Using ConceptNet to Teach Common Sense to an Automated Theorem Prover*

Claudia Schon[1], Sophie Siebert[2], and Frieder Stolzenburg[2]

[1] Institute for Web Science and Technologies, University of Koblenz-Landau, Koblenz, Germany
schon@uni-koblenz.de
[2] Harz University of Applied Sciences, Automation and Computer Sciences Department, Friedrichstr. 57–59, 38855 Wernigerode, Germany
{ssiebert,fstolzenburg}@hs-harz.de

Abstract

The CoRg system is a system to solve commonsense reasoning problems. The core of the CoRg system is the automated theorem prover Hyper that is fed with large amounts of background knowledge. This background knowledge plays a crucial role in solving commonsense reasoning problems. In this paper we present different ways to use knowledge graphs as background knowledge and discuss challenges that arise.

1 Introduction

In recent years, numerous benchmarks for commonsense reasoning have been presented which cover different areas: the COPA (Choice of Plausible Alternatives) Challenge [17] requires causal reasoning in everyday situations, the Winograd Schema Challenge [8] addresses difficult cases of pronoun disambiguation, the TriangleCOPA Challenge [9] focuses on human relationships and emotions, and the Story Cloze Test with the ROCStories Corpora [11] focuses on the ability to determine a plausible ending for a given short story, to name just a few. See Fig. 1a for an example of a COPA problem. Most approaches tackling these problems are either based on machine learning or exploit statistical properties of the natural language input and are therefore unable to provide explanations for the decisions made (see e.g. [14, 16]).

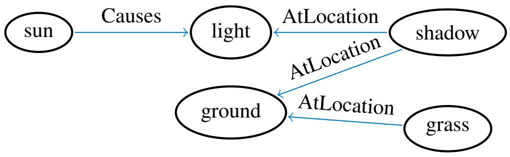
In the CoRg project, we take a different approach by using an automated theorem prover as a key component in our system. Fig. 2 gives an overview of the CoRg system: In a first step, the problem description as well as the two alternative answers are converted into first-order logic formulae using KNEWS [2]. KNEWS (Knowledge Extraction With Semantics) is a software that bridges semantic parsing, word sense disambiguation, and entity linking to produce a unified, abstract representation of meaning. For example the formula generated by KNEWS for the first alternative (*The sun was rising.*) is:

$$\exists A(sun(A) \land \exists B(r1Actor(B,A) \land rise(B)))$$

*The authors gratefully acknowledge the support of the German Research Foundation (DFG) under the grants SCHO 1789/1-1 and STO 421/8-1 *CoRg – Cognitive Reasoning*.

```
1: My body cast a shadow over the grass.
What was the CAUSE of this?
A1: The sun was rising.
A2: The grass was cut.
```

(a) Problem 1 from the COPA benchmark set.



(b) Some information in ConceptNet relevant for COPA problem 1.

Figure 1: COPA problem 1 together with some relevant knowledge from ConceptNet.

Each of these formulae is then, along with appropriate background knowledge, passed to the automated theorem prover Hyper [3]. Currently, we are using Adimen-SUMO [1] and WordNet [10] as background knowledge. Hyper computes a (possibly partial) model for each formula together with the background knowledge. This (possibly partial) model contains inferences performed by Hyper starting from the formula created for the natural language sentences of the two possible answers. In the last step, the three models created for the problem and the alternatives are further processed by a machine learning component in order to decide which model points more into the direction of the problem description. This includes a preprocessing step and a deep learning neural network.

In the preprocessing step, the model has to be decoded, so the neural network can process it. On the one hand, the logical facts in the model contain symbols corresponding to natural language words like *sun* or *astronomicalBody* which can be extracted. On the other hand, there is structural information in the model given by the term structure of the logical facts. For instance, the two facts $sun(sk0)$ and $is\_instance(sk0, astronomicalBody)$ mean that the sun is an astronomical body which is expressed via the Skolem constant $sk0$. However, we currently only care about the symbols corresponding to natural language words, i.e., *sun* and *astronomicalBody*. The structural information is yet dismissed, however future work might address it as well.

After the preprocessing step each premise and alternative consist of a sequence of words, i.e. the extracted symbols. These words are looked up in the ConceptNet Numberbatch* word embedding to transform each sequence of words into a sequence of 300 dimensional word vectors. Word embeddings hold semantic value by assigning numerical values to words, thus making them comparable in a mathematical way. In the next step the looked up sequences of word vectors are fed into the neural network in premise answer pairs, such that each problem generates n training examples, with n being the number of alternatives to choose from.

Eventually, our neural network has two inputs, each encoding the information of the problem and one of the answer candidates. In the core, the encoded information are merged together using an attentive reader approach [21] with bidirectional LSTMs (long short-term memory) [5]. At the output layer a likelihood is calculated of how well an alternative fits to the respective premise. The n likelihoods of the alternatives for one problem are compared and the higher one is chosen to be the selected answer of our system [18]. The inferences performed to construct the model of the chosen answer can be used to provide an explanation for the answer.

2 Using Knowledge Graphs as Background Knowledge

Besides ontologies like SUMO [13, 15], Adimen SUMO [1], Cyc [7] and Yago [20], knowledge graphs constitute an important source of background knowledge. The term *knowledge graph* was coined by an

*https://github.com/commonsense/conceptnet-numberbatch, accessed 12-June-2019

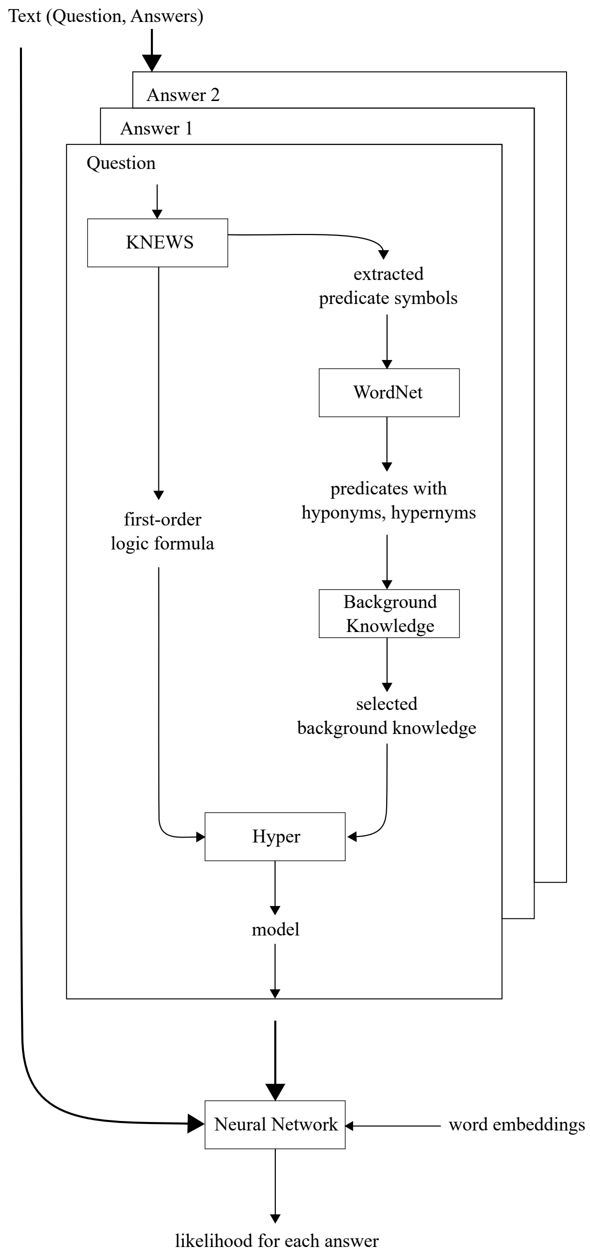


Figure 2: The CoRg system.

initiative of the same name by Google in 2012 and is now widely used as a generalized term for interlinked descriptions of entities. Compared to ontologies, these knowledge graphs usually contain mainly factual knowledge as triples of the form (s, p, o) (subject – predicate – object). However, they contain very large amounts of this factual knowledge. Examples for knowledge graphs are BabelNet [12] and ConceptNet [19]. ConceptNet is a freely-available semantic network, designed to help computers understand the meanings of words that people use. Since ConceptNet contains large amounts of commonsense knowledge that is hardly present in other knowledge bases, we would like to use ConceptNet as a source for background knowledge in the CoRg project. One example for knowledge represented in ConceptNet that is difficult to find in other sources is the following triple

$$(\mathit{snore}, \mathit{HasSubevent}, \mathit{annoy\ your\ spouse}) \tag{1}$$

If knowledge represented in a knowledge graph like ConceptNet is supposed to be used by a first-order logic theorem prover, the triples have to be translated to first-order logic. The easiest way to do that would be to translate p into a predicate name and both s and o into constants leading to $p(s, o)$. Since this is only factual knowledge, it is only of limited use for the commonsense reasoning task under consideration. Due to the fixed set of predicates used in ConceptNet it is possible to create translations of ConceptNet triples to first-order logic formulae depending on the predicate used in the triple. Another way to translate a triple (s, p, o) given in ConceptNet into the first-order logic formula would be:

$$\forall x \Big(s(x) \rightarrow \big(\exists y (p(x,y) \wedge o(y)) \big) \Big) \tag{2}$$

The triples given in Fig. 1b would be translated to:

$$\forall x \Big(\mathit{sun}(x) \rightarrow \big(\exists y (\mathit{causes}(x,y) \wedge \mathit{light}(y)) \big) \Big) \tag{3}$$

$$\forall x \Big(\mathit{shadow}(x) \rightarrow \big(\exists y (\mathit{atlocation}(x,y) \wedge \mathit{light}(y)) \big) \Big) \tag{4}$$

$$\forall x \Big(\mathit{shadow}(x) \rightarrow \big(\exists y (\mathit{atlocation}(x,y) \wedge \mathit{ground}(y)) \big) \Big) \tag{5}$$

$$\forall x \Big(\mathit{grass}(x) \rightarrow \big(\exists y (\mathit{atlocation}(x,y) \wedge \mathit{ground}(y)) \big) \Big) \tag{6}$$

At first glance, this translation looks quite promising. On closer inspection, however, one realizes that starting from a fact $\mathit{sun}(a)$, it is not possible for a constant a to derive something using the predicate *shadow*. The problem is that the direction of the edges in ConceptNet affects the generated FOL formulas. One possible solution would be to generate two formulas for each edge in ConceptNet. For example, for the edge from light to shadow we could generate the additional formula:

$$\forall x \Big(\mathit{light}(x) \rightarrow \big(\exists y (\mathit{maysituate}(x,y) \wedge \mathit{shadow}(y)) \big) \Big) \tag{7}$$

Another problem is the large amount of information available in ConceptNet. The information in Fig. 1b presents a manually selected part of ConceptNet which is relevant for problem 1 of the COPA challenge. In total, node *sun* has 637 incoming and 1,000 outgoing edges and node *shadow* 399 incoming and 626 outgoing edges in ConceptNet. It is not trivial to select from this variety of edges those that are relevant to the problem under consideration.

To solve this problem, we plan three things:

- We will only consider a subset of the relations used in ConceptNet. ConceptNet uses a fixed set of around 40 relations in its triples. Examples for these relations are general relations like *IsA* and *PartOf* as well as more specific relations like *CapableOf* and *Desires*. All relations can be negated by prefixing them with the word *Not*. Many of these relations are not interesting for our purpose. For example the relation *ExternalURL* can be used to point to an URL outside of ConceptNet where further linked data about a certain node can be found. Furthermore, there are relations providing information relevant for languages other than English. This is why we plan to manually selected set of relations that are interesting for the COPA problems.

- Despite the restriction to a subset of the relations used in ConceptNet, the formula set generated from ConceptNet is likely to be too large. Therefore, we will try to select suitable formulas from this formula set. Here we will experiment with SInE [6] and Similarity SInE [4].

- Starting from the manually selected relations from ConceptNet, we only consider the triples whose third component is similar to the words in the COPA problem currently under consideration. To measure the similarity we are planning to use word embeddings like Concept Numberbatch. This would result in not using the triple $(sun, IsA, star)$ for COPA problem 1, since none of the words in COPA problem one is very similar to the word star. In contrast to that, $(sun, Causes, light)$ would be used.

3 Conclusion/Future Work

In this paper, we introduced the CoRg system which is able to tackle commonsense reasoning problems by combining a first-order logic theorem prover, background knowledge bases and machine learning. We discussed how to integrate knowledge represented in a knowledge graph into the CoRg system such that the theorem prover is able to use this knowledge. In future work, we plan to investigate how to deal with the overwhelming amount of triples in knowledge graphs. In addition to that, we would like to integrate other knowledge graphs, like for example BabelNet [12] into our system for commonsense reasoning.

References

[1] J. Álvez, P. Lucio, and G. Rigau. Adimen-SUMO: Reengineering an ontology for first-order reasoning. *Int. J. Semantic Web Inf. Syst.*, 8(4):80–116, 2012.

[2] V. Basile, E. Cabrio, and C. Schon. KNEWS: Using logical and lexical semantics to extract knowledge from natural language. In *Proceedings of the European Conference on Artificial Intelligence (ECAI) 2016 conference*, 2016.

[3] M. Bender, B. Pelzer, and C. Schon. System description: E-KRHyper 1.4 – extensions for unique names and description logic. In M. P. Bonacina, editor, *CADE-24*, LNCS 7898, pages 126–134. Springer, 2013.

[4] U. Furbach, T. Krämer, and C. Schon. Names are not just sound and smoke: Word embeddings for axiom selection. In *to appear in CADE-27 – The 27th International Conference on Automated Deduction*, volume 11716 of *LNCS*. Springer, 2019.

[5] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[6] K. Hoder and A. Voronkov. Sine qua non for large theory reasoning. In *CADE*, volume 6803 of *Lecture Notes in Computer Science*, pages 299–314. Springer, 2011.

[7] D. B. Lenat. Cyc: A large-scale investment in knowledge infrastructure. *Communications of the ACM*, 38(11):33–38, 1995.

[8] H. J. Levesque. The Winograd Schema Challenge. In *Logical Formalizations of Commonsense Reasoning, Papers from the 2011 AAAI Spring Symposium, Technical Report SS-11-06, Stanford, California, USA, March 21-23, 2011*. AAAI, 2011.

[9] N. Maslan, M. Roemmele, and A. S. Gordon. One hundred challenge problems for logical formalizations of commonsense psychology. In *Twelfth International Symposium on Logical Formalizations of Commonsense Reasoning, Stanford, CA*, 2015.

[10] G. A. Miller. WordNet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[11] N. Mostafazadeh, M. Roth, A. Louis, N. Chambers, and J. Allen. LSDSem 2017 shared task: The story cloze test. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*, pages 46–51, 2017.

[12] R. Navigli and S. P. Ponzetto. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250, 2012.

[13] I. Niles and A. Pease. Towards a standard upper ontology. In *Proceedings of the international conference on Formal Ontology in Information Systems-Volume 2001*, pages 2–9. ACM, 2001.

[14] S. Ostermann, M. Roth, A. Modi, S. Thater, and M. Pinkal. SemEval-2018 task 11: Machine comprehension using commonsense knowledge. In *Proceedings of the 12th International Workshop on Semantic Evaluation*, pages 747–757, 2018.

[15] A. Pease. *Ontology: A Practical Guide*. Articulate Software Press, Angwin, CA, 2011.

[16] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding by generative pre-training. Technical report, Open AI, 2018.

[17] M. Roemmele, C. A. Bejan, and A. S. Gordon. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning*, 2011.

[18] S. Siebert, C. Schon, and F. Stolzenburg. Commonsense reasoning using theorem proving and machine learning. In A. Holzinger, P. Kieseberg, E. Weippl, and A. M. Tjoa, editors, *CD-MAKE 2019 – Machine Learning and Knowledge Extraction*, LNCS, Canterbury, UK, 2019. Springer Nature Switzerland. To appear.

[19] R. Speer, J. Chin, and C. Havasi. ConceptNet 5.5: An open multilingual graph of general knowledge. In S. P. Singh and S. Markovitch, editors, *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA.*, pages 4444–4451. AAAI Press, 2017.

[20] F. M. Suchanek, G. Kasneci, and G. Weikum. YAGO: A large ontology from Wikipedia and WordNet. *Web Semant.*, 6(3):203–217, Sept. 2008.

[21] M. Tan, C. d. Santos, B. Xiang, and B. Zhou. LSTM-based deep learning models for non-factoid answer selection. CoRR – Computing Research Repository abs/1511.04108, Cornell University Library, 2015.