} \text{ave}_{w_1, w_2 \in \mathcal{W}_T} \cos(F_{w_1}(r, e), F_{w_2}(r, e)), \\ & \frac{1}{N} \sum_{e \in \mathcal{E}_N(r)} \max_{w_1, w_2 \in \mathcal{W}_T} \cos(F_{w_1}(r, e), F_{w_2}(r, e)), \\ & \frac{1}{N} \sum_{e \in \mathcal{E}_N(r)} \min_{w_1, w_2 \in \mathcal{W}_T} \cos(F_{w_1}(r, e), F_{w_2}(r, e)), \end{aligned}$$

where \mathcal{W}_T is a set of all time windows of the size T . For example, if we set T to 3 (months) in the 68-month’s worth of blog posts, \mathcal{W}_T consists of 66 ($= 68 - 3 + 1$) time windows. Although we still have to specify the number of entities N and the window size T , this is not a serious problem in practice. We set N to 100. We use four window sizes (1, 3, 6, and 12 months) and induce different features for each window size. As a result, we have 12 real-valued features.

3.2 Linguistic cues

This subsection presents two types of linguistically-motivated features for discriminating between constant and non-constant relations.

Nominal modifiers We observe that non-constant relations could be indicated by some nominal modifiers:

- (2) a. George Bush, *ex*-president of USA.
- b. Lincoln is the *first* president of the USA.

The use of the prefix *ex*- and the adjective *first* implies that the president changes, and hence the relation $\langle \text{arg1} \text{ ’s president is arg2} \rangle$ is not constant.

前 (ex-), 現 (present), 次期 (next), 元 (former), 新 (new), 旧 (old), 歴代 (successive), 初代 (first), 初 (first)

Table 3: Japanese prefixes and adjectives indicating non-constant relations. The translations are provided in the parentheses.

We propose making use of such modifiers as features. Although the above examples are in English, we think modifiers also exist that have similar meanings in other languages including Japanese, our target language.

Our new features are induced as follows:

- First, we manually list eight nominal modifiers that indicate the non-constancy (Table 3).
- Next, we extract nouns from a relation to be classified (e.g., *president*), and count the frequency with which each modifier modifies those nouns. We use the same blog posts as in section 3.1 for counting the frequency. Since time information is not important in this case, the frequency is simply accumulated over the entire time span.
- We then generate eight features, one for each of the eight modifiers. The value of the features is one if the frequency exceeds threshold θ_1 ,³ otherwise it is zero. Note that the value of this feature is always zero if the relation includes no nouns.

Tense and aspect Tense and aspect of verbs are also important indicators of the non-constancy:

- (3) The U.S. president *was* George Bush.

If a relation, such as $\langle \text{arg1} \text{ ’s president is arg2} \rangle$, can often be rephrased in the past tense as in (3), it is likely to be, if not always, a non-constant relation.

It is, fortunately, straightforward to recognize tense and aspect in Japanese, because they are expressed by attaching suffixes to verbs. In this study, we use three common suffixes: “た”, “ている”, and “てる”. The first suffix expresses past tense, while the other two express present continuous or progressive aspects depending on context.

³ $\theta_1 = 10$ in our experiment.

A given relation is transformed into different forms by attaching the suffixes to a verb in the relation, and their frequencies are counted. By using the frequency estimates, we generate three new features, each of which corresponds to one of the three suffixes. The value of the new features is one if the frequency exceeds threshold θ_2 ,⁴ otherwise it is zero.

The frequency is counted in the same way as in the case of the nominal modifiers. The value of this feature is always zero if the relation includes no verbs.

4 Features for Uniqueness Classification

This section provides features for identifying unique relations. These features are also based on the time-series text and linguistic cues, as in the case of constancy classification.

4.1 Time-series frequency

Number of entity types A straightforward approach to identifying unique relations is, for a given arg1 , to count the number of entity types appearing in arg2 (Lin et al., 2010). For unique relations, the number of entity types should be one in an ideal noiseless situation. Even if the estimate is contaminated by noise, a small number of entity types can still be considered to indicate the uniqueness of the relation.

A shortcoming of such a simple approach is that it never considers the (non-)constancy of relations. Presume counting the number of entity types in arg2 of the relation $\langle \text{arg1 is headquartered in arg2} \rangle$, which is non-constant and unique. If we use large size of time window to obtain counts, we will observe multiple types of entities in arg2 , not because the relation is non-unique, but because it is non-constant. This problem cannot be resolved by trivially using very small windows, since a time window that is too small in turn causes a data sparseness problem.

This problem is attributed to the difficulty in determining the appropriate size of the time window. We tackle this problem by using the same technique presented in section 3.1. Specifically, we use the fol-

⁴ $\theta_2 = 3000$ in our experiment.

lowing three measures as features:

$$\frac{1}{N} \sum_{e \in \mathcal{E}_N(r)} \text{ave}_{w \in \mathcal{W}_T} \# \text{type}(F_w(r, e)),$$

$$\frac{1}{N} \sum_{e \in \mathcal{E}_N(r)} \text{max}_{w \in \mathcal{W}_T} \# \text{type}(F_w(r, e)),$$

$$\frac{1}{N} \sum_{e \in \mathcal{E}_N(r)} \text{min}_{w \in \mathcal{W}_T} \# \text{type}(F_w(r, e)),$$

where the function $\# \text{type}(\cdot)$ denotes the number of entity types appearing in arg2 .

Ratio of entity frequency Since it is not reliable enough to use only the number of entity types, we also exploit the frequency of the entity. Let e_{1st} and e_{2nd} be the most and the second most frequent entities found in arg2 . If the frequency of e_{1st} is much larger than that of e_{2nd} , the relation is likely to be constant.

To encode this intuition, the following measures are used as features:

$$\frac{1}{N} \sum_{e \in \mathcal{E}_N(r)} \text{ave}_{w \in \mathcal{W}_T} \frac{f_w(e, r, e_{1st})}{f_w(e, r, e_{2nd})}$$

$$\frac{1}{N} \sum_{e \in \mathcal{E}_N(r)} \text{max}_{w \in \mathcal{W}_T} \frac{f_w(e, r, e_{1st})}{f_w(e, r, e_{2nd})}$$

$$\frac{1}{N} \sum_{e \in \mathcal{E}_N(r)} \text{min}_{w \in \mathcal{W}_T} \frac{f_w(e, r, e_{1st})}{f_w(e, r, e_{2nd})}$$

where the $f_w(e, r, e')$ is the frequency of the relation r in which arg1 and arg2 take e and e' , respectively. The subscript w denotes the time window.

4.2 Linguistic cues

Coordination structures and some keywords indicate non-unique relations:

- (4) a. France borders on *Italy and Spain*.
- b. France borders on *Italy etc*.

The coordination structure in the first example implies an entity can border on more than one entity, and hence the relation $\langle \text{arg1 borders on arg2} \rangle$ is not unique. The keyword *etc*. in the second example also indicates the non-uniqueness.

と, とか, や, やら, だの, なり, か

Table 4: List of Japanese particles that are used to form coordination structures.

To capture this intuition, we introduce two types of linguistic features for classifying unique and non-unique relations. The first feature checks whether entities in `arg2` form coordination structures. The feature is fired if the number of times that coordination structures are found in `arg2` exceeds threshold θ_3 .⁵ Coordination structures are identified by a list of Japanese particles, which roughly correspond to *and* or *or* in English (Table 4). If two entities are connected by one of those particles, they are seen as forming a coordination structure.

The second feature exploits such keywords as *etc.* for identifying non-unique relations. We list four Japanese keywords that have similar meaning to the English word *etc.*, and induce another binary feature⁶. The feature is fired if the number of times that an entity in `arg2` is followed by one of the four keywords exceeds threshold θ_3 .

5 Experiments and discussions

We built labeled data and examine the classification performance of the proposed method. We also analyzed the influence of window size T on the performance, as well as major errors caused by our method.

5.1 Data

We built a dataset for evaluation by extracting relations from the time-series text (section 3.1) and then manually annotating 1000 relations. The detailed procedure is as follows.

First, we parsed the time-series text and extracted as relation dependency paths connecting two named entities. We used J.DepP,⁷ an efficient shift-reduce parser with feature sequence trie (Yoshinaga and Kitsuregawa, 2009; Yoshinaga and Kitsuregawa, 2010), for parsing. All Japanese words that conjugate were normalized into standard forms.

⁵ $\theta_3 = 10$ in our experiment.

⁶The keywords we used are 等, ら, たち, and 達.

⁷<http://www.tkl.iis.u-tokyo.ac.jp/~ynaga/jdepp/>

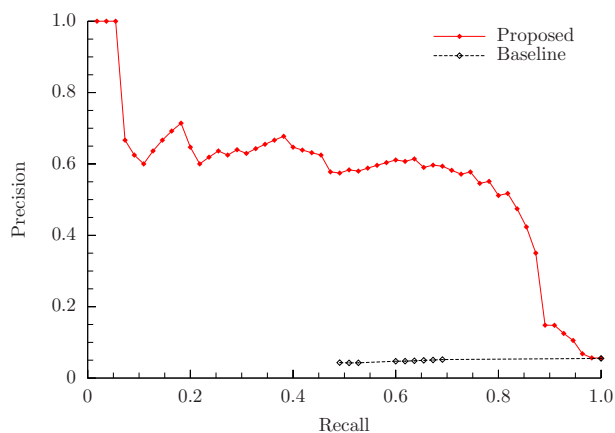


Figure 2: Recall-precision curve (constancy classification).

Then, annotators were asked to label 1000 relations as not only constant or non-constant but also unique or non-unique. Three annotators were assigned to each relation, and the goldstandard label is determined by majority vote. The Fleiss kappa (Fleiss, 1971) was 0.346 for constancy classification and was 0.428 for uniqueness classification. They indicate fair and moderate agreement, respectively (Landis and Koch, 1977).

We have briefly investigated the relations whose labels assigned by the annotators conflicted. The major cause was that the annotators sometimes assumed different types of named entities as values of arguments. A typical case in which this problem arises is that the relation has polysemous meanings, e.g., $\langle \mathbf{arg1}$ was born in $\mathbf{arg2} \rangle$, or a vague meaning, e.g., $\langle \mathbf{arg1}$ makes $\mathbf{arg2} \rangle$. For example, `arg2` of $\langle \mathbf{arg1}$ was born in $\mathbf{arg2} \rangle$ can be filled with different types of entities such as date and place. We can address this problem by typing arguments (Lin et al., 2010).

5.2 Result

Using the dataset, we performed 5-fold cross-validation for both classification tasks. We used the passive-aggressive algorithm for our classifier (Crammer et al., 2006).

Constancy classification Figure 2 illustrates the recall-precision curve in constancy classification. Because we are unaware of any previous methods for classifying constant and non-constant relations, a simple method based on the cosine similarity was

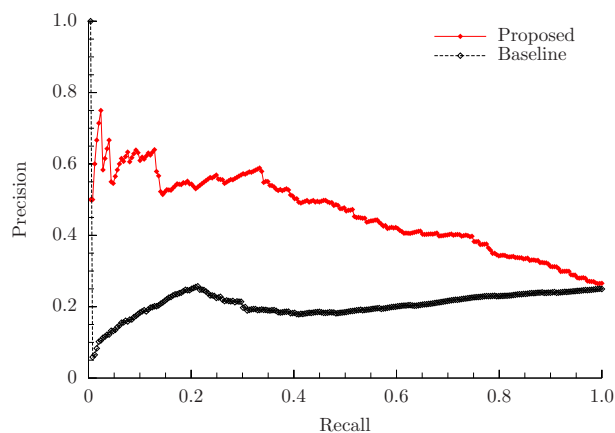


Figure 3: Recall-precision curve (uniqueness classification).

used as a baseline:

$$\frac{1}{N} \sum_{e \in \mathcal{E}_N(r)} \cos(F_{w_1}(r, e), F_{w_2}(r, e)),$$

where the time windows w_1 and w_2 are determined as the first and last month in which the relation r is observed. A given relation is classified as non-constant if the above similarity exceeds a threshold. The recall-precision curve was drawn by changing the threshold.

The results demonstrated that our method outperforms the baseline. This indicates the effectiveness of using time-series frequency and linguistic cues as features.

The poor performance of the baseline was mainly due to data sparseness. Since the baseline method is dependent on the frequency estimates obtained from only two months of texts, it is less reliable than the proposed method.

Uniqueness classification Figure 3 illustrates the recall-precision curve in uniqueness classification. As a baseline we implemented the method proposed by Lin et al. (2010). While they have presented three methods (KLFUNC, KLDIFF, and their average), we report the results of the last one because it performed the best among the three in our experiment.

From the figure, we can again see that the proposed method outperforms the baseline method. Lin’s method is similar to ours, but differs in that they do not exploit time-series information at all.

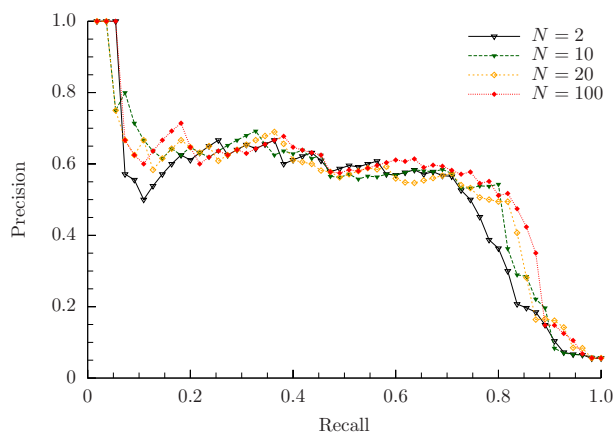


Figure 4: Comparison with the methods varying a value of N for constancy classification.

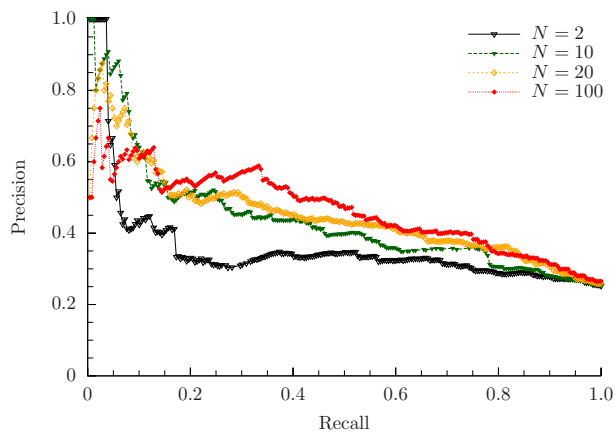


Figure 5: Comparison with the methods varying a value of N for uniqueness classification.

We hence conclude time-series information is useful for classifying not only constant but also unique relations.

5.3 Investigation into the number of entities, N

We ranged the value of N in $\{2, 10, 20, 100\}$. Setting N to a larger value yields the better recall for constancy classification and the better precision for uniqueness classification (Figures 4 and 5). These results meet our expectations, since features derived from frequency distributions of arg2 over various arg1 s capture the generic nature of the target relation.

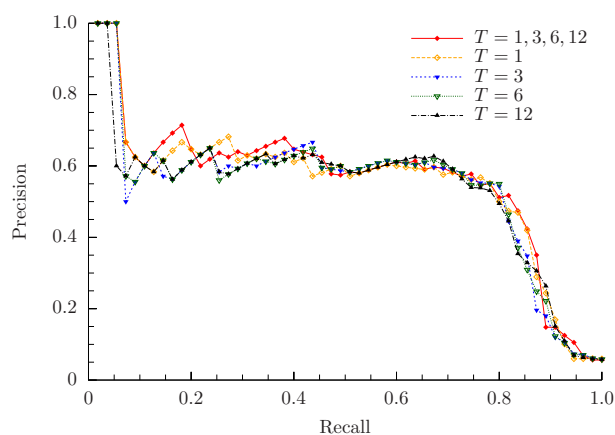


Figure 6: Comparison with the methods using only a single value of T for constancy classification.

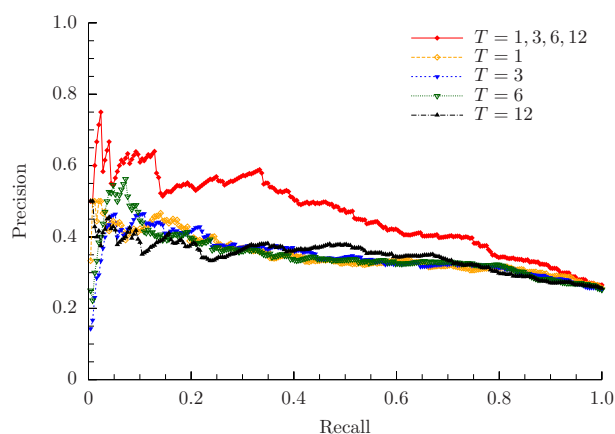


Figure 7: Comparison with the methods using only a single value of T for uniqueness classification.

5.4 Investigation into the window size, T

Our method uses multiple time windows of different sizes (i.e., different values of T) to induce features, as detailed in sections 3.1 and 4.1. To confirm the effect of this technique, we investigated the performance when we use only a single value of T (Figures 6 and 7).

The results in the uniqueness classification task demonstrated that our method achieves better overall results than the methods using a single value of T . We can therefore consider that using multiple values of T as features is a reasonable strategy. On the other hand, we could not confirm the effect of using multiple time windows of different sizes in the constancy classification task.

5.5 Error analysis

We randomly selected and analyzed 200 misclassified relations for both tasks. The analysis revealed four types of errors.

Paraphrases We observed that constant relations are prone to be miss-classified as non-constant when more than one paraphrase appear in arg2 and thus the value of arg2 is pretended to change. For example, America was also referred to as USA or United States of America. A similar problem was observed for unique relations as well.

Topical bias Topics mentioned in the blog posts are sometimes biased, and such bias can have a negative effect on classification, especially when a relation takes a small number of entity types in arg2 for given arg1 . For example, Jaden Smith, who is one of Will Smith’s sons, is frequently mentioned in our time-series text because he co-starred with his father in a movie, while Will Smith’s other sons never appeared in our text. We consider this a possible reason for our method wrongly identifying $\langle \text{arg1}$ ’s son is $\text{arg2} \rangle$ as a unique relation.

Short-/Long-term evolution Since we have aggregated on a monthly basis the 6-year’s worth of blog posts, the induced features cannot capture evolutions that occur in shorter or longer intervals. For example, consider relation $\langle \text{arg1}$ beats $\text{arg2} \rangle$ taking Real Madrid as arg1 . Since Real Madrid usually have more than one football match in a month, they can beat several teams in a month, which misleads the classifier to recognize the relation as non-unique. Similarly when a relation takes more than 6 years to evolve, it will be regarded as constant.

Reference to past, future, or speculative facts

The blog authors sometimes refer to relations that do not occur around when they write their posts; such relations actually occurred in the past, will occur in the future, or even speculative. Since our method exploits the time stamps attached to the posts to associate the relations with time, those relations introduce noises in the frequency distributions. Although our robust feature induction could in most cases avoid an adverse effect caused by these noises, they sometimes led to misclassification.

6 Related Work

In recent years, much attention has been given to extracting relations from a massive amount of textual data, especially the web (cf. section 1). Most of those studies, however, explored just extracting relations from text. Only a few studies, as described below, have discussed classifying those relations.

There has been no previous work on identifying the constancy of relations. The most relevant research topic is the temporal information extraction (Verhagen et al., 2007; Verhagen et al., 2010; Ling and Weld, 2010; Wang et al., 2010; Hovy et al., 2012). This is the task of extracting from textual data an event and the time it happened, e.g., Othello was written by Shakespeare in 1602. Such temporal information alone is not sufficient for identifying the constancy of relations, while we think it would be helpful.

On the other hand, the uniqueness of relations has so far been discussed in some studies. Ritter et al. (2008) have pointed out the importance of identifying unique relations for various NLP tasks such as contradiction detection, quantifier scope disambiguation, and synonym resolution. They proposed an EM-style algorithm for scoring the uniqueness of relations. Lin et al. (2010) also proposed three algorithms for identifying unique relations. While those studies discussed the same problem as this paper, they did not point out the importance of the constancy in identifying unique relations (cf. section 4.1).

7 Conclusion

This paper discussed that the notion of constancy is essential in compiling relations between entities extracted from real-world text and proposed a method for classifying relations on the basis of constancy and uniqueness. The time-series web text was fully exploited to induce frequency-based features from time-series frequency distribution on relation instances as well as language-based features tailored for individual classification tasks. Experimental results confirmed that the frequency-based features contributed much to the precision and recall in both identification tasks.

We will utilize the identified properties of the relations to adopt an appropriate strategy to compile

their instances. We also plan to start a spin-off research that acquires paraphrases by grouping values of *arg2s* for each value of *arg1* in a constant, unique relation.

We consider that the notion of constancy will even be beneficial in acquiring world knowledge, other than relations between entities, from text; we aim at extending the notion of constancy to other types of knowledge involving real-world entities, such as concept-instance relations.

Acknowledgments

This work was supported by the Multimedia Web Analysis Framework towards Development of Social Analysis Software program of the Ministry of Education, Culture, Sports, Science and Technology, Japan. The authors thank the annotators for their hard work. The authors are also indebted to the three anonymous reviewers for their valuable comments.

References

- Michele Banko, Michael J. Cafarella, Stephen Soderland, Matt Broadhead, and Oren Etzioni. 2007. Open information extraction from the web. In *Proceedings of IJCAI*, pages 2670–2676.
- Koby Crammer, Ofer Dekel, Joseph Keshet, Shai Shalev-Shawartz, and Yoram Singer. 2006. Online passive-aggressive algorithms. *Journal of Machine Learning Research*, 7:551–583.
- David Ferrucci, Eric Brown, Jennifer Chu-Carroll, James Fan, David Gondek, Aditya A. Kalyanpur, Adam Lally, J. William Murdock, Eric Nyberg, John Prager, Nico Schlaefter, and Chris Welty. 2010. Building Watson: An overview of the DeepQA project. *AI Magazine*, 31(3):59–79.
- Joseph L. Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378–382.
- Dirk Hovy, James Fan, Alfio Gliozzo, Siddharth Patwardhan, and Christopher Welty. 2012. When did that happen? — linking events and relations to timestamps. In *Proceedings of EACL*, pages 185–193.
- Richard J. Landis and Gary G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 1(33):159–174.
- Thomas Lin, Mausam, and Oren Etzioni. 2010. Identifying functional relation in web text. In *Proceedings of EMNLP*, pages 1266–1276.

- Xiao Ling and Daniel S. Weld. 2010. Temporal information extraction. In *Proceedings of AAAI*, pages 1385–1390.
- Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of ACL-IJCNLP*, pages 1003–1011.
- Patrick Pantel and Marco Pennacchiotti. 2006. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Proceedings of ACL*, pages 113–120.
- Alan Ritter, Doug Downey, Stephen Soderland, and Oren Etzioni. 2008. It’s a contradiction—no, it’s not: A case study using functional relations. In *Proceedings of EMNLP*, pages 11–20.
- Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. YAGO: A core of semantic knowledge unifying WordNet and Wikipedia. In *Proceedings of WWW*, pages 697–706.
- Marc Verhagen, Robert Gaizauskas, Frank Schilder, Mark Hepple, Graham Katz, and James Pustejovsky. 2007. SemEval-2007 task 15: TempEval temporal relation identification. In *Proceedings of SemEval*, pages 75–80.
- Marc Verhagen, Roser Sauri, Tommaso Caselli, and James Pustejovsky. 2010. SemEval-2010 task 13: TempEval-2. In *Proceedings of SemEval*, pages 57–62.
- Yafang Wang, Mingjie Zhu, Lizhen Qu, Marc Spaniol, and Gerhard Weikum. 2010. Timely YAGO: harvesting, querying, and visualizing temporal knowledge from Wikipedia. In *Proceedings of EDBT*, pages 697–700.
- Gerhard Weikum, Srikanta Bedathur, and Ralf Schenkel. 2011. Temporal knowledge for timely intelligence. In *Proceedings of BIRTE*, pages 1–6.
- Fei Wu and Daniel S. Weld. 2010. Open information extraction using Wikipedia. In *Proceedings of ACL*, pages 118–127.
- Fei Wu, Raphael Hoffmann, and Daniel S. Weld. 2008. Information extraction from Wikipedia: moving down the long tail. In *Proceedings of KDD*, pages 731–739.
- Naoki Yoshinaga and Masaru Kitsuregawa. 2009. Polynomial to linear: Efficient classification with conjunctive features. In *Proceedings of EMNLP*, pages 1542–1551.
- Naoki Yoshinaga and Masaru Kitsuregawa. 2010. Kernel slicing: Scalable online training with conjunctive features. In *Proceedings of COLING*, pages 1245–1253.
- Jun Zhu, Zaiqing Nie, Xiaojiang Liu, Bo Zhang, and Ji-Rong Wen. 2009. StatSnowball: a statistical approach to extracting entity relationships. In *Proceedings of WWW*, pages 101–110.