Relation Acquisition using Word Classes and Partial Patterns

Stijn De Saeger[†*] **Kentaro Torisawa**[†] **Masaaki Tsuchida**[§] **Jun'ichi Kazama**[†]
Chikara Hashimoto[†] **Ichiro Yamada**[‡] **Jong Hoon Oh**[†] **István Varga**[†] **Yulan Yan**[†]

[†] Information Analysis Laboratory, National Institute of
Information and Communications Technology, 619-0289 Kyoto, Japan
{stijn,torisawa,kazama,ch,rovellia,istvan,yulan}@nict.go.jp

[§] Information and Media Processing Laboratories, NEC Corporation, 630-0101 Nara, Japan
m-tsuchida@cq.jp.nec.com

[‡] Human & Information Science Research Division,
NHK Science & Technology Research Laboratories, 157-8510 Tokyo, Japan
yamada.i-hy@nhk.or.jp

Abstract

This paper proposes a semi-supervised relation acquisition method that does not rely on extraction patterns (e.g. "*X causes Y*" for causal relations) but instead learns a combination of indirect evidence for the target relation — *semantic word classes* and *partial patterns*. This method can extract long tail instances of semantic relations like causality from rare and complex expressions in a large Japanese Web corpus — in extreme cases, patterns that occur only once in the entire corpus. Such patterns are beyond the reach of current pattern based methods. We show that our method performs on par with state-of-the-art pattern based methods, and maintains a reasonable level of accuracy even for instances acquired from infrequent patterns. This ability to acquire long tail instances is crucial for risk management and innovation, where an exhaustive database of high-level semantic relations like causation is of vital importance.

1 Introduction

Pattern based relation acquisition methods rely on *lexico-syntactic patterns* (Hearst, 1992) for extracting relation instances. These are templates of natural language expressions such as "X causes Y" that signal an instance of some semantic relation (i.e., causality). Pattern based methods (Agichtein and Gravano, 2000; Pantel and Pennacchiotti, 2006b; Paşca et al., 2006; De Saeger et al., 2009) learn many such patterns to extract new instances (word pairs) from the corpus.

However, since extraction patterns are learned using statistical methods that require a certain frequency of observations, pattern based methods fail to capture relations from complex expressions in which the pattern connecting the two words is rarely observed. Consider the following sentence:

> "Curing hypertension alleviates the deterioration speed of the renal function, thereby lowering the risk of causing intracranial bleeding"

Humans can infer that this sentence expresses a causal relation between the underlined noun phrases. But the actual *pattern* connecting them, i.e., "*Curing X alleviates the deterioration speed of the renal function, thereby lowering the risk of causing Y*", is rarely observed more than once even in a 10^8 page Web corpus. In the sense that the term *pattern* implies a recurring event, this expression contains no pattern for detecting the causal relation between *hypertension* and *intracranial bleeding*. This is what we mean by "long tail instances" — words that co-occur infrequently, and only in sparse extraction contexts.

Yet an important application of relation extraction is mining the Web for so-called *unknown unknowns* — in the words of D. Rumsfeld, "things we don't know we don't know" (Torisawa et al., 2010). In knowledge discovery applications like risk management and innovation, the usefulness of relation extraction lies in its ability to find many unexpected remedies for diseases, causes of social problems, and so on. To give an example, our relation extrac-

[*] This work was done when all authors were at the National Institute of Information and Communications Technology.

Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, pages 825–835,
Edinburgh, Scotland, UK, July 27–31, 2011.

tion system found a blog post mentioning Japanese automaker Toyota as a hidden cause of Japan's deflation. Several months later the same connection was made in an article published in an authoritative economic magazine.

We propose a semi-supervised relation extraction method that does not rely on direct pattern evidence connecting the two words in a sentence. We argue that the role of binary patterns can be replaced by a combination of two types of *indirect* evidence: *semantic class information* about the target relation and *partial patterns*, which are *fragments* or *subpatterns* of binary patterns. The intuition is this: if a sentence like the example sentence above contains some word X belonging to the class of *medical conditions* and another word Y from the class of *traumas*, and X matches the partial pattern "*. . . causing* X", there is a decent chance that this sentence expresses a causal relation between X and Y. We show that just using this combination of indirect evidence we can pick up semantic relations with roughly 50% precision, regardless of the complexity or frequency of the expression in which the words co-occur. Furthermore, by combining this idea with a straightforward machine learning approach, the overall performance of our method is on par with state-of-the-art pattern based methods. However, our method manages to extract a large number of instances from sentences that contain no pattern that can be learned by pattern induction methods.

Our method is a two-stage system. Figure 1 presents an overview. In Stage 1 we apply a state-of-the-art pattern based relation extractor to a Web corpus to obtain an initial batch of relation instances. In Stage 2 a supervised classifier is built from various components obtained from the output of Stage 1. Given the output of Stage 1 and access to a Web corpus, the Stage 2 extractor is completely self-sufficient, and the whole method requires no supervision other than a handful of seed patterns to start the first stage extractor. The whole procedure is therefore minimally supervised. Semantic word classes and partial patterns play a crucial role throughout all steps of the process.

We evaluate our method on three relation acquisition tasks (*causation*, *prevention* and *material* relations) using a 600 million Japanese Web page corpus (Shinzato et al., 2008) and show that our system can successfully acquire relations from both frequent and infrequent patterns. Our system extracted 100,000 causal relations with 84.6% precision, 50,000 prevention relations with 58.4% precision and 25,000 material relations with 76.1% precision. In the extreme case, we acquired several thousand word pairs co-occurring only in patterns that appear once in the entire corpus. We call such patterns *single occurrence* (SO) patterns. Word pairs that co-occur only with SO patterns represent the theoretical limiting case of relations that cannot be acquired using existing pattern based methods. In this sense our method can be seen as complementary with pattern based approaches, and merging our method's output with that of a pattern based method may be beneficial.

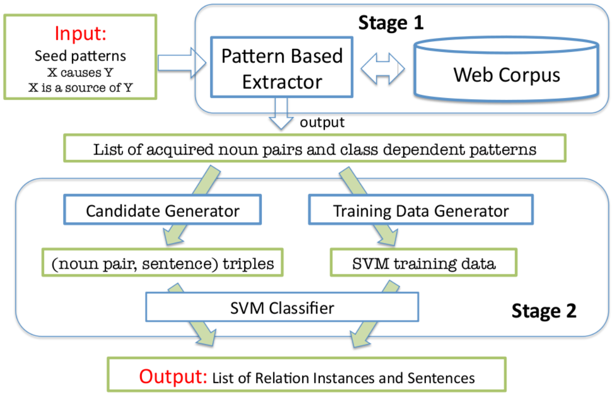

Figure 1: Proposed method: data flow.

2 Stage 1 Extractor

This section introduces our Stage 1 extractor: the pattern based method from (De Saeger et al., 2009), which we call CDP for "*class dependent patterns*". We give a brief overview below, and refer the reader to the original paper for a more comprehensive explanation.

CDP takes a set of *seed patterns* as input, and automatically learns new *class dependent patterns* as paraphrases of the seed patterns. Class dependent patterns are semantic class restricted versions of ordinary lexico-syntactic patterns. Existing methods use class *independent* patterns such as "X causes Y" to learn causal relations between X and Y. Class dependent patterns however place semantic class restrictions on the noun pairs they may extract, like "$Y_{accidents}$ *causes* $X_{incidents}$". The *accidents* and *incidents* subscripts specify the semantic class of the X and Y slot fillers.

These class restrictions make it possible to distinguish between multiple senses of highly ambiguous patterns (so-called "generic" patterns). For instance, given the generic pattern "Y *by* X", if we restrict Y and X in to the semantic classes of *injuries* and *accidents* (as in "*death by drowning*"), the class dependent pattern "$Y_{injuries}$ *by* $X_{accidents}$" becomes a valid paraphrase of "X *causes* Y" and can safely be used to extract causal relations, whereas other class dependent versions of the same generic pattern (e.g., "$Y_{products}$ *by* $X_{companies}$", as in "*iPhone by Apple*") may not.

CDP ranks each noun pair in the corpus according to a score that reflects its likelihood of being a proper instance of the target relation, by calculating the semantic similarity of a set of seed patterns to the class dependent patterns this noun pair co-occurs with. The output of CDP is a list of noun pairs ranked by score, together with the highest scoring class dependent pattern each noun pair co-occurs with. This list becomes the input to Stage 2 of our method, as shown in Figure 1. We adopted CDP as Stage 1 extractor because, besides having generally good performance, the class dependent patterns provide the two fundamental ingredients for Stage 2 of our method — the target semantic word classes for a given relation (in the form of the semantic class restrictions attached to patterns), and partial patterns.

To obtain fine-grained semantic word classes we used the large scale word clustering algorithm from (Kazama and Torisawa, 2008), which uses the EM algorithm to compute the probability that a word w belongs to class c, i.e., $P(c|w)$. Probabilistic clustering defines no discrete boundary between members and non-members of a semantic class, so we simply assume w belongs to c whenever $P(c|w) \geq 0.2$. For this work we clustered 10^6 nouns into 500 classes.

Finally, we adopt the structural representation of patterns introduced in (Lin and Pantel, 2001). All sentences in our corpus are dependency parsed, and patterns consist of words on the path of dependency relations connecting two nouns.

3 Stage 2 Extractor

We use CDP as our Stage 1 extractor, and the top N noun pairs along with the class dependent patterns that extract them are given as input to Stage 2, which represents the main contribution of this work. As shown in Figure 1, Stage 2 consists of three modules: a *candidate generator*, a *training data generator* and a *supervised classifier*. The training data generator builds training data for the classifier from the top N output of CDP and sentences retrieved from the Web corpus. This classifier then scores and ranks the candidate relations generated by the candidate relation generator. We introduce each module below.

Candidate Generator This module generates sentences containing candidate word pairs for the target relation from the corpus. It does so using the semantic class restrictions and partial patterns obtained from the output of CDP. The set of all semantic class pairs obtained from the class dependent patterns that extracted the top N results become the target semantic class pairs from which new candidate instances are generated. We extract all sentences containing a word pair belonging to one of the target class pairs from the corpus.

From these sentences we keep only those that contain a trace of evidence for the target semantic relation. For this we decompose the class dependent patterns from the Stage 1 extractor into *partial patterns*. As mentioned previously, patterns consist of words on the path of dependency relations connecting the two target words in a syntactic tree. To obtain partial patterns we split this dependency path into its two constituent branches, each one leading from the leaf word (i.e. variable) to the syntactic head of the pattern. For example, "$X \xleftarrow{\text{subj}}$ *causes* $\xrightarrow{\text{obj}} Y$" is split into two partial patterns "$X \xleftarrow{\text{subj}}$ *causes*" and "*causes* $\xrightarrow{\text{obj}} Y$". These partial patterns capture the predicate structures in binary patterns.[1] We discard partial patterns with syntactic heads other than verbs or adjectives.

The candidate genarator retrieves all sentences from the corpus in which two nouns belonging to one of the target semantic classes co-occur and

[1] In Japanese, case information is encoded in post-positions attached to the noun.

where at least one of the nouns matches a partial pattern. As shown in Figure 1, these sentences and the candidate noun pairs they contain (called *(noun pair, sentence)* triples hereafter) are submitted to the classifier for scoring. Restricting candidate noun pairs by this combination of semantic word classes and partial pattern matching proved to be quite powerful. For instance, in the case of causal relations we found that close to 60% of the *(noun pair, sentence)* triples produced by the candidate generator were correct (Figure 6).

Training Data Generator As shown in Figure 1, the *(noun pair, sentence)* triples used as training data for the SVM classifier were generated from the top results of the Stage 1 extractor and the corpus. We consider the noun pairs in the top N output of the Stage 1 extractor as *true* instances of the target relation (even though they may contain erroneous extractions), and retrieve from the corpus all sentences in which these noun pairs co-occur and that match one of the partial patterns mentioned above. In our experiments we set N to $25,000$. We randomly select positive training samples from this set of *(noun pair, sentence)* triples.

Negative training samples are also selected randomly, as follows. If one member of the target noun pair in the positive samples above matches a partial pattern but the other does not, we randomly replace the latter by another noun found in the same sentence, and generate this new *(noun pair, sentence)* triple as a negative training sample. In the causal relation experiments this approach had about 5% chance of generating *false* negatives — noun pairs contained in the top N results of the Stage 1 extractor. Such samples were discarded. Our experimental results show that this scheme works quite well in practice. We randomly sample M positive and negative samples from the autogenerated training data to train the SVM. M was empirically set to 50,000 in our experiments.

SVM Classifier We used a straightforward feature set for training the SVM classifier. Because our classifier will be faced with sentences containing long and infrequent patterns where the distance between the two target nouns may be quite large, we did not try to represent lexico-syntactic patterns as features but deliberately restricted the feature set to local context features of the candidate noun pair in the target sentence. Concretely, we looked at bigrams and unigrams surrounding both nouns of the candidate relation, as the local context around the target words may contain many telling expressions like "*increase in X*" or "*X deficiency*" which are useful clues for causal relations. Also, in Japanese case information is encoded in post-positions attached to the noun, which is captured by the unigram features.

In addition to these base features, we include the semantic classes to which the candidate noun pair belongs, the partial patterns they match in this sentence, and the infix words inbetween the target noun pair. Note that this feature set is not intended to be optimal beyond the actual claims of this paper, and we have deliberately avoided exhaustive feature engineering so as not to obscure the contribution of semantic classes and partial pattern to our approach. Clearly an *optimal* classifier will incorporate many more advanced features (see GuoDong et al. (2005) for a comprehensive overview), but even without sophisticated feature engineering our classifier achieved sufficient performance levels to support our claims. An overview of the feature set is given in Table 1. The relative contribution of each type of features is discussed in section 4. In preliminary experiments we found a polynomial kernel of degree 3 gave the best results, which suggests the effectiveness of combining different types of indirect evidence.

The SVM classifier outputs *(noun pair, sentence)* triples, ranked by SVM score. To obtain the final output of our method we assign each unique noun pair the maximum score from all *(noun pair, sentence)* triples it occurs in, and discard all other sentences for this noun pair. In section 4 below we evaluate the acquired noun pairs in the context of the sentence that maximizes their score.

4 Evaluation

We demonstrate the effectiveness of semantic word classes and partial pattern matching for relation extraction by showing that the method proposed in this paper performs at the level of other state-of-the-art relation acquisition methods. In addition we demonstrate that our method can successfully extract relation instances from infrequent patterns, and we

Feature type	Description	Number of features
Morpheme features	Unigram and bigram morphemes surrounding both target words.	554,395
POS features	Coarse- and fine-grained POS tags of the noun pair and morpheme features.	2,411
Semantic features	Semantic word classes of the target noun pair.	1000 (500 classes $\times 2$)
Infix word features	Morphemes found inbetween the target noun pair.	94,448
Partial patterns	Partial patterns matching the target noun pair.	86

Table 1: Feature set used in the Stage 2 classifier, and their number for the causal relation experiments.

explore several criteria for what constitutes an infrequent pattern — including the theoretical limiting case of patterns observed only once in the entire corpus. These instances are impossible to acquire by pattern based methods. The ability to acquire relations from extremely infrequent expressions with decent accuracy demonstrates the utility of combining semantic word classes with partial pattern matching.

4.1 Experimental Setting

We evaluate our method on three semantic relation acquisition tasks: *causality*, *prevention* and *material*. Two concepts stand in a *causal relation* when the source concept (the "cause") is directly or indirectly responsible for the subsequent occurrence of the target concept (its "effect"). In a *prevention* relation the source concept directly or indirectly acts to avoid the occurrence of the target concept, and in a *material* relation the source concept is a material or ingredient of the target concept.

For our experiments we used the latest version of the TSUBAKI corpus (Shinzato et al., 2008), a collection of 600 million Japanese Web pages dependency parsed by the Japanese dependency parser KNP[2]. In our implementation of CDP, lexicosyntactic patterns consist of words on the path connecting two nouns in a dependency parse tree. We discard patterns from dependency paths longer than 8 constituent nodes. Furthermore, we estimated pattern frequencies in a subset of the corpus (50 million pages, or 1/12th of the entire corpus) and discarded patterns that co-occur with less than 10 unique noun pairs in this smaller corpus. These restrictions do not apply to the proposed method, which can extract noun pairs connected by patterns of arbitrary length, even if found only once in the corpus. For our purpose we treat dependency paths whose observed frequency is below this threshold as insufficiently frequent to be considered as "patterns". This threshold is of course arbitrary, but in section 4 we show that our results are not affected by these implementation details.

We asked three human judges to evaluate random *(noun pair, sentence)* triples, i.e. candidate noun pairs in the context of some corpus sentence in which they co-occur. If the judges find the sentence contains sufficient evidence that the target relation holds between the candidate nouns, they mark the noun pair correct. To evaluate the performance of each method we use two evaluation criteria: *strict* (all judges must agree the candidate relation is correct) and *lenient* (decided by the judges' majority vote). Over all experiments the interrater agreement (Kappa) ranged between 0.57 and 0.82 with an average of 0.72, indicating substantial agreement (Landis and Koch, 1977).

4.1.1 Methods Compared

We compare our results to two pattern based methods: CDP (the Stage 1 extractor) and *Espresso* (Pantel and Pennacchiotti, 2006a).

Espresso is a popular bootstrapping based method that uses a set of seed instances to induce extraction patterns for the target relation and then acquire new instances in an iterative bootstrapping process. In each iteration Espresso performs pattern induction, pattern ranking and selection using previously acquired instances, and uses the newly acquired patterns to extraction new instances. Espresso computes a reliability score for both instances and patterns based on their pointwise mutual information (PMI) with the top-scoring patterns and instances from the previous iteration.[3] We refer to (Pantel and

[2] http://nlp.kuee.kyoto-u.ac.jp/nl-resource/knp.html

[3] In our implementation of Espresso we found that, despite the many parameters for controlling the bootstrapping process,

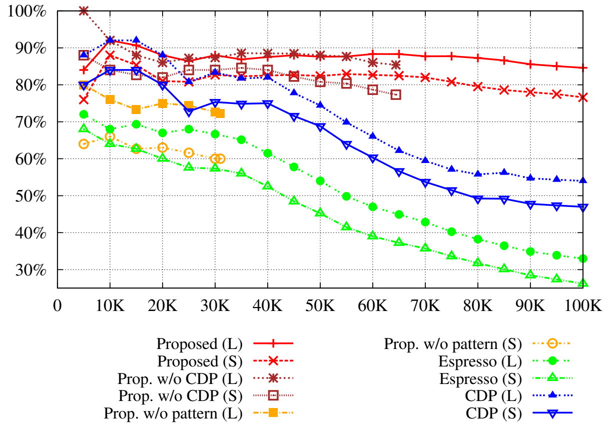

Figure 2: Precision of acquired relations (causality). L and S denote lenient and strict evaluation.

Pennacchiotti, 2006a) for a more detailed description.

For all methods compared we rank the acquired noun pairs by their score and evaluated 500 random samples from the top 100,000 results. For noun pairs acquired by CDP and Espresso we select the pattern that extracted this noun pair (in the case of Espresso, the pattern with the highest PMI for this noun pair), and randomly select a sentence in which the noun pair co-occurs with that pattern from our corpus. For the proposed method we evaluate noun pairs in the context of the *(noun pair, sentence)* triple with the highest SVM score.

4.2 Results and Discussion

The performance of each method on the causality, prevention and material relations are shown in Figures 2, 3 and 4 respectively. In the causality experiments (Figure 2) the proposed method performs on par with CDP for the top 25,000 results, both achieving close to 90% precision. But whereas CDP's performance drops from there our method maintains the same high precision throughout (84.6%, lenient). Both our method and CDP outperform Espresso by a large margin.

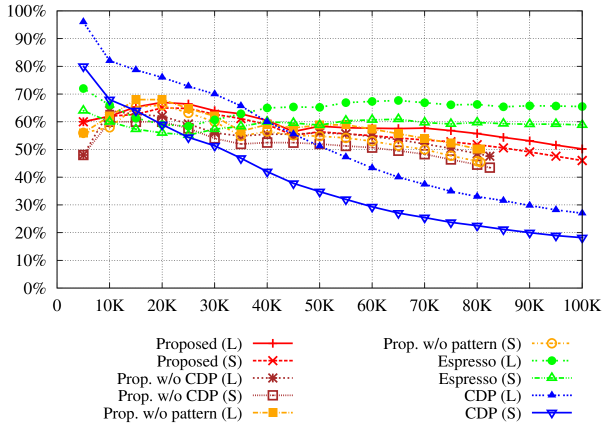

Figure 3: Precision of acquired relations (prevention). L and S denote lenient and strict evaluation.

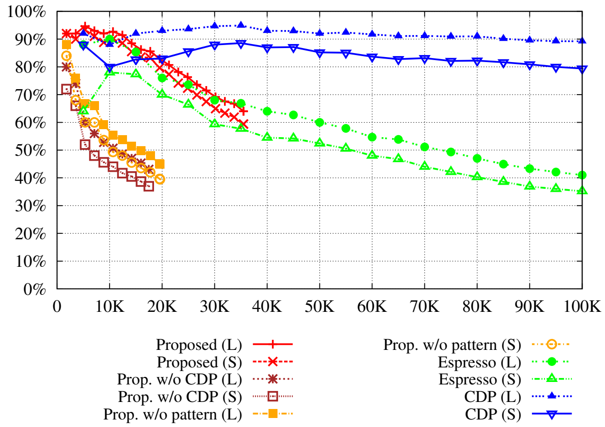

Figure 4: Precision of acquired relations (material). L and S denote lenient and strict evaluation.

For the prevention relation (Figure 3), precision is considerably lower for all methods except the top 10,000 of CDP (82% precision, lenient). The proposed method peaks at around 20,000 results (67% precision, lenient) and performance remains more or less constant from there on. The proposed method overtakes CDP's performance around the top 45,000 mark, which suggests that combining the results of both methods may be beneficial.

In the material relations the proposed method slightly outperforms both pattern based methods in the top 10,000 results (92% precision, lenient).

it remains very difficult to prevent *semantic drift* (Komachi et al., 2008) from occurring. One small adjustment to the algorithm stabilized the bootstrapping process considerably and gave overall better results. In the pattern induction step (section 3.2 in (Pantel and Pennacchiotti, 2006a)), Espresso computes a reliability score for each candidate pattern based on the weighted PMI of the pattern with all instances extracted so far. As the number of extracted instances increases this disproportionally favours high frequency (i.e. generic) patterns, so instead of using *all* instances for computing pattern reliability we only use the m most reliable instances from the previous iteration, which were used to extract the candidate patterns of the current iteration ($m = 200$, like the original).

However for this relation our method produced only 35,409 instances. The reason is that the top 25,000 results of CDP were all extracted by just 12 patterns, and these contained many patterns whose syntactic head is not a verb or adjective (like "*Y rich in X*" or "*Y containing X*"). Only 12 partial patterns were obtained, which greatly reduced the output of the proposed method. Taking into account the high performance of CDP for material relations, this suggests that for some relations our method's N and M parameters could use some tuning. In conclusion, in all three relations our method performs at a level comparable to state-of-the-art pattern based methods, which is remarkable given that it only uses indirect evidence.

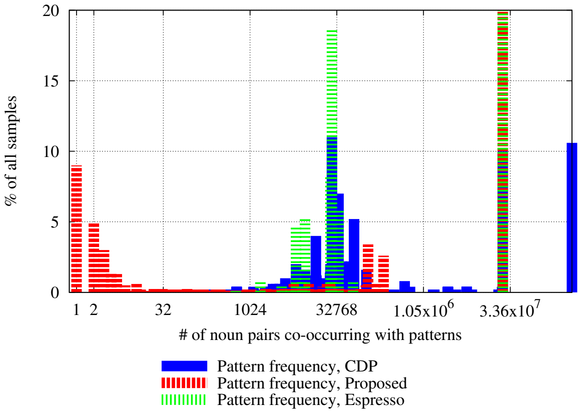

Figure 5: Frequencies of patterns in the evaluation data (causation).

Dealing with Difficult Extractions How does our method handle noun pairs that are difficult to acquire by pattern based methods? The graphs marked "Prop. w/o CDP" (Proposed without CDP) in Figures 2 , 3 and 4 show the number and precision of evaluated samples from the proposed method that do not co-occur in our corpus with any of the patterns that extracted the top N results of the first stage extractor. These graphs show that our method is not simply regenerating CDP's top results but actually extracts many noun pairs that do not co-occur in patterns that are easily learned. Figure 2 shows that roughly two thirds of the evaluated samples are in this category, and that their performance is not significantly worse than the overall result. The same conclusion holds for the prevention results (Figure 3), where over 80% of the proposed method's samples are noun pairs that do not co-occur with easily learnable patterns. Their precision is about 5% worse than all samples from the proposed method. For material relations (Figure 4) about half of all evaluated samples are in this category, but their precision is markedly worse compared to all results.

For genuinely *infrequent* patterns, the graphs marked "Prop. w/o pattern" (Proposed without pattern) in Figures 2 , 3 and 4 show the number and precision of noun pairs evaluated for the proposed method that were acquired from sentences without any discernible pattern. As explained in section 4 above, these constitute noun pairs co-occurring in a sentence in which the path of dependency relations connecting them is either longer than 8 nodes or can extract fewer than 10 noun pairs in 50 million Web pages. Note that in theory it is possible that these noun pairs could not be acquired by pattern based methods due to this threshold — patterns must be able to extract more than 10 different noun pairs in a subset of our corpus, while the proposed method does not have this constraint. So at least in theory, pattern based methods might be able to acquire all noun pairs obtained by our method by lowering this threshold. To see that this is unlikely to be the case, consider Figure 5, which shows the pattern frequency of the patterns induced by CDP and Espresso for the causality experiment. The x-axis represents pattern frequency in terms of the number of unique noun pairs a pattern co-occurs with in our corpus (on a log scale), and the y-axis shows the percentage of samples that was extracted by patterns of a given frequency.[4] Figure 5 shows that for the pattern based methods, the large majority of noun pairs was extracted by patterns that co-occur with several thousand different noun pairs. Extrapolating the original frequency threshold of 10 nounpairs to the size of our entire corpus roughly corresponds to about 120 distinct noun pairs (10 times in 1/12th of the entire corpus). In Figure 5, the histograms for the pattern based methods CDP and Espresso start around 1000 noun pairs, which is far above this new lowerbound.

[4] In the case of CDP we ignore semantic class restrictions on the patterns when comparing frequencies. For Espresso, the most frequent pattern ("Y by X" at the 24,889,329 data point on the x-axis) extracted up to 53.8% of the results, but the graph was cut at 20% for readability.

Causality	⟨ カテコラミン ⟩ が心拍数の急上昇をもたらすことから、血管内の血行状態の変化が血管内障害につながり、[血栓形成] を促進する。 Because ⟨catecholamine⟩ causes a rapid increase of heart rate, the change of circulation inside the blood vessels leads to blood vessel disorders and promotes [thrombus generation].
	⟨ 頻脈発作 ⟩ が始まってキシロカインを静注したところ、患者さんが突然意識をなくして [全身痙攣] を起こしたということです。 When we injected Xylocaine during a ⟨tachycardia seizure⟩, the patient suddenly lost consciousness and fell into a fit of [convulsions].
	その理由としては ⟨ 動物性たんぱく質 ⟩ を多く取ることで [わきが] の原因物質が増加するからです。 (. . .) The reason is that by taking a lot of ⟨animal proteins⟩ the causative agents of [tragomaschalia] increase.
	* 抗酸化機能を高め、老化や ⟨ 生活習慣病 ⟩ の原因となる活性化酸素を消去する作用がある [ラドン]。 * [Radon] heightens the (body's) antioxidative function and is effective for eliminating activated oxygen, which is a cause of aging and ⟨lifestyle-related⟩ diseases.
Prevention	マグロのトロ、中トロの部分には、ＤＨＡや ⟨ ＥＰＡ ⟩ が豊富に含まれているので、[神経痛] の予防食材として有効です。 Because the fatty meat of tuna contains DHA and ⟨EPA⟩ in abundance, it is effective for preventing [neuralgia].
	空気の代わりに ⟨ 窒素ガス ⟩ を使用すれば [粉塵爆発] の予防になります。 If you use ⟨nitrogen gas⟩ instead of air you may prevent [dust explosions].
	ヨーロッパでは古くから ⟨ クミスクチン ⟩ のお茶は「やせるお茶である」といわれ中性脂肪や [成人病] の予防に良いそうです。 In ancient Europe ⟨orthosiphon aristatus⟩ tea was called a "diet tea", and supposedly it helps preventing triglycerides and [adult diseases].
	* ⟨ 幌 ⟩ の収納時にスクリーンの間に挟んで、スクリーンの [スリキズ] を防ぐものです。 * (It) is something that prevents [scratches] on the screen if the ⟨calash⟩ gets stuck between the screens during storage.

Table 2: Causality and Prevention relations acquired from Single Occurrence (SO) patterns. ⟨X⟩ and [Y] indicate the relation instance's source and target words, and "*" indicates erroneous extractions.

Thus, pattern based methods naturally tend to induce patterns that are much more frequent than the range of patterns our method can capture, and it is unlikely that this is a result of implementation details like pattern frequency threshold.

The precision of noun pairs in the category "Prop. w/o pattern" is clearly lower than the overall results, but the graphs demonstrate that our method still handles these difficult cases reasonably well. The 500 samples evaluated contained 155 such instances for causality, 403 for prevention and 276 for material. For prevention, the high ratio of these noun pairs helps explain why the overall performance was lower than for the other relations.

Finally, the theoretical limiting case for pattern based algorithms consists of patterns that only co-occur with a single noun pair in the entire corpus (*single occurrence* or SO patterns). Pattern based methods learn new patterns that share many noun pairs with a set of reliable patterns in order to extract new relation instances. If a noun pair that co-occurs with a SO pattern also co-occurs with more reliable patterns there is no need to learn the SO pattern. If that same noun pair does *not* co-occur with any other reliable pattern, the SO pattern is beyond the reach of *any* pattern induction method. Thus, SO patterns are effectively useless for pattern based methods. For the 500 samples evaluated from the causality and prevention relations acquired by our method we found 7 causal noun pairs that co-occur only in SO patterns and 29 such noun pairs for prevention. The precision of these instances was 42.9% and 51.7% respectively. In total we found 8,716 causal noun pairs and 7,369 prevention noun pairs that co-occur only with SO patterns. Table 2 shows some example relations from our causality and prevention experiments that were extracted from SO patterns. To conclude, our method is able to acquire correct relations even from the most extreme infrequent expressions.

Semantic Classes, Partial Patterns or Both? In the remainder of this section we look at how the combination of semantic word classes and partial patterns benefits our method. For each relation we evaluated 1000 random *(noun pair, sentence)* triples satisfying the two conditions from section 3 — matching semantic class pairs and partial patterns. Surprisingly, the precision of these samples was 59% for causality, 40% for prevention and 50.4% for material, showing just how compelling these two types of indirect evidence are in combination.

To estimate the relative contribution of each heuristic we compared our candidate generation method against two baselines. The first baseline evaluates the precision of random noun pairs from

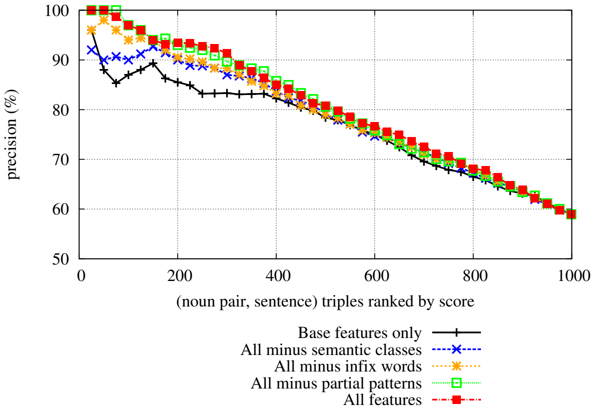

Figure 6: Contribution of feature sets (causality).

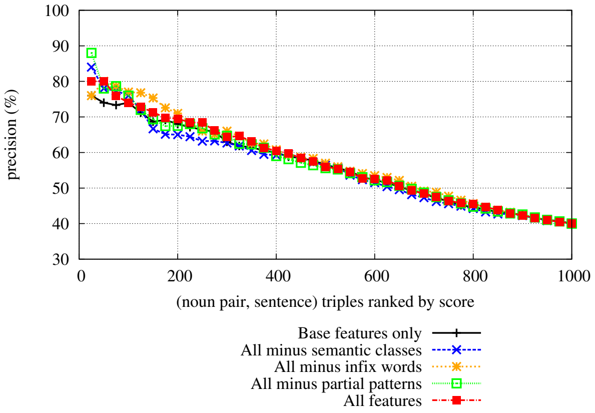

Figure 7: Contribution of feature sets (prevention).

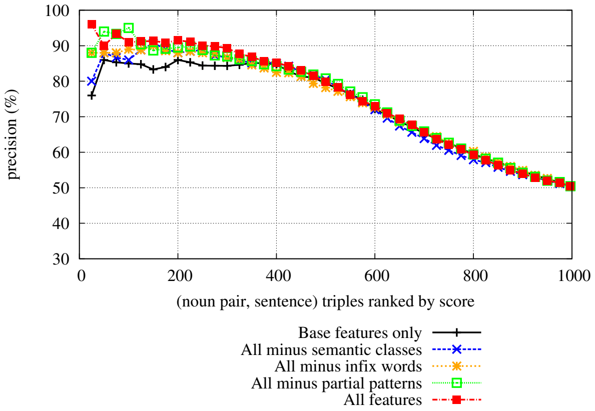

Figure 8: Contribution of feature sets (material).

the target semantic classes that co-occur in a sentence. The second baseline does the same for the second heuristic, selecting random sentences containing a noun pair that matches some partial pattern. Evaluating 100 samples for causality and prevention, we found the precision of the semantic class baseline was 16% for causality and 5% for prevention. The pattern fragment baseline gave 9% for causality and 22% for prevention. This is considerably lower than the precision of random samples that satisfy both the semantic class and partial pattern conditions, showing that the combination of semantic classes and partial patterns is more effective than either one individually.

Finally, we investigated the effect of the various feature sets used in the classifier. Figures 6, 7 and 8 show the results for the respective semantic relations. The "Base features" graph shows the performance the unigram, bigram and part-of-speech features. "All features" uses all features in Table 1. The other graphs show the effect of removing one type of features. These graphs suggest that the contribution of the individual feature types (semantic class information, partial patterns or infix words) to the classification performance is relatively minor, but in combination they do give a marked improvement over the base features, at least for some relations like causation and material. In other words, the main contribution of semantic word classes and partial patterns to our method's performance lies not in the final classification step but seems to occur at earlier stages of the process, in the candidate and training data generation steps.

5 Related Work

Using lexico-syntactic patterns to extract semantic relations was first explored by Hearst (Hearst, 1992), and has inspired a large body of work on semi-supervised relation acquisition methods (Berland and Charniak, 1999; Agichtein and Gravano, 2000; Etzioni et al., 2004; Pantel and Pennacchiotti, 2006b; Paşca et al., 2006; De Saeger et al., 2009), two of which were used in this work.

Some researchers have addressed the sparseness problems inherent in pattern based methods. Downey et al. (2007) starts from the output of the unsupervised information extraction system TextRunner (Banko and Etzioni, 2008), and uses language modeling techniques to estimate the reliability of sparse extractions. Paşca et al. (2006) alleviates pattern sparseness by using infix patterns that are generalized using classes of distributionally similar words. In addition, their method employs clustering based semantic similarities to filter newly extracted instances in each iteration of the bootstrapping process. A comparison with our method would have been instructive, but we were unable to implement their method because the original paper contains insufficient detail to allow replication.

There is a large body of research in the *supervised* tradition that does not use explicit pattern representations — kernel based methods (Zelenko et al., 2003; Culotta, 2004; Bunescu and Mooney, 2005) and CRF based methods (Culotta et al., 2006). These approaches are all fully supervised, whereas in our work the automatic generation of candidates and training data is an integral part of the method. An interesting alternative is distant supervision (Mintz et al., 2009), which trains a classifier using an existing database (Freebase) containing thousands of semantic relations, with millions of instances. We believe our method is more general, as depending on external resources like a database of semantic relations limits both the range of semantic relations (i.e., Freebase contains only relations between named entities, and none of the relations in this work) and languages (i.e., no resource comparable to Freebase exists for Japanese) to which the technology can be applied. Furthermore, it is unclear whether distant supervision can deal with noisy input such as automatically acquired relation instances.

Finally, inference based methods (Carlson et al., 2010; Schoenmackers et al., 2010; Tsuchida et al., 2010) are another attempt at relation acquisition that goes beyond pattern matching. Carlson et al. (2010) proposed a method based on inductive logic programming (Quinlan, 1990). Schoenmackers et al. (2010) takes relation instances produced by TextRunner (Banko and Etzioni, 2008) as input and induces first-order Horn clauses, and new instances are infered using a Markov Logic Network (Richardson and Domingo, 2006; Huynh and Mooney, 2008). Tsuchida et al. (2010) generated new relation hypotheses by substituting words in seed instances with distributionally similar words. The difference between these works and ours lies in the treatment of evidence. While the above methods learn inference rules to acquire new relation instances from independent information sources scattered across different Web pages, our method takes the other option of working with all the clues and indirect evidence a single sentence can provide. In the future, a combination of both approaches may prove beneficial.

6 Conclusion

We have proposed a relation acquisition method that is able to acquire semantic relations from infrequent expressions by focusing on the evidence provided by semantic word classes and partial pattern matching instead of direct extraction patterns. We experimentally demonstrated the effectiveness of this approach on three relation acquisition tasks, causality, prevention and material relations. In addition we showed our method could acquire a significant number of relation instances that are found in extremely infrequent expressions, the most extreme case of which are *single occurrence patterns*, which are beyond the reach of existing pattern based methods. We believe this ability is of crucial importance for acquiring valuable long tail instances. In future work we will investigate whether the current framework can be extended to acquire inter-sentential relations.

References

Eugene Agichtein and Luis Gravano. 2000. Snowball: extracting relations from large plain-text collections. In *Proc. of the fifth ACM conference on Digital libraries*, pages 85–94.

Michele Banko and Oren Etzioni. 2008. The tradeoffs between open and traditional relation extraction. In *Proc. of the 46th ACL-08:HLT*, pages 28–36.

Matthew Berland and Eugene Charniak. 1999. Finding parts in very large corpora. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 57–64, College Park, Maryland, USA, June.

Razvan C. Bunescu and Raymond J. Mooney. 2005. A shortest path dependency kernel for relation extraction. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing (HLT '05)*, pages 724–731.

Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka Jr., and Tom M. Mitchell. 2010. Toward an architecture for neverending language learning. In *Proc of the 24th AAAI*, pages 1306–1313.

Aron Culotta, Andrew McCallum, and Jonathan Betz. 2006. Integrating probabilistic extraction models and data mining to discover relations and patterns in text. In *Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (HLT/NAACL)*, pages 296–303.

Aron Culotta. 2004. Dependency tree kernels for relation extraction. In *In Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04*, pages 423–429.

Stijn De Saeger, Kentaro Torisawa, Jun'ichi Kazama, Kow Kuroda, and Masaki Murata. 2009. Large Scale Relation Acquisition Using Class Dependent Patterns. In *Proc. of the 9th International Conference on Data Mining (ICDM)*, pages 764–769.

Doug Downey, Stefan Schoenmackers, and Oren Etzioni. 2007. Sparse information extraction: Unsupervised language models to the rescue. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL2007)*.

Oren Etzioni, Michael Cafarella, Doug Downey, Stanley Kok, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel Weld, and Alexander Yates. 2004. Web-scale information extraction in KnowItAll. In *Proc. of the 13th international conference on World Wide Web (WWW04)*, pages 100–110.

Zhou GuoDong, Su Jian, Zhang Jie, and Zhang Min. 2005. Exploring various knowledge in relation extraction. In *Proc. of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 419–444.

Marti Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proc. of the 14th International Conference on Computational Linguistics (COLING'92)*, pages 539–545.

Tuyen N. Huynh and Raymond J. Mooney. 2008. Discriminative structure and parameter learning for markov logic networks. In *Proc. of the 25th ICML*, pages 416–423.

Jun'ichi Kazama and Kentaro Torisawa. 2008. Inducing gazetteers for named entity recognition by large-scale clustering of dependency relations. In *Proc. of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-08: HLT)*, pages 407–415.

Mamoru Komachi, Taku Kudo, Masashi Shimbo, and Yuji Matsumoto. 2008. Graph-based analysis of semantic drift in espresso-like bootstrapping algorithms. In *Proc. of EMNLP'08. Honolulu, USA*, pages 1011–1020.

Dekang Lin and Patrick Pantel. 2001. Dirt - discovery of inference rules from text. In *Proc. of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 323–328.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proc. of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011.

Marius Paşca, Dekang Lin, Jeffrey Bigham, Andrei Lifchits, and Alpa Jain. 2006. Names and Similarities on the Web: Fact Extraction in the Fast Lane. In *Proc. of the COLING-ACL06*, pages 809–816.

Patrick Pantel and Marco Pennacchiotti. 2006a. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Proc. of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (COLING-ACL-06*, pages 113–120.

Patrick Pantel and Pennacchiotti Pennacchiotti, Marco. 2006b. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Proc. of the COLING-ACL06*, pages 113–120.

J. R. Quinlan. 1990. Learning logical definitions from relations. *Machine Learning*, 5(3):239–266.

Matthew Richardson and Pedro Domingo. 2006. Markov logic networks. *Machine Learning*, 26:107–136.

Stefan Schoenmackers, Oren Etzioni, Daniel S. Weld, and Jesse Davis. 2010. Learning first-order horn clauses from web text. In *Proc. of EMNLP2010*, pages 1088–1098.

Keiji Shinzato, Tomohide Shibata, Daisuke Kawahara, Chikara Hashimoto, and Sadao Kurohashi. 2008. TSUBAKI: An open search engine infrastructure for developing new information access. In *Proc. of IJCNLP*, pages 189–196.

Kentaro Torisawa, Stijn De Saeger, Jun'ichi Kazama, Asuka Sumida, Daisuke Noguchi, Yasunari Kakizawa, Masaaki Murata, Kow Kuroda, and Ichiro Yamada. 2010. Organizing the web's information explosion to discover unknown unknowns. *New Generation Computing*, 28(3):217–236.

Masaaki Tsuchida, Stijn De Saeger, Kentaro Torisawa, Masaki Murata, Jun'ichi Kazama, Kow Kuroda, and Hayato Ohwada. 2010. Large scale similarity-based relation expansion. In *Proc of the 4th IUCS*, pages 140–147.

Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. 2003. Kernel methods for relation extraction. *Journal of Machine Learning Research*, pages 1083–1106.