

Cause-Effect Relation Learning

Zornitsa Kozareva

USC Information Sciences Institute

4676 Admiralty Way

Marina del Rey, CA 90292-6695

kozareva@isi.edu

Abstract

To be able to answer the question *What causes tumors to shrink?*, one would require a large *cause-effect* relation repository. Many efforts have been paid on *is-a* and *part-of* relation learning, however few have focused on *cause-effect* learning. This paper describes an automated bootstrapping procedure which can learn and produce with minimal effort a *cause-effect* term repository. To filter out the erroneously extracted information, we incorporate graph-based methods. To evaluate the performance of the acquired *cause-effect* terms, we conduct three evaluations: (1) human-based, (2) comparison with existing knowledge bases and (3) application driven (SemEval-1 Task 4) in which the goal is to identify the relation between pairs of nominals. The results show that the extractions at rank 1500 are 89% accurate, they comprise 61% from the terms used in the SemEval-1 Task 4 dataset and can be used in the future to produce additional training examples for the same task.

1 Introduction

Over the years, researchers have successfully shown how to build ground facts (Etzioni et al., 2005), semantic lexicons (Thelen and Riloff, 2002), encyclopedic knowledge (Suchanek et al., 2007), and concept lists (Katz et al., 2003). Among the most well developed repositories are those focusing on *is-a* (Hearst, 1992) and *part-of* (Girju et al., 2003; Pennacchiotti and Pantel, 2006) relations. However, to be able to answer the question “*What causes tumors to shrink?*”, one requires knowledge about *cause-effect* relation.

Other applications that can benefit from *cause-effect* knowledge are the relational search engines

which have to retrieve all terms relevant to a query like: “*find all X such that X causes wrinkles*” (Cafarella et al., 2006). Unfortunately to date, there is no universal repository of *cause-effect* relations that can be used or consulted. However, one would still like to dispose of an automated procedure that can accurately and quickly acquire the terms expressing this relation.

Multiple algorithms have been created to learn relations. Some like TextRunner (Etzioni et al., 2005) rely on labeled data, which is used to train a sequence-labeling graphical model (CRF) and then the system uses the model to extract terms and relations from unlabeled texts. Although very accurate, such methods require labeled data which is difficult, expensive and time consuming to create. Other more simplistic methods that rely on lexico-syntactic patterns (Hearst, 1992; Riloff and Jones, 1999; Pasca, 2004) have shown to be equally successful at learning relations, temporal verb order (Chklovski and Pantel, 2004) and entailment (Zanzotto et al., 2006). Therefore, in this paper, we have incorporated an automated bootstrapping procedure, which given a pattern representing the relation of interest can quickly and easily learn the terms associated with the relation. In our case, the pattern captures the *cause-effect* relation. After extraction, we apply graph-based metrics to rerank the information and filter out the erroneous terms.

The contributions of the paper are:

- an automated procedure, which can learn terms expressing *cause-effect* relation.
- an exhaustive human-based evaluation.
- a comparison of the extracted knowledge with the terms available in the SemEval-1 Task 4 dataset for interpreting the relation between pairs of nominals.

The rest of the paper is organized as follows. The next section describes the term extraction procedure. Section 3 and 4 describe the extracted data

and its characteristics. Section 5 focuses on the evaluation and finally we conclude in Section 6.

2 Cause-Effect Relation Learning

2.1 Problem Formulation

The objectives of cause-effect relation learning are similar to those of any general open domain relation extraction problem (Etzioni et al., 2005; Pannacchiotti and Pantel, 2006). The task is formulated as:

Task: Given a *cause-effect* semantic relation expressed through lexico-syntactic pattern and a seed example for which the relation is true, the objective is to learn from large unstructured amount of texts terms associated with the relation.

For instance, given the relation *cause* and the term *virus* for which we know that it can cause something, we express the statement in a recursive pattern¹ “* and *virus* cause *” and use the pattern to learn new terms that cause or have been caused by something. Following our example, the recursive pattern learns from the Web on the left side terms like {*bacteria*, *worms*, *germs*} and on the right side terms like {*diseases*, *damage*, *contamination*}.

2.2 Knowledge Extraction Procedure

For our study, we have used the general Web-based class instance and relation extraction framework introduced by (Kozareva et al., 2008; Hovy et al., 2009). The procedure is minimally supervised and achieves high accuracy of the produced extractions.

Term Extraction: To initiate the learning process, the user must provide as input a seed term *Y* and a recursive pattern “*X** and *Y* verb *Z**” from which terms on the *X** and *Z** positions can be learned. The input pattern is submitted to Yahoo!Boss API as a web query and all snippets matching the query are retrieved, part-of-speech tagged and used for term extraction. Only the previously unexplored terms found on *X** position are used as seeds in the subsequent iteration, while the rest of the terms² are kept. The knowledge extraction terminates when there are no new extractions.

Term Ranking: Despite the specific lexico-syntactic construction of the pattern, erroneous

extractions are still produced. To filter out the information, we incorporate the harvested terms on *X** and *Y** positions in a directed graph $G=(V, E)$, where each vertex $v \in V$ is a candidate term and each edge $(u, v) \in E$ indicates that the term v is generated by the term u . An edge has weight w corresponding to the number of times the term pair (u, v) is extracted from different snippets. A node u is ranked by $u=(\sum_{\forall(u,v) \in E} w(u, v) + \sum_{\forall(v,u) \in E} w(v, u))$ which represents the weighted sum of the outgoing and incoming edges to a node. The confidence in a correct argument u increases when the term discovers and is discovered by many different terms. Similarly, the terms found on *Z** position are ranked by the total number of incoming edges from the *XY* pairs $z= \sum_{\forall(xy,z) \in E'} w(xy, z)$. We assume that in a large corpus as the Web, a correct term *Z** would be frequently discovered by various *XY* term pairs.

3 Data Collection

To learn the terms associated with a cause-effect relation, the user can use as input any verb expressing causality³. In our experiment, we used the verb *cause* and the pattern “* and <seed> cause *”, which was instantiated with the seed term *virus*. We submitted the pattern to Yahoo!Boss API as a search query and collected all snippets returned during bootstrapping. The snippets were cleaned from the html tags and part-of-speech tagged (Schmid, 1994). All nouns (proper names) found on the left and right hand side of the pattern were extracted and kept as potential candidate terms of the cause-effect relation.

Table 1 shows the total number of terms found for the *cause* pattern on *X** and *Z** positions in 19 bootstrapping iterations. In the same table, we also show some examples of the obtained extractions.

Term Position	#Extractions	Examples
<i>X</i> cause	12790	pressure, stress, fire, cholesterol, wars, ice, food, cocaine, injuries bacteria
cause <i>Z</i>	52744	death, pain, diabetes, heart disease, damage, determination, nosebleeds chain reaction

Table 1: *Extracted Terms.*

¹A recursive pattern is a lexico-syntactic pattern for which one of the terms is given as input and the other one is an open slot, allowing the learned terms to replace the initial term directly.

²Including the terms found on *Z** position.

³The user can use any pattern from the thesauri of <http://demo.patrickpantel.com/demos/lexsem/thesaurus.htm>

4 Characteristic of Learning Terms

An interesting characteristic of the bootstrapping process is the speed of leaning, which can be measured in terms of the number of unique terms acquired on each bootstrapping iteration. Figure 1 shows the bootstrapping process for the “cause” relation. The term extraction starts of very slowly and as bootstrapping progresses a rapid growth is observed until a saturation point is reached. This point shows that the intensity with which new elements are discovered is lower and practically the bootstrapping process can be terminated once the amount of newly discovered information does not exceed a certain threshold. For instance, instead of running the algorithm until complete exhaustion (19 iterations), the user can terminate it on the 12th iteration.

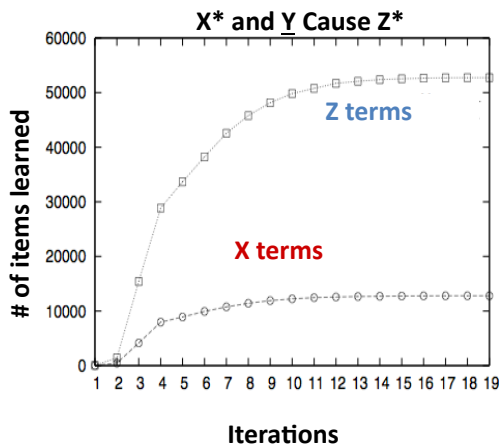


Figure 1: Learning Curve.

The speed of leaning depends on the way the X and Y terms relate to each other in the lexico-syntactic pattern. For instance, the more densely connected the graph is, the shorter (i.e., fewer iterations) it will take to acquire all terms.

5 Evaluation and Results

In this section, we evaluate the results of the term extraction procedure. To the extent to which it is possible, we conduct a human-based evaluation, we compare results to knowledge bases that have been extracted in a similar way (i.e., through pattern application over unstructured text) and we show how the extracted knowledge can be used by NLP applications such as relation identification between nominals.

5.1 Human-Based Evaluation

For the human based evaluation, we use two annotators to judge the correctness of the extracted terms. We estimate the correctness of the produced extractions by measuring *Accuracy* as the number of *correctly* tagged examples divided by the total number of examples.

Figure 2, shows the accuracy of the bootstrapping algorithm with graph re-ranking in blue and without graph re-ranking in red. The figure shows that graph re-ranking is effective and can separate out the erroneous extractions. The overall extractions produced by the algorithm are very precise, at rank 1500 the accuracy is 89%.

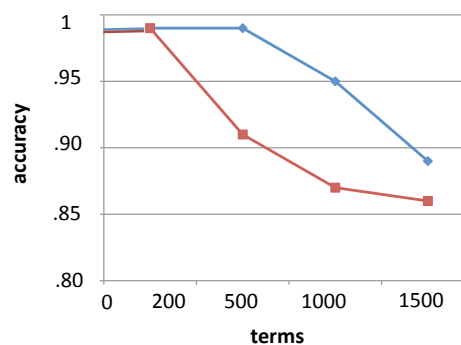


Figure 2: Term Extraction Accuracy.

Next, in Table 2, we also show a detailed evaluation of the extracted X and Z terms. We define five types according to which the humans can classify the extracted terms. The types are: *PhysicalObject*, *NonPhysicalObject*, *Event*, *State* and *Other*. We used *Other* to indicate erroneous extractions or terms which do not belong to any of the previous four types. The Kappa agreement for the produced annotations is 0.80.

X Cause	A1	A2	Cause Z	A1	A2
PhysicalObj	82	75	PhysicalObj	15	20
NonPhysicalObj	69	66	NonPhysicalObj	89	91
Event	21	24	Event	72	72
State	29	31	State	50	50
Other	3	4	Other	5	4
Acc.	.99	.98	Acc.	.98	.98

Table 2: Term Classification.

5.2 Comparison against Existing Resources

To compare the performance of our approach with knowledge bases that have been extracted in a similar way (i.e., through pattern application over unstructured text), we consult the freely available resources NELL (Carlson et al., 2009), Yago

(Suchanek et al., 2007) and TextRunner (Etzioni et al., 2005). Although these bases contain millions of facts, it turns out that NELL and Yago do not have information for the cause-effect relation. While the online demo of TextRunner has query limitation, which returns only the top 1000 snippets. Since we do not have the complete and ranked output of TextRunner, comparing results in terms of relative recall and precision is impossible and unfair. Therefore, we decided to conduct an application driven evaluation and see whether the extracted knowledge can aid an NLP system.

5.3 Application: Identifying Semantic Relations Between Nominals

Task Description (Girju et al., 2007) introduced the SemEval-1 Task 4 on the Classification of Semantic Relations between Nominals. It consists in given a sentence: “*People in Hawaii might be feeling $\langle e1 \rangle$ aftershocks $\langle /e1 \rangle$ from that powerful $\langle e2 \rangle$ earthquake $\langle /e2 \rangle$ for weeks.*”, an NLP system should identify that the relationship between the nominals *earthquake* and *aftershocks* is *cause-effect*.

Data Set (Girju et al., 2007) created a dataset for seven different semantic relations, one of which is *cause-effect*. For each relation, the nominals were manually selected. This resulted in the creation of 140 training and 80 testing cause-effect examples. From the train examples 52.14% were positive (i.e. correct cause-effect relation) and from the test examples 51.25% were positive.

Evaluation and Results The objective of our application driven study is to measure the overlap of the cause-effect terms learned by our algorithm and those used by the humans for the creation of the SemEval-1 Task4 dataset. There are 314 unique terms in the train and test dataset for which the *cause-effect* relation must be identified. Out of them 190 were also found by our algorithm.

The 61% overlap shows that either our cause-effect extraction procedure can be used to automatically identify the relationship of the nominals or it can be incorporated as an additional feature by a more robust system that relies on semantic and syntactic information. In the future, the extracted knowledge can be also used to create additional training examples for the machine learning systems working with this dataset.

Table 3 shows some of the overlapping terms in our system and the (Girju et al., 2007) dataset.

tremor, depression, anxiety, surgery, exposure, sore throat, fulfillment, yoga, frustration, inhibition, inflammation, fear, exhaustion, happiness, growth, evacuation, earthquake, blockage, zinc, vapour, sleep deprivation, revenue increase, quake
--

Table 3: *Overlapping Terms.*

6 Conclusion

We have described a simple web based procedure for learning cause-effect semantic relation. We have shown that graph algorithms can successfully re-rank and filter out the erroneous information. We have conducted three evaluations using human annotators, comparing knowledge against existing repositories and showing how the extracted knowledge can be used for the identification of relations between pairs of nominals.

The success of the described framework opens up many challenging directions. We plan to expand the extraction procedure with more lexico-syntactic patterns that express the *cause-effect* relation⁴ such as *trigger*, *lead to*, *result* among others and thus enrich the recall of the existing repository. We also want to develop an algorithm for extracting cause-effect terms from non contiguous positions like “*stress is another very important cause of diabetes*”. We are also interested in studying how the extracted knowledge can aid a commonsense causal reasoner (Gordon et al., 2011; Gordon et al., 2012) in understanding that if a girl wants to wear earrings it is more likely for her to get her ears pierced rather than get a tattoo. This example is taken from the Choice of Plausible Alternatives (COPA) dataset⁵, which presents a series of forced-choice questions such that each question provides a premise and two viable cause or effect scenarios. The goal is to choose a correct answer that is the most plausible cause or effect. Similarly, the cause-effect repository can be used to support a variety of applications, including textual entailment, information extraction and question answering

Acknowledgments

We would like to thank the reviewers for their comments and suggestions. The research was supported by DARPA contract number FA8750-09-C-3705.

⁴These patterns can be acquired from an existing paraphrase system.

⁵<http://people.ict.usc.edu/gordon/copa.html>

References

- Michael Cafarella, Michele Banko, and Oren Etzioni. 2006. Relational Web Search. In *World Wide Web Conference, WWW 2006*.
- Andrew Carlson, Justin Betteridge, Estevam R. Hruschka Jr., and Tom M. Mitchell. 2009. Coupling semi-supervised learning of categories and relations. In *Proceedings of the NAACL HLT 2009 Workshop on Semi-supervised Learning for Natural Language Processing*.
- Timothy Chklovski and Patrick Pantel. 2004. Verbocean: Mining the web for fine-grained semantic verb relations. In *Proceedings of EMNLP 2004*, pages 33–40.
- Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates. 2005. Un-supervised named-entity extraction from the web: an experimental study. *Artificial Intelligence*, 165(1):91–134, June.
- Roxana Girju, Adriana Badulescu, and Dan Moldovan. 2003. Learning semantic constraints for the automatic discovery of part-whole relations. In *Proc. of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 1–8.
- Roxana Girju, Preslav Nakov, Vivi Nastaste, Stan Szpakowicz, Peter Turney, and Deniz Yuret. 2007. SemEval-2007 task 04: Classification of semantic relations between nominals. In *SemEval 2007*.
- Andrew Gordon, Cosmin Bejan, and Kenji Sagae. 2011. Commonsense causal reasoning using millions of personal stories. In *Proceedings of the Twenty-Fifth Conference on Artificial Intelligence (AAAI-11)*.
- Andrew Gordon, Zornitsa Kozareva, and Melissa Roemmele. 2012. Semeval-2012 task 7: Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval 2012)*.
- Marti Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proc. of the 14th conference on Computational linguistics*, pages 539–545.
- Eduard Hovy, Zornitsa Kozareva, and Ellen Riloff. 2009. Toward completeness in concept extraction and classification. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 948–957.
- Boris Katz, Jimmy Lin, Daniel Loreto, Wesley Hildebrandt, Matthew Bilotti, Sue Felshin, Aaron Fernandes, Gregory Marton, and Federico Mora. 2003. Integrating web-based and corpus-based techniques for question answering. In *Proceedings of the twelfth text retrieval conference (TREC)*, pages 426–435.
- Zornitsa Kozareva, Ellen Riloff, and Eduard Hovy. 2008. Semantic class learning from the web with hyponym pattern linkage graphs. In *Proceedings of ACL-08: HLT*, pages 1048–1056.
- Marius Pasca. 2004. Acquisition of categorized named entities for web search. In *Proc. of the thirteenth ACM international conference on Information and knowledge management*, pages 137–145.
- Marco Pennacchiotti and Patrick Pantel. 2006. Ontologizing semantic relations. In *ACL-44: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 793–800.
- Ellen Riloff and Rosie Jones. 1999. Learning dictionaries for information extraction by multi-level bootstrapping. In *AAAI '99/IAAI '99: Proceedings of the Sixteenth National Conference on Artificial intelligence*.
- Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees.
- Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: a core of semantic knowledge. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 697–706.
- Michael Thelen and Ellen Riloff. 2002. A Bootstrapping Method for Learning Semantic Lexicons Using Extraction Pattern Contexts. In *Proc. of the 2002 Conference on Empirical Methods in Natural Language Processing*, pages 214–221.
- Fabio Massimo Zanzotto, Marco Pennacchiotti, and Maria Teresa Pazienza. 2006. Discovering asymmetric entailment relations between verbs using selectional preferences. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 849–856.