Causal Relation Detection Using Cue Patterns and Association Metrics

Abstract

This paper aims to detect causal relations that exist between two events or phenomena represented by a pair of noun phrases. We introduce four features for causal extraction, which leverage both cue patterns and associativity between the two noun phrases. We then present two supervised methods and an unsupervised method that utilize these features and show that they compare favorably against two state-of-the-art approaches in causal noun-phrase extraction. Our best supervised approach achieves an F-1 score of 0.778, which is 11.2 percent ahead of previous work.

1 Introduction

Automatic identification of semantic relations in text is an important task in many natural language processing applications, such as question answering (QA), information extraction (IE), information retrieval (IR), etc. Causal relation is one of the most common and important semantic relations in these applications. For example, Q is a causal question for QA system by Girju et al.(Girju, 2003), and the causation module can help the system improve the answer from A-1 to A-2.

Q What are the *effects* of acid rain?

A-1 Projects, reports, and information about the effects of acid rain.

A-2 *Acid rain* is known to *contributes to* the *corrosion of metals*.

In this work, we aim to detect and extract causal relations between noun phrases from open domain text. Given a text corpus, we first identify the candidate noun phrases by matching the text with causal cue patterns(Girju and Moldovan, 2002). The causal patterns around the noun phrases and their statistical information provide good causal association metric. Next, we extract four numeric features based on cue patterns and associativity to help identify causal relations. We then develop supervised and unsupervised methods to classify causal pairs using these features. The main contributions of this paper are summarized below:

- We propose four numeric features (Section 3.2) to represent causality between noun phrases. These features demonstrated superior effectiveness over the features proposed in previous work (Girju, 2003; Chang and Choi, 2006).

- We developed both supervised and unsupervised learning algorithms (Section 3.3) which utilize the four features to extract and classify causal pairs. Our algorithms, especially the supervised algorithm, achieve better accuracy on the entire Wikipedia corpus by significant margin (see Section 4) than the previous state-of-the-art approaches reported in Section 2.

2 Previous Work

Previous work on causal relation extraction is relatively sparse, especially on noun-phrase causality discovery. The existing approaches use hand-coded and domain-specific patterns to extract causal knowledge. Girju et al. first focused on casual knowledge discovery between nominals (Girju, 2003). They semi-automatically extracted causal cue, but only extracted noun category features for the head noun. Chang et al. developed an unsupervised method and utility lexical pairs and cue contained in noun phrases as features to identify causality between them (Chang and Choi, 2006). Both of them ignored how the remaining causal text span between noun phrases effects the semantics. We proposed numeric features based on that, and get a better result. Blanco et al. used different patterns to detect the causation in sentences that contain clauses (Blanco et al., 2008). And most recently, Do et al. (Do et al., 2011) first introduced a form of association metric into causal relation extraction. They used discourse connectives and similarity distribution to identify event causality between predicate, not noun phrases, and achieved an F1-score around 0.47.

3 The Approach

Causal relation exists between two events, one being the cause, and the other being effect. We focus on noun phrase causality which both the cause and the effect are noun phrases. Our approach generally follows three steps. First, we extract candidate triples which include the cause, the effect, and a text span that contains the causal cue patterns between the two events. The *text span* is defined as the string between the two noun phrases. Second, we extract some crucial features from the triples as alternate representations. And finally, we employ a learning algorithm to classify each of the triples to either a true causal relation or not.

3.1 Causal Candidates Extraction

In order to extract causal candidates from open domain text, we first employ the open information extraction system Reverb (Banko et al., 2007) to extract all relation triples that satisfy one of the 71 causal cue patterns which were previously reported by Girju et al.(Girju, 2003). We extract causal text span and noun-phrase chunkers together. Then we filter out those chunkers whose head word are not included in WordNet(Fellbaum, 1998). At this point, we obtain the causal candidates from text. The candidates are the triples in the following form: $\langle NP1$, *cue*, $NP2\rangle$, where *cue* stands for the causal cue pattern such as “cause”, “lead to”, etc. For example, from sentence A-2 in Section 1, we can get the causal candidate triple as follows: ⟨acid_rain, contribute_to, corrosion_of_metals⟩. We may get more than one triple from each sentence.

3.2 Feature Extraction

The features extracted in this paper are based on the *causal association intuition*. That is, all the causal pairs are correlated, but not all correlated pairs are causal. If we only use association metrics to detect causal pairs, it will not be sufficient. However, we obtained the causal candidates by causal cue patterns in Section 3.1, then the association metric can be additional hint to identify the valid causal pairs. Based on these considerations, we extract four numeric features from each candidate. All these features are simple but effective.

The most intuitive feature is the pointwise manual information (PMI) value pmi between two noun phrases of each candidate triple:

$$pmi(np_1, np_2) = \log\left(\frac{prob(np_1, np_2)}{prob(np_1)prob(np_2)}\right).$$

This feature presents the events distribution on original corpus.

Then, we analyze the *cues* in candidate triples as well as their original text spans and define another three features. We propose a feature vbs to punish the association metric score if the cue patterns contained in the text span is other part of speech than verb. This feature can disambiguate candidates which contain low quality cue patterns, such as “start”, “spark”, etc.

We further analyze the causality using statistics of noun concepts. Some nominal sequences with high or medium frequency (depending on the corpus) tend to co-occur with certain causal cue patterns. This tendency can help to measure causal correlation between two noun phrases. For triple $\langle np_1$, cue, $np_2\rangle$, we define the *balanced tanimoto score (bts)* feature as follows:

$$bts = \frac{shr}{n_1} + \frac{shr}{n_2}$$

where n_1 and n_2 are the number of causal cue patterns that have co-occurred with np_1 and np_2 respectively, and shr is the number of causal cue patterns which co-occurred with both np_1 and np_2.

The last feature $rdist$ is to measure the relative distance between np_1 and np_2, we just divide the length of *cue* by the length of text span that contains it.

So far, we extract all the four numeric features: pmi, vbs, bts, $rdist$, which can be used to represent the candidate triples.

3.3 Learning Algorithm

In this part, we build a classifier to identify true causal relations in the candidate triples.

3.3.1 Supervised Learning Method

Our first attempt of a supervised learning method uses a linear model to combine the numeric features. The linear classifier is defined as follows:

$$s(t) = \sum_{i=1}^{4} w_i * f_i$$

where s is the metric value of the triple t, f_i is the feature of the triple t, and w_i is the weights of f_i. To solve the model, we only need to get the weights w_i and the threshold m. We use the threshold to classify triples. For triple $t = \langle np_i,\ cue,\ np_j \rangle$, if $s(t) \geq m$, we take $\langle np_i,\ np_j \rangle$ as causal pair, otherwise we reject this pair as non-causal. To this end, we use the simulated annealing algorithm (Kirkpatrick et al., 1983) to search for a reasonable solution (w_1, w_2, w_3, w_4, m).

Since the features we extract are all numeric, we further adopt an implementation of Logistic Regression model(Witten and Frank, 2005) to build the classifier.

3.3.2 Unsupervised Learning Method

The annotated work for supervised models is time consuming and cannot scale up. We next propose an unsupervised method to solve this problem. Our inspiration comes from Chang and Choi(Chang and Choi, 2006). Generally speaking, we embed our causal association metrics model into an EM algorithm. During each iteration, we update the model parameters in E-step, and classify the candidates in M-step until the parameters converge. We built the initial classifier with pmi feature only. The top 45% of triples are classified into “causal”, and the rest are classified into “non-causal”. Then, for the following classification of each iteration, we reconstruct a Naive Bayesian Classifier with new parameters. The class c of triple t is computed as follows:

$$c = \arg\max_{c_k} P(c_k|t_i) = \arg\max_{c_k} \frac{P(c_k)P(t_i|c_k)}{P(t_i)}$$

We assume that all the causal features are independent. In our model, $P(t_i|c_k)$ can be rewritten as:

$$P(t_i|c_k) = p(ccpt_i|c_k) \prod_{j=1}^{4} P(t_i(f_j)|c_k)$$

where $t_i(f_j)$ gives the jth numeric feature value of triple t_i, and *cppt*$_i$ is the causal cue phrase of triple t_i. We define the $P(t_i(f_j)|c_k)$ as follows:

$$P(t_i(f_j)|c_k) = \frac{freq(\forall t_x(f_j) \in \theta(t_i(f_i)))}{freq(\forall t_x \in c_k)}$$

where t_x presents the arbitrary triple, gives an interval which $t_i(f_i)$ belongs to. The range of each feature is pre-divided into 10 intervals, so that they would not change in each iteration.

4 Preliminary Results

We present some preliminary results on Wikipedia articles from different domains.

4.1 Data Set

We use the full set of Wikipedia articles as our original corpus. After the preprocessing and filter step in Section 2, we obtained 5754 candidates. Then, we manually labeled 200 instances as “causal” (103 instances) or “non-causal” (97 instances). We take 150 out of 200 instances as the training set, and take the other 50 instances as the test set. The complete set of training and test data used in this paper can be found at http://adapt.seiee.sjtu.edu.cn/~jessie/causal/dataset.txt.

4.2 Causality Detection Accuracy

We use manually labeled data set for evaluation. Table 1 shows the comparison of Sup+Logistic and Sup+SA approach against with previous work (Girju, 2003) and (Chang and Choi, 2006).

Model	**Precision**	**Recall**	**F-Score**
Sup + SA	0.715	0.713	0.714
Sup + Logistic	**0.797**	**0.76**	**0.778**
unSup	0.630	0.723	0.673
Girju	0.691	0.667	0.665
Chang	0.666	0.667	0.666

Table 1: The results of different models.

Causal Pairs
⟨monoxide, incomplete_combustion⟩
⟨nuclear_holocaust, world_war_iii⟩
⟨epstein-barr_virus_infection, cancer⟩
⟨mud_volcano, earthquake_zone⟩
⟨inbreeding_depression, population_bottleneck⟩
⟨pesticide, air_pollution⟩
⟨population, environmental_stress⟩
⟨anxiety, destructive_behavior⟩
⟨hyperbilirubinemia, red_blood_cell_destruction⟩
⟨colic, premature_death⟩

Table 2: The list of highly ranked causal pairs.

We see the Sup+Logistic model gives the best performance. The Girju's approach only extracted noun category feature for head noun, and missed some semantic meanings for noun phrase. For example, it cut noun phrase pair ⟨inbreeding_depression, population_bottleneck⟩ into ⟨depression, bottleneck⟩. It loses important semantic information, because the cut pairs don't have the causal meaning any more. Chang's method fixed this problem by using lexical probability feature of noun phrases, but still worked weak on its cue probability feature. Our approaches consider the real noun phrase and extract more effective features for candidates which insufficiently express the causal association metrics as described in Section 3.2.

Our Sup+SA model also can give a rank of extracted causal pairs and help improve the QA system better. The list of highly ranked causal pairs are shown in Table 2.

5 Conclusion

We utilize extracted causal association features to develop both supervised and unsupervised models for extracting causation relations between noun phrases. Our preliminary experiment shows their effectiveness over features proposed in prior work. The combination of causal cue phrases and the causal association metrics demonstrated significant advantages in identifying the causal pairs. The results show that the supervised logistic regression approach has the best performance, with an F-1 score around 0.78. We compare our methods with the existing approaches (Chang and Choi, 2006; Girju, 2003) and shows an 11.2% improvement on the F-1 score. Our unsupervised version of the algorithm also posts a moderate gain over the previous unsupervised approach.

References

Michele Banko, Michael J. Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. 2007. Open information extraction from the web. In *IJCAI*, pages 2670–2676.

Eduardo Blanco, Nuria Castell, and Dan I Moldovan. 2008. Causal relation extraction. In *LREC*.

Du-Seong Chang and Key-Sun Choi. 2006. Incremental cue phrase learning and bootstrapping method for causality extraction using cue phrase and word pair probabilities. *Information processing & management*, 42(3):662–678.

Quang Xuan Do, Yee Seng Chan, and Dan Roth. 2011. Minimally supervised event causality identification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 294–303. Association for Computational Linguistics.

Christiane Fellbaum, editor. 1998. *WordNet: an electronic lexical database*. MIT Press.

Roxana Girju and Dan I. Moldovan. 2002. Text mining for causal relations. In *FLAIRS Conference*, pages 360–364.

Roxana Girju. 2003. Automatic detection of causal relations for question answering. In *Proceedings of the ACL 2003 workshop on Multilingual summarization and question answering*, pages 76–83. Association for Computational Linguistics.

Scott Kirkpatrick, D. Gelatt Jr., and Mario P. Vecchi. 1983. Optimization by simmulated annealing. *Science*, 220(4598):671–680.

Ian H Witten and Eibe Frank. 2005. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.