An Approach to Solve Winograd Schema Challenge Using Automatically Extracted Commonsense Knowledge

Arpit Sharma, Nguyen H. Vo , Shruti Gaur & Chitta Baral
Department of Computer Science, Arizona State University
Tempe, Arizona 85281

Abstract

The Winograd Schema Challenge has recently been proposed as an alternative to the Turing test. A Winograd Schema consists of a sentence and question pair such that the answer to the question depends on the resolution of a definite pronoun in the sentence. The answer is fairly intuitive for humans but is difficult for machines because it requires commonsense knowledge about words or concepts in the sentence. In this paper we propose a novel technique which semantically parses the text, hunts for the needed commonsense knowledge and uses that knowledge to answer the given question.

1 Introduction

Recently, the Winograd Schema Challenge (Levesque, Davis, and Morgenstern 2011) has been proposed as an alternative to the Turing test. A schema in it consists of pairs of a sentence and a question such that the question can be answered by resolving a definite pronoun or possessive adjective to one of its two antecedents in the sentence. The antecedents belong to the same gender (both are either two objects, two males or two females) and they have a number agreement between them (both are either singular or plural). This property makes it harder to resolve the pronoun to its antecedent. The sentence also contains a *"special word"* which when replaced by another word (*alternate word*), the answer to the question also changes. For example, given the **Sentence**, *"The man couldn't lift his son because he was so **heavy**. "*, and the **Question** *"Who was heavy ?"*, its expected **Answer** is *"the son "*. However, changing the **Sentence** slightly to *"The man couldn't lift his son because he was so **weak**."*, and changing the **Question** to *"Who was weak ?"*, the expected **Answer** is now *"the man"*.

One of the motivation behind this challenge is to simulate human-like reasoning in machines. A machine can be said to demonstrate such a behavior if it can do human-like reasoning on text using commonsense knowledge. However, such knowledge may not be explicitly stated in a given text. Hence, in this paper we propose to automatically hunt such commonsense knowledge and reason using it to answer a given question. We demonstrate its applicability by identifying two types of commonsense knowledge that we are able to hunt and that is helpful in solving a subset of the Winograd Schema Challenge corpus.

2 An Overview of Our Approach

The Winograd Schema (WS) Challenge corpus consists of 282 sentence and question pairs and they represent a wide variety of commonsense reasoning categories. We identified two categories that we address in this paper.

• **Direct Causal Events - Event-event causality**: In this category, the commonsense knowledge required for resolution has two mutually causal events (*explained* and *convince* in the example below) such that a pronoun participates in one of the event and its candidate co-referent participates in another. For example, for the text *"Sid explained his theory to Mark but he could not convince him ."* and the question *"Who could not convince ?"*, the expected answer is *"Sid"*. To resolve *he* to *Sid*, the commonsense knowledge required is of the form, **IF** (*X explained to Y but Z could not convince*) **THEN** (*Z=X* **i.e.** agent of *explained*=agent of *could not convince*).

• **Causal Attributive**: In this category, the commonsense knowledge required has an event and a participant entity in that event has an associated attribute that is causally related to the event. For example, for the text *"Pete envies Martin because he is very successful."* and the question *"Who is successful ?"*, the expected answer is *"Martin"*. The commonsense knowledge required to get this answer is of the form: *X envies Y* **because** *X* **has trait** *successful*. Here *X* is the participant entity and *envies* is the event in the sentence.

We identified a total of 71[1] WS corpus sentences in the above mentioned categories. The remaining 211 sentences[2] do not fall into these categories because the kind of commonsense knowledge required to solve above categories is not suitable for these sentences. For example, for the text *"There is a pillar between me and the stage, and I can not see it ."* and the question *"What can not I see ?"* the required commonsense knowledge is that *"If something big comes in between me and then stage then my sight is blocked; pillar*

---

[1]including #4, #6, #72 and #23 from http://www.cs.nyu.edu/davise/papers/WS.html

[2]including #34, #35, #41, #48, #50 and #43

represents something big; and if my sight is blocked then I can not see." As we can see that this commonsense knowledge does not fit into the above two categories.

In the paper, we report on our work to automatically hunt down the above mentioned two kinds of commonsense knowledge about the given Winograd sentences and using that knowledge to get the answer to the respective questions. The subsections below briefly explain the functionality of our system.

2.1 Defining Formal Representation and Translating Winograd Sentence and Question

A representation is considered good if it can express the structure of the text, can distinguish between the events and their environment in the text, uses a general set of relations between the events and their participants, and is able to represent the same events or entities in different perspectives.

Keeping these features in mind, we have used the Stanford dependency parser (De Marneffe et al. 2006) to parse the text. The dependency relations (De Marneffe and Manning 2008) are then mapped into a widely used Component Library in Knowledge Machine (KM) (Clark, Porter, and Works 2004), (Barker, Porter, and Clark 2001).

Furthermore, to distinguish between words while keeping their conceptual class information, the class for each word in the representation is added by using Word Sense Disambiguation (Basile et al. 2007) and lexical class information from WordNet (Miller 1995). It helps in keeping the identity of entities protected in the text without loosing the information in case they belong to the same conceptual class. For example, in the sentence "*Ann asked Mary what time the library closes , but she had forgotten .*", the information that both *ann_1* and *mary_3* (_number represents position of word in the sentence) are different is protected and their belongingness to the same superclass *person* is preserved. Also, if a sentence contains two exactly same words in different context (say *students_2* and *students_4*) then they are represented as instances of the same class, i.e., *student*.

We also defined an algorithm consisting of a set of rules to match a question's formal representation with that of the respective sentence's and extract the *pronoun to be resolved* from the sentence.

2.2 Extraction of Commonsense Knowledge about the Sentence

The questions in Winograd Schema Challenge can be easily answered by human beings using simple knowledge that they have learned over the years. One of the ways they learn is by reading. This is supported by studies such as the one mentioned in `http://www.theguardian.com/books/2014/jan/23/can-reading-make-you-smarter`. The article states that *"There is evidence that reading can increase levels of all three major categories of intelligence."*. For example if a person attempts to answer the question *Who had forgotten?* based on the sentence, *Ann asked Mary what time the library closes, but she had forgotten .* then she can easily resolve *she* to *Mary*. This is done by using the

commonsense knowledge that *if A asked something to B but Z forgot* then *B=Z*.

In our system we try to emulate this. The only difference is that we are retrieving the commonsense knowledge only relevant to the given sentence and the question. We do that by creating string queries from the concepts in the sentence and the question and later use the queries to retrieve sentences from a large corpus of raw text.

We now use the above mentioned example to explain the two step process. The first step is to create a query set by using the representation of the given sentence and question. Following are the sets of queries that are created:

- The first set of queries is created by using formal representations of both the Winograd sentence and the question. All the nodes from the question's formal representation (except the ones which represent "Wh" words) are mapped into the formal representation of the given sentence. From the mapped output, all the words/nodes which do not specify a nominal entity are extracted and their different combinations are joined together using a wild card (*) and quotes (""). An example query for the *asked* example mentioned above is, *".*asked.*but.*forgotten.*"*.

- The second set of queries is created by replacing the verbs in the previously created set of queries by their synonyms. For eg. a new query for *asked* example that is generated is: *".*inquired.*but.*forgotten.*"*, where *inquired* is a synonym of *asked*.

Finally, a combined set of queries is formed by merging the above two sets.

The second sub-step in the commonsense knowledge extraction process is to automatically search a large corpus of raw English text using the queries and extract the sentences which contain all the words (in any form) in the respective query. Currently we are using the Google search engine API along with sentence splitting to get such sentences from the textual part of WWW but the searching can be performed on other datasets too. The idea here is to extract the sentences which contain the commonsense knowledge that is required to answer the question about the given Winograd sentence. One of the sentences extracted from Google by using the above mentioned queries for the *asked* example is "*But you asked me the security question but I forgotten*". Since in English when a sentence has the two words *me* and *I* they both represent the same entity, we use that in post-processing. Such post processing is performed on the semantic representation of the sentence.

2.3 Reasoning on Formal Representations

The last step is to use the formal representations of the given sentence and the commonsense knowledge extracted in the previous step to get the final answer. As mentioned earlier, we focus on two categories of sentences i.e. *Direct Causal Events* and *Causal Attributive*. Similar type of logical reasoning is used for both the types. A subgraph from the given Winograd sentence is compared with the extracted commonsense sentence. The entity that replaces the *pronoun to be resolved* in the given commonsense sentence is extracted.

In the section below, we briefly explain the tasks accomplished for solving both the types of sentences.

Defining a general set of rules applicable to both the types of sentences: The following set of properties are defined for both Winograd and commonsense sentences' semantic representations.

- The basic transitivity relationship between two event nodes is defined if an event node is reachable from another event node, traversing along any directed edge in the semantic graph.

- If two different nodes in different sentences (Winograd or Commonsense) are instances of the same conceptual class then they are defined as cross context siblings.

- A node in any semantic graph is defined to have negative polarity if it has a *negative* edge coming out of it.

Defining the type specific set of ASP rules: **1. Type1: Direct Causal Events**

- There are two event nodes connected transitively in the semantic graph of the Winograd sentence. First step is to identify the similar chain of two transitive event nodes from commonsense sentence's semantic graph by using the general properties defined in the previous section.

- A subgraph from the semantic representation of the given Winograd sentence is extracted which consists of the *pronoun to be resolved* and the events and entities required to resolve it to its antecedent. A similar subgraph (based on the general properties and matching event nodes extracted in previous step) is extracted from the semantic representation of the commonsense sentence.

- Finally, both the subgraphs extracted in the previous steps are compared and the resolution of pronoun is done.

2. Causal Attributive: The reasoning for this type also follows a similar graph based reasoning as explained above. The only difference is that in this case the trait/property of entities is retrieved from the commonsense knowledge graph instead of their relations with event chains.

3 Evaluation and Error Analysis

There are 282 total sentence and question pairs in Winograd Schema Challenge corpus. Out of those, we identified a total of 71 sentences from both the categories *Causal Attributive* and *Direct Causal Events* combined.

Among the 71 pairs, our system is able to answer 53 and rest 18 are left unanswered. Out of the 53 answered, 49 are correctly answered. Four of them are incorrectly answered because the commonsense knowledge found was inappropriate. For example the commonsense knowledge "*I paid the price for my stupidity. How grateful I am*" was found for the Winograd Schema sentence, *Bob paid for Charlie's college education, he is very grateful*. In this sentence, there is only one entity in the commonsense sentence (represented by *I* and *me* words, which are post processed as one in our semantic representation, as mentioned earlier). Hence, this particular extracted knowledge is not appropriate for the given sentence.

It must also be noted that our system does not return any answer if no commonsense knowledge is found or it is not sufficient to answer the question. This property is advantageous because it provides the ability to use another commonsense knowledge source and this process is repeatable.

4 Related Works

There are two published techniques that aim at solving the Winograd Schema Challenge corpus or a similar corpus. One of them is a system (Rahman and Ng 2012) which uses a number of techniques to resolve pronouns in Winograd Schema like corpus. Their system uses various techniques and combine their results on a corpus of 941 such schema. There are a few issues with the system such as there is redundancy in the corpus. For example the test corpus contains the sentences, *John shot Bill and he died.* and *The man shot his friend and he died.* In both of these sentences the reasoning is same, so if the system is able to answer one of them correctly then other should also be correctly answered. Another technique used by them is to create string queries from given sentences and find the support for them from Google search result counts. The issue with this technique is that sometimes the queries do not make sense. For example, a query for the sentence *Lions eat zebras because they are predators* is *"Lions are predators"*, which make perfect sense. But if the sentence is changed to *Lions eat zebras because they are hungry* then the query *"Lions are hungry"* does not make sense because *zebras* are equally likely to be hungry without any context.

Another work by Peter Schuller (2014), demonstrates a graph based technique and performs experiments on 4 out of 141 Winograd Schema pairs. It converts the given Winograd sentence to a dependency graph using Stanford dependency parser and then manually creates a background knowledge dependency graph which is required to answer the question. The main contribution is to formalize a way to combine both given sentence dependency graph and the *manually created background knowledge* dependency graph in using relevance theory and then use Answer Set Programming (ASP) (Gelfond and Lifschitz 1988), (Baral 2003) to extract the answer.

Extraction of commonsense knowledge is one of the main components of reasoning process. It is manually encoded into the system proposed by Peter Schuller.

5 Conclusion and Future Work

In this paper, we presented an automated approach to solve two categories of sentences in the Winograd Schema Challenge. There are 71 Winograd Schema sentences that fall in those categories. We achieved a notable accuracy on the schemas. Our approach includes semantic parsing of the given text, automatic extraction of the commonsense knowledge required to answer a specific question about the given sentence and finally use of the commonsense knowledge to extract the answer to the given question.

Furthermore, we we are in a process of identifying other categories of Winograd Schema sentences and as future work we are planning to implement our technique on them.

References

[Baral 2003] Baral, C. 2003. *Knowledge representation, reasoning and declarative problem solving*. Cambridge university press.

[Barker, Porter, and Clark 2001] Barker, K.; Porter, B.; and Clark, P. 2001. A library of generic concepts for composing knowledge bases. In *Proceedings of the 1st international conference on Knowledge capture*, 14–21. ACM.

[Basile et al. 2007] Basile, P.; Degemmis, M.; Gentile, A. L.; Lops, P.; and Semeraro, G. 2007. The jigsaw algorithm for word sense disambiguation and semantic indexing of documents. In *AI* IA 2007: Artificial Intelligence and Human-Oriented Computing*. 314–325.

[Clark, Porter, and Works 2004] Clark, P.; Porter, B.; and Works, B. P. 2004. Kmthe knowledge machine 2.0: Users manual. *Department of Computer Science, University of Texas at Austin*.

[De Marneffe and Manning 2008] De Marneffe, M.-C., and Manning, C. D. 2008. Stanford typed dependencies manual. *URL http://nlp. stanford. edu/software/dependencies manual. pdf*.

[De Marneffe et al. 2006] De Marneffe, M.-C.; MacCartney, B.; Manning, C. D.; et al. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of LREC*, volume 6, 449–454.

[Gelfond and Lifschitz 1988] Gelfond, M., and Lifschitz, V. 1988. The stable model semantics for logic programming. In *ICLP/SLP*, volume 88, 1070–1080.

[Levesque, Davis, and Morgenstern 2011] Levesque, H. J.; Davis, E.; and Morgenstern, L. 2011. The winograd schema challenge. In *AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning*.

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

[Rahman and Ng 2012] Rahman, A., and Ng, V. 2012. Resolving complex cases of definite pronouns: the winograd schema challenge. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 777–789. ACL.

[Schuller 2014] Schuller, P. 2014. Tackling winograd schemas by formalizing relevance theory in knowledge graphs. International Conference on Principles of Knowledge Representation and Reasoning.