DRESDEN UNIVERSITY OF TECHNOLOGY

Research project

On Commonsense Domains within the Winograd Schema Challenge

Aneta Koleva
(Mat.-No.: 4734043)

Supervisors: Prof. Dr. Sebastian Rudolph Dr. Emmanuelle Dietz

Dresden, June 17, 2019

Abstract

We investigate how to formalize commonsense reasoning in machines by analyzing The Winograd Schema Challenge (WSC). WSC is a complex coreference resolution task and requires applying knowledge on commonsense reasoning. In addition, to identify the main challenges of the WSC, we provide a survey of the different state-of-the-art approaches and their methods for addressing the WSC. After that, we perform an elaborate analysis of the WSC problems and identify six categories of commonsense knowledge required for their resolving. These categories might be helpful for further developments in the formal specification of characterizing meta-knowledge, that is required to solve the WSC.

Contents

1 Introduction

The Winograd Schema Challenge (WSC) was proposed by Levesque et al. [LDM12] as a new test in Artificial Intelligence (AI) and possibly as an alternative to the Turing test. The test captures a difficult pronoun disambiguation problem which is an easy task for humans, but remains a still unsolved challenge for computers. The WSC is formulated in such manner that it requires genuine understanding of real world situations and intelligence for solving it. For this reason it is argued that a computer that is able to solve the WSC with human-like accuracy must be able to perform human-like thinking [LDM12]. Different types of commonsense knowledge and reasoning are required to solve the Winograd Schema problems. Thus, the WSC has been proposed as a method for testing automated commonsense reasoning. This idea of designing a machine which could apply commonsense reasoning was first proposed by McCarthy [McC60]. He recognized that commonsense reasoning is a trait of intelligence and therefore tried to express it in formal logic so that it can be used for building a truly intelligent machine. Since then, formalizing commonsense reasoning has been an open challenge in the AI field.

Inspired by an example from Terry Winograd [AW72], the WSC corpus consists of sentences like the following:

S1: The city councilmen refused the demonstrators a permit because they feared violence.

S2: The city councilmen refused the demonstrators a permit because they advocated violence.

The task in the WSC is to identify the correct referent of the pronoun *they*. The difference between the sentences is the special word, in this case *feared/advocated*. Depending on this word, the referent of the pronoun *they* changes. In the first sentence it referes to *the city councilmen* and in the second to *the demonstrators*. This characteristic is what makes the WSC task a restricted form of the coreference resolution problem. The goal in the coreference resolution problem is to identify all the correct antecedents for a pronoun by relying on information about the gender and the number of the pronoun [Cry11]. In the WSC problems the candidates antecedents are always the same gender and number as the ambiguous pronoun, so this information is not sufficient for resolving the correct referent. In order to identify the referent of the ambiguous pronoun, one needs to have knowledge about the relations between the nouns, the verb phrase and the special word from the Winograd sentence.

Various approaches have been suggested for solving the WSC. More generally, the studies of tackling the WSC can be divided into two categories: Machine learning and Knowledge-based. The division is based on the main techniques applied for obtaining the correct answer. In the first category are the approaches which rely on machine learning and deep learning techniques ([RN12, LJL+16b, TL18]). Indeed, approaches from this category ([TL18, RWC+19] are the most recent ones to perform the best, with reported accuracy of over 70%. In the second category are the approaches which rely on knowledge-based systems ([SB16, ECT+18, Sch14]). These require to have formally represented knowledge and procedures that should be able to reason with that knowledge. Many of the previously proposed approaches [SVAB15, Sch14, LJL+16b, ECT+18] recognized that answering the WSC correctly requires additional knowledge to what is in the given sentence. However, to the best of our knowledge few approaches have analyzed the knowledge in the available WSC sentences so far.

For humans, commonsense reasoning comes naturally because of the available background knowledge and because of the understanding about the surrounding world that humans have. In order for machines to be able to appear as if they would do commonsense reasoning, a huge amount of non-domain specific knowledge is needed [MMS+02]. To address this issue, there have been attempts to develop repositories of common knowledge such as Cyc [Len95] and ConceptNet [LS04]. However, it is unclear whether these knowledge bases can ever be completed or if they contain all the necessary information for commonsense reasoning.

Motivated by the need for background knowledge we analyzed the sentences from the WSC corpus and identified six different categories of commonsense reasoning. We describe the process of annotating the WSC problems with these categories and we describe their characteristics.

The rest of this report is structured as follows: In the next chapter we introduce the WSC and explore approaches from both categories which have made significant contributions towards solving the WSC. In chapter 3 we discuss in more details the approach Sharma and Baral [SB18] and their proposed Reasoning Algorithm (RA). We then present the result of our analysis along with examples and description for the identified categories. In this section we also analyze the semantic graphs of the WSs which we analyzed thoroughly and formalized. After that we discuss all the challenges and findings that we came across during this work. Finally, we give concluding remarks and ideas for potential future work.

2 Background

In this chapter we introduce the structure of the WSC and the In section 2.1 we introduce the features of the WSC as proposed by Levesque et al. [LDM12]. Additionally we explain the dataset of Pronoun Disambiguation Problems (PDPS) which was proposed by [MJ15]. Finally, we discuss what makes the WSC so hard for solving and what are the main limitations of this challenge. In section 2.2 we analyze the different state-of-the-art approaches and give a summary of the results from their evaluation.

2.1 The Winograd Schema Challenge

The WSC was originally conceived in 2012 [LDM12] and since then it has caught the attention of many researchers from different areas. However, the first and so far only WSC competition was held in 2016 as part of IJCAI[1]. Before the competition, the rules for executing and evaluating the participants were presented by Morgenstern et al. [MDJ16]. The competition consisted of two rounds, one qualifying and one final, each with 60 questions. For the qualifying round, the questions were randomly chosen from the PDPs dataset and for the final round the questions were from the WSC dataset. In the qualifying round none of the participants achieved accuracy of 90%, which was the requirement for advancing to the final round.

2.1.1 Description

A Winograd Schema (WS) consists of three main parts:

1. A sentence that consists of:
    - Two noun phrases of the same semantic class and of the same gender
    - One ambiguous pronoun that could refer to either of the antecedent noun phrases
    - A special word such that when changed, the resolution of the pronoun is changed

[1] http://ijcai-16.org/

2. A question, possibly containing the special word, asking about the referent of the ambiguous pronoun
3. Two possible answers corresponding to the noun phrases in the sentence

The WSC dataset currently consists of 285[2] such WSs. A typical example of a WS is the following one:
Example 2.1.1:

S: The trophy does not fit into the brown suitcase because it is too small.

Q: What is too small?

A: The suitcase/the trophy

Here, the special word is the adjective at the end of the sentence which can be *small* or *big*. The ambiguous pronoun is *it* and the two antecedents are *trophy* and *suitcase*. The adjective in the question depends on the chosen special word in the sentence. While it is obvious that answering the question in this example is easy for an adult English speaker, it still requires thinking to derive the correct answer. In order to identify the correct referent of *it*, one needs to use world knowledge about the size of objects, more specifically that a small object can fit into a large one, but not the other way around. Hence, if a machine is able to resolve the pronoun correctly, we can conclude that background knowledge was involved. Answering the question in this example is known as a problem of pronoun disambiguation. However, the sentences in the WSs are distinct in that one special word, which when changed causes the other noun phrase to be the correct referent. For example, consider the twin sentence for the WS given in *Example 2.1.1*:

S: The trophy does not fit into the brown suitcase because it is too large.

Q: What is too large?

The possible answers are the same as given in *Example 2.1.1*, but here the correct answer is *the trophy*. Having the special word in the sentences prevents the solver to rely on sentence structure and word order for finding the answer to the question. Moreover, Levesque et al. [LDM12] explain another feature of the WS sentences, which they call *Google proof*: The idea is to prevent finding the answer by using statistical learning on large corpora of text or by just typing the question and the answers into a search engine, such as Google. Here is an example of a WS that is not Google proof:
Example 2.1.2:

S: Tom's books are full of mistakes. Some of them are quite [foolish/worthless].

Q: What are [foolish/worthless]?

[2]https://cs.nyu.edu/faculty/davise/papers/WinogradSchemas/WSCollection.xml

A: The mistakes/the books.

Example 2.1.2 has been provided as a failed example[3] because the term *foolish mistake* is commonly used and can be returned as an answer when querying Google. Additional feature is that resolving a WS should be easy for a human reader. For this reason, human performance is expected to be near 100%. Indeed, Bender [Ben15] conducted a large online experiment and reported 92% success rate for humans. For the execution and evaluation part of the WSC, two distinct datasets have been formulated and published by Morgenstern et al. [MDJ16]. The first dataset consists of pronoun disambiguation problems (PDPs)[4] that are taken from examples found in literature, biographies, essays and news or have been constructed by the organizers of the competition. The second dataset consists of WSs[5] which have been constructed by the challenge organizers. The following is an example of the PDPs:
Example 2.1.3:

Text: When they had eventually calmed down a bit, and had gotten home, Mr. Farley put the magic pebble in an iron safe. Some day they might want to use it, but really for now, what more could they wish for?

Snippet: to use it

Answers: magic pebble/safe

Along with the text, a snippet with the ambiguous pronoun that needs to be resolved and the possible answers are provided. As in *Example 2.1.3*, a PDP may consist of more than one sentence and it can also have more than two possible answers. Similar to resolving WSs, considerable use of commonsense reasoning is required when answering the PDPs. Since the structure of the PDPs is not always consistent, resolving these problems is more difficult than resolving WSs.

Having the two different sets allows the evaluation of the participating systems in the challenge to be done gradually. If a system successfully pases the questions from the first dataset, then it will be challenged with the WSs set. Morgensten et al. [MDJ16] argue that if a system can answer the problems from the PDP set correctly, then it will surely have success when answering the WSs set. Additionally, the PDPs are taken from various literature sources and may include different aspects of commonsense knowledge which would otherwise not be included in the created WSs.

[3] https://cs.nyu.edu/faculty/davise/papers/WinogradSchemas/WSfailed.html
[4] https://cs.nyu.edu/faculty/davise/papers/WinogradSchemas/PDPChallenge2016.xml
[5] https://cs.nyu.edu/faculty/davise/papers/WinogradSchemas/WSCollection.html

2.1.2 Main challenges and limitations

We now analyze and discuss the main challenges that an approach for addressing the WSC must overcome in order to achieve good results. Furthermore, we will identify and discuss the main limitations of the WSC.

Finding the correct noun phrase for an ambiguous pronoun is a known problem in the Natural Language Processing (NLP) research field. Coreference resolution problem depends on different features, such as grammatical role, number, gender, syntactic structure and the distance between the referent and the pronoun. Although existing automatic systems rely on such grammatical features, they do not use world knowledge in the resolution process. Consequently, the Stanford resolver CoreNLP [LPC+11] achieved accuracy of around 55% on the dataset provided by Rahman and Ng [RN12], which is just above the random baseline (50%).

From the examples given earlier, it is clear that background knowledge is required in order for a human reader to derive the correct answer. Since the sentences from both datasets, PDPs and WSs, are non-domain specific it is difficult to determine what kind of background knowledge is needed. Therefore, the first challenge imposed by the WSC is how to obtain knowledge for commonsense reasoning. The second is how to formalize this knowledge such that all important information are preserved. Lastly, how to reason on top of formalized knowledge is the third main challenge when trying to solve the WSC.

Regarding the limitations of the WSC, as a first and probably most important limitation is the number of available sentences. Because the process of creating new WS is difficult, the number of available sentences is very small. Moreover no training data is provided for any of the datasets which makes it hard to approach the task as a machine learning problem. Another limitation is the constraint posed by the language of the sentences. For some other languages, as explained by Davis [Dav16], the disambiguation of a pronoun for different genders may not be as clear as in English. Nevertheless, there are translation of 144 WSs in Japanese[6] and 107 WSs in French [7].

A third limitation is the lack of a more detailed evaluation protocol. In the proposal for the WSC competition the evaluation is measuring the accuracy of predicting the correct answers from both the PDPs and WSs datasets. However, Trichelair et al. [TEC+18] argue that this evaluation does not measure whether commonsense reasoning was involved in the prediction of the correct answer. For this reason, they propose a different evaluation protocol which additionally includes measures of the accuracy on a changed subset and a consistency score. The changed subset contains WSs with switched antecedents

[6] https://cs.nyu.edu/faculty/davise/papers/WinogradSchemas/collection_ja.pdf
[7] http://www.llf.cnrs.fr/winograd-fr

such as in the following example.

Example 2.1.4:

- **S: Emma did not pass the ball to Janie although she saw that she was open.**
- **S': Janie did not pass the ball to Emma although she saw that she was open.**

The consistency score is the percentage of predicted correct answer after the antecedents have changed. This new evaluation protocol gives a better understanding on how the models perform and opportunity to analyze in more details the achieved results.

2.2 Related Work

Since the publication of the WSC in 2012 many approaches for solving it have been proposed. For a structured overview, we categorized the existing approaches as follows: The first category are the approaches which employ machine learning and in particular deep learning techniques. The second category covers the approaches which use formalized background knowledge and rules. In this section we will discuss some of the proposed solutions from both categories which had notable contributions as well as approaches which achieved state-of-the-art results.

2.2.1 Machine learning approaches

The authors of the WSC were skeptical about the usage of learning methods being sufficient to resolve the WSC. Their concern was that these methods do not employ any reasoning which they believe is essential for solving the WSC. Nevertheless, there have been some implementations which rely on these methods and still achieve relatively good results.

One early contribution towards these approaches is the work by Rahman and Ng [RN12]. Along with their machine learning framework, an additional dataset was provided. This dataset was constructed by undergraduate students and consists of 943 WSs twin sentences divided into training (70%) and test (30%) sets. In the framework, a ranking-based approached was used. In other words, a Ranking Support Vector Machine (SVM) model [Joa02] is trained given pronoun and the two possible antecedents. The model ranks the antecedents and should assign a higher rank to the correct one. After the training phase, the same model is applied to the instances from the test set. Although the evaluation of the framework on the additional dataset showed that 73% of the test set was answered correctly, this approach was not tested on the original Winograd dataset. Nevertheless, the provided additional dataset will later serve for the training and testing of learning models from other proposals.

More recent proposals employ deep learning techniques. Such as the sequence ranking task suggested by Optiz and Frank [JA18]. The problem of choosing one of the possible antecedents for the missing pronoun is translated to a classification task which distinguishes a more preferred solution from a less preferred solution. As a first step, the ambiguous pronoun in the given sentence is substituted with the two possible solutions. For *Example 2.1.1* this would result in the following two sentences:

S1: The trophy does not fit into the brown suitcase because the trophy is too small.

S2: The trophy does not fit into the brown suitcase because the suitcase is too small.

After that, a neural model designed by Optiz and Frank [JA18], called a Siamese neural network, is specified. The model compares the two -representations of the input sentences and a ranking function that ranks them. For the training of the network, the additional dataset provided in [RN12] is used. Since this set is small and in order to prevent the model to memorize the noun phrase candidates, an anonymization technique is used. With this technique, during training the model omits the noun phrase candidates. This forces the model to focus to the rest of the sentence in order to identify a more general meaning. During the evaluation, the test on data which was anonymized had significantly better results than when anonymization was not applied. In contrast to [RN12], the model is tested on both the original Winograd dataset and the test set from the additional dataset. The model achieved 56% accuracy on the Winograd dataset and 63% accuracy on the additional dataset.

The implementation by Liu et al. [LJL$^+$16b] which competed and had the best results at the WSC held in 2016 also used deep neural network classifier to predict the answers. In the proposed framework, named Knowledge Enhanced Embeddings (KEE), three commonsense knowledge bases (ConceptNet [LS04], WordNet [Mil95] and CauseCom [LJL$^+$16a]) were used to extract commonsense knowledge. The extracted knowledge was then incorporated as knowledge constraint during the word embedding training process. An example of such knowledge constraint is the following semantic similarity inequality: *similarity(happy,glad)* $>$ *similarity(happy,sad)*. For the training process the skip-gram model [MCCD13] was used, which learns to predict the context given a target word. The KEE model was used for feature extraction from the PDP test problems. After feature extraction, two different solvers were employed for extracting the correct answer. The first solver relied on an unsupervised method for calculating the semantic similarity between the extracted embeddings and the antecedent candidates. The second solver used the extracted embeddings for supervised training of a neural network classifier. The experiments were initially conducted only on the PDP set and showed that the two solvers combined achieve 66.7%. During the WSC, the KEE model was trained on a small Wikipedia text corpus and the second solver was used, achieving 58.3% accuracy which was the best result at that time.

The method described by Trinh and Le [TL18], uses language models based on neural networks to capture commonsense knowledge. They use unsupervised learning to train neural networks on different large datasets. Based on what was learned during the training phase, the language models are able to assign probabilities to given text. The idea here is to reduce the coreference resolution problem to a decision which relies on probabilities. As a first step, a substitution as in [JA18], explained above, is done on the input sentence with the two possible antecedents. Next, the pre-trained language models assign one of the two different scores, full or partial, to these sentences. The full score is obtained by computing the probability of the substitution on the full sentence while the partial is the probability of the part of the sentence with the special word, conditioned on the substitution in the antecedent. The sentence with the higher probability is assumed to be the one with the correct answer. The conducted experiments showed that the models which assigned partial scoring were the most successful ones. According to the results of the experiments, the language models correctly resolved 70% of the PDP dataset and 63.7% of the WSC dataset. While these are very good results for the WSC, compared to the previous state-of-the-art [LJL+16b], this method relies on learning with no inferential reasoning involved. Thanks to available computational power and the access to large datasets this method achieves very good results. At the same time, the training process of the networks is what makes this method very expensive. Moreover, it has been suggested as one possible extension of the WSC to add a requirement for the system to provide explanations of how the answers have been chosen [MJ15]. This would be a challenge for this particular approach and the other machine learning approaches since no explanation for a reasoning process is provided.

Recently, a similar method which relies on deep learning has been published by Radford et al. [RWC+19]. This method uses an unsupervised learning on a language model which is trained using 1.5 billion parameters and a dataset of 8 million web pages. In order to ensure the quality in the training dataset, a new web scrape was created which scrapes data only from pages with content created and modified by humans, for example blogs. The model, named GPT-2, uses Transformer based architecture [VSP+17] which is an encoder-decoder structure with self-attention layers. Although this method is controversial because the implementation is not publicly available, the authors claim that their language model outperforms the current state-of-the-art in different NLP tasks (reading comprehension, summarization of text, translation). For solving the WSC, they used the GPT-2 model in the same setting as Trinh and Le [TL18] and also achieved better results when applying the partial scoring. The reported accuracy of 70.70% on the WSC corpus is the current state-of-art result.

2.2.2 Knowledge-based approaches

The authors of the WSC are from the Knowledge Representation and Reasoning area, so it is not surprising that they suggest the use of knowledge-based systems as a possible way of addressing the WSC. At the core of the knowledge-based approaches lies the idea of formulating rules which can be used for resolving the missing pronoun. To be able to apply the rules during a reasoning process, structured background knowledge is required.

One of the earliest proposals that rely on this methodology was by Schüller [Sch14] in which Relevance Theory (RT) [WS02] is combined with Knowledge Graphs (KG). Inspired by Schank's model for Knowledge Representation [Sch72], Schüller proposed a framework for combining nodes of KG and reasoning on the resulting graph. In this framework a KG data structure based on the domains for labeling the nodes and labeling the dependencies in the graph is defined. Additionally two sets of constraints and conditions for the KG structure are defined in order to enforce certain linguistic and structural properties. An example of such condition is that all nodes in the graph are of the same type. This data structure is then used for representing the input sentence and the background knowledge. The input sentence represented as a KG and a background KG related to it is activated, after which the two are joined in a third KG. Finally, applying concepts from RT during the reasoning process over the third graph, a resulting graph is extracted which represents the correct solution for the input sentence. Schüller [Sch14] presents the evaluation on 4 WSs. The downside of this approach is that the graphs for the input sentence and for the knowledge have been manually constructed which limits the possibility for evaluation and it misses the core point of the challenge. However, the idea to represent the given Winograd sentence as a dependency graph served later as a starting point to other approaches such as [SVAB15].

In Sharma et al. [SVAB15] the authors recognized two different types of commonsense knowledge which are needed in order to resolve the pronoun in the input sentence. 71 WSs have been divided into two categories: the first category is called the Direct Causal Events and the second one is called Causal Attribute. Furthermore, a non domain-specific semantic parser is developed, called KParser[8]. This parser uses different NLP techniques, such as syntactic dependency parsing, sense disambiguation and discourse parsing for preprocessing the input sentence and produces a semantic graph representation of the sentence. In the next step, a set of queries based on the concepts in the sentences and in the question is created, which is later used to search a large corpus of text. The goal of the search is to automatically extract sentences with relevant commonsense knowledge for the input sentence and represent them as a semantic graph. Finally, different reasoning processes[9] are applied to the sentences from the two

[8] www.kparser.org

[9] The difference is in the predicates that extract nodes from the semantic graphs.

categories and by comparing the graphs of the input sentence and the extracted sentences, the predicted answer is found. During the evaluation of this approach, 53 sentences were answered with 4 of them answered incorrectly. Although in [SVAB15], the focus is on a subset of sentences, the results from the evaluation are promising. This leaves open the possibility to identify more types of commonsense reasoning which would help in the extraction of the background knowledge and the reasoning process.

The work by Sharma and Baral [SB18] follows this direction. Although this approach is currently under review and still not published, we decided to consider it in more detail because it seems to be a promising knowledge-based approach. In contrast to the two types of reasoning identified by Sharma et. al [SVAB15], Sharma and Baral [SB18] present twelve different knowledge types which capture the entire corpus of the WSC. We now present an overview of the steps in this approach and the results of the conducted evaluation. How the twelve knowledge types were identified and which are they will be discussed in the next chapter. Additional contribution of this work is the provided reasoning algorithm which will also be discussed in the next chapter.
Following are the steps in this approach:

1. The given Winograd sentence and question are represented as semantic graphs.
2. Background knowledge is extracted using the same principle as in [SVAB15]. Sentences similar to the input Winograd sentence are retrieved and then represented as semantic graphs. The idea is to find patterns in the graphs which are similar to the identified knowledge types.
3. The semantic graph of the input sentence is merged with the graph representation of the extracted knowledge, which again results in a graph.
4. This resulting graph is then merged with the graph representation of the question.
5. An answer is retrieved by matching the node of the question word from the question graph with the node of the ambiguous pronoun and with an additional node that represents an entity(one of the two possible antecedents) from the resulting graph.

This approach was evaluated on problems from the WSC dataset and from the additional dataset by Rahman and Ng [RN12]. Because the developed reasoning algorithm can handle 10 out of the 12 knowledge types, only those WSs which belong to the first 10 knowledge types were considered. For the problems from the WSC dataset, the knowledge extraction algorithm retrieved relevant knowledge for 100 WSs. The reasoning algorithm was evaluated over those WSs and retrieved correct answers for all of them. For the additional dataset, 138 sentences which are covered by the identified knowledge types were evaluated. The required background knowledge for these problems was encoded manually. In this case, the reasoning algorithm answered correctly 111 WSs, whereas the remaining 27 were answered wrong

because of a parsing error and not a reasoning one. The main drawback of this approach is the missing background knowledge which limits the evaluation part.

A very recent approach, which achieves state-of-the-art results, is a knowledge hunting framework proposed by Emami et al. [ECT+18]. The focus here is on solving all the instances without relying on any existing knowledge bases or statistical methods for capturing the commonsense knowledge. To this end they have developed a framework which is divided into four stages.

1. Using syntactic parsing of the given sentence the antecedent candidates, the verb which connects them as well as the verb phrase which contains the missing pronoun are identified.
2. Based on the identified elements from the first step, queries are generated. In addition, a semantic similarity algorithm is developed which filters out elements from the generated query sets based on their relevance to other words. An example of a query set for *Example 2.1.1* is the following: *{"doesn't fit into", "brown", "fit"}*
3. The results from the search engines are processed and only sentences with a structure similar to the input sentence are kept.
4. Finally, the retrieved sentences are scored according to a scoring system designed by the authors. The scores are assigned to both possible antecedents and the one with the higher score is assumed to be the correct one.

The experiments done during the evaluation of this framework show the a correct answer is given for 119 WSs. Furthermore, the precision, recall and F1 score[10] is calculated and compared with results from previous approaches ([SB16, LJL+17]). The knowledge-hunting framework outperformed all the previous once with achieved higher F1 score and higher recall.

Table 2.1 presents a summary of all the previously analyzed approaches. For each approach, the size of the evaluation dataset and the size of correctly answered problems is expressed with both the number of sentences and what percentage this number is for the used dataset. If the evaluation was not provided for any of the datasets, then there is NA for Not Available. In the last column are remarks for each of the approaches stating their main contribution or main drawback.
From Table 2.1 can be observed that almost all of the approaches are evaluated on different or on smaller datasets. For the machine learning approaches this is because of the small number of available problems, which is also the main disadvantage for these approaches. In the case of the knowledge based approaches, the general drawback is that the test set is directly analyzed and based on these observations the models

[10]These are performance measures, usually used in binary classification tasks. They all rely on confusion matrix in which are presented the outcome of the prediction and the actual values.

Technique	**PDPs** Size | Correct	**WSC** *additional dataset Size | Correct	**Remarks**
Supervised ranking SVM model [RN12]	NA	NA 282* - 30% | 205* - 73%	-provided additional dataset set -no evaluation on WSC dataset
Classification task with NN [JA18]	NA	282 - 100% | 157 - 56% 282* - 30% | 177* - 63%	-first to use substitution of the pronoun with the antecedents
Knowledge Enhanced Embeddings (KEE) [LJL+16b]	60-100% | 40 - 66.7%	NA NA	-best results in the 2016 WSC competition
Google's language model [TL18]	60-100% | 42 - 70%	273 - 100% | 173 - 63.7% NA	-no reasoning involved in the discovery of the correct answer -state-of-the-art for PDPs
OpenAI language model [RWC+19]	NA	273 - 100% | 193 - 70.70% NA	-current state-of-the-art for WSC -requires a lot of data for training -results are not reproducible
Graphs with Relevance theory [Sch14]	NA	4 - 2.6% | 4 - 100% NA	-manual construction of graphs -first representation of WS as dependency graph
2 identified categories [SVAB15]	NA	71 -25% | 49 - 69% NA	-first attempt of identifying commonsense knowledge types -developed the KParser
Semantic relations categories [SB18]	NA	100 - 34%| 100 - 100% 138* - 14%| 111 - 80%	-provided Reasoning Algorithm -identified 12 commonsense types which capture the entire WSC
Knowledge hunting framework [ECT+18]	NA	273 - 100%| 119 - 43.5% NA	-refined query generation -developed an algorithm for scoring the retrieved sentences

Table 2.1: Summary of the analyzed approaches

are build. As a consequence, the size of the evaluation sets is significantly smaller in comparison to the machine learning approaches. Moreover, none of these approaches achieves an accuracy close to 90%, which means there is room for improvement for both machine learning and knowledge-based approaches. One additional remark is that to the best of our knowledge, there is no approach which combines the strengths of both the machine learning and knowledge based approaches.

3 An approach to Commonsense Knowledge Types Identification and Reasoning for the WSC

Intrigued by the achieved high accuracy of the Sharma and Baral [SB18] approach, we now describe this work in more details. In the previous chapter, in section 2.2 we outlined the steps of the approach. Here, we first focus on the process of identification of the presented commonsense knowledge types. Then, we discuss the proposed reasoning algorithm and the provided worked out example. We conclude this chapter with some final observations about this approach.

3.1 Identification of Commonsense Knowledge Types

The first step of the Sharma and Baral [SB18] approach is to represent the input Winograd sentence and question as semantic graphs. To achieve this, the KParser from [SVAB15] is used. To illustrate the results of the semantic parsing, we analyze part of a WS which is shown in *Example 3.1.1*. The resulting graphs are shown in Figure 3.1 and Figure 3.2.

Example 3.1.1

S: The man could not lift his son because he was weak.

Q: Who was weak?

In these graphs, the different colors of the nodes indicate the predefined class of nodes to which they belong. Nodes with red color represent events which correspond to the verbs in the sentence. Nodes with blue color represent entities and qualities of entities, and the nodes with gray color represent conceptual classes. The labels on the directed edges are the semantic relations between the different nodes in the graph. The number next to a word refers to the position of the word in the sentence.

For the representation of the question, one additional conceptual class labeled as "q" is introduced. In this conceptual class are the question words from the WSC problems, such as: *Who, Which and What*.

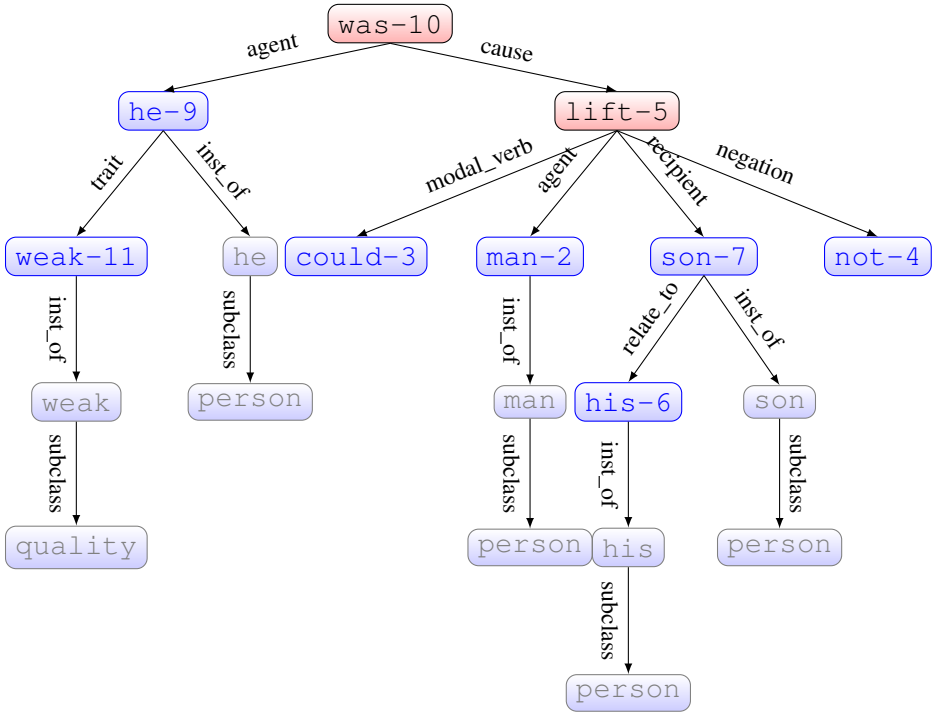

Figure 3.1: “The man couldn’t lift his son because he was so weak."

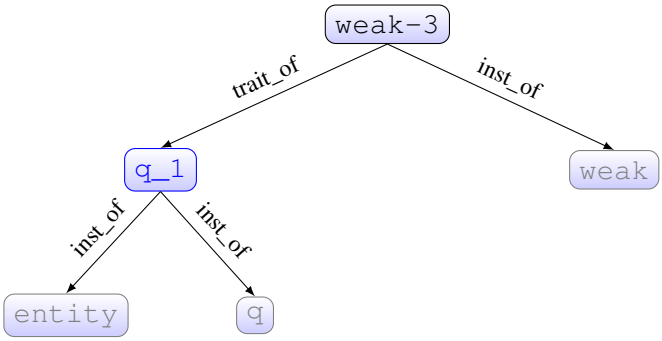

Figure 3.2: “Who was weak?"

Figure 3.3: “Prevents type?"

From the observation of these two graphs, Sharma and Baral [SB18] come to conclusion that the missing knowledge required to answer the question, must connect the trait of an entity being weak with its inability to lift. In Figure 3 is the representation of this knowledge as shown in [SB18].

By applying similar analysis to all problems from the WSC corpus, Sharma and Baral [SB18] identified 12 different knowledge types. From these, the first 10 knowledge types share the same structure as they are all based on different interactions between actions and properties. Each of the 10 knowledge types consists of three parts: the first and third part are sentences containing entities, properties or actions, whereas the second part is a semantic relation (*causes, prevents* or *followed by*) which connects them. The last two knowledge types have different structure because one of them requires multiple knowledge and the other one is based on the conditional likelihood of a previous event. For the representation of the knowledge type shown in Figure 1.3, the first part is *weak y*, the third part is *y lifts* and the semantic relation is *prevents*. The presented knowledge types are shown in Table 3.4.

Knowledge Type	**# of WSs**
1. Property *prevents* Action	16
2. Action1 *prevents* Action2	6
3. Action1 *causes* Action2	41
4. Property *causes* Action	27
5. Action *causes* Property	13
6. Property1 *causes* Property2	4
7. Action1 *followed by* Action2	17
8. Action *followed by* Property	2
9. Property *followed by* Action	112
10. Co-existing Action(s) and Property(s)	2
11. Statement1 *is more likely than* Statement2	26
12. Multiple Knowledge	25

Figure 3.4: Knowledge Types

3.2 Reasoning Algorithm

In this section we explain the proposed logical Reasoning Algorithm (RA) from Sharma and Baral [SB18]. This RA was used for solving the WSC problems by applying the previously explained knowledge types. It takes a formal representation of Winograd sentence and question and additional knowledge as input and it returns an answer to the question. The implementation of the RA together with a running example is available online[1].

The RA is based on Answer Set Programming (ASP) [Lif08]. Therefore, after the Winograd sentence and question have been represented as semantic graphs, the observations from these graphs are encoded as ASP rules. The retrieved answer should be entailed by these input rules. The RA is split into two main phases, each including several steps. The execution of each step results in new ASP rules.

- **Phase 1: Generating merged representation**

  In this phase, the graph representation of the Winograd sentence and the additional knowledge are merged. The idea here is to extend the information from the input sentence with the information from the additional knowledge. To achieve this, the following information from both graph representations is extracted and later merged.

  1. All the nodes which are an instance of a defined class of nodes in the KParser (for example: person, motion, description), are identified as constant nodes. All constant nodes have a directed edge labeled as *instance_of* to another node in the graph.
  2. Constant nodes with parents/children as constant nodes are identified. These are nodes previously identified as constant nodes which have directed edges to other constant nodes.
  3. The cross domain siblings in both representations are identified. These are different constant nodes which appear in both graphs and they both are instances of nodes of the same type.
  4. Nodes which are identical in both graphs are identified. These nodes are called cross domain clones and are of the same type as the cross domain sibling. Additionally, they have the same number of child/parent nodes which are connected through the same semantic relations.
  5. In this last step a merged representation is created. The merged representation is a copy of the sentence's representation, expanded with the additional knowledge by adding edges between nodes from both graphs that have been identified as cross domain clones.

- **Phase 2: Extracting possible answers**

  In this phase, the graph representation of the WSs question is projected onto the merged represen-

[1] https://drive.google.com/file/d/1WN0T98HaMFhWEEIH-3AlWoIPxAdFYlT_/view

tation from the previous step. The goal is to retrieve the nodes which are cross domain clones of the unknown node (q) from the question in the merged representation. Similarly as in the previous phase, information from both graph representations is extracted and merged.

1. Constant nodes are identified from both graphs.
2. Nodes with children/parent as constant node are identified.
3. Cross domain siblings of the nodes from the question graph are identified in the merged representation.
4. Using the previously extracted nodes, the cross domain clones of the nodes from the question graph are identified in the merged representation.
5. Nodes from the merged graph which are cross domain clones to the unknown node (q) from the question graph are retrieved as an answer.

The implementation of the RA with the indicated phases and steps can be is available in AppendixA. According to Sharma and Baral [SB18] this RA can be applied to WSC problems which belong to the first 10 from the identified 12 knowledge types.

Worked-out example To check the work of the RA, we tried the code for the example which was provided together with the implementation of the RA in the supplementary document for Sharma and Baral [SB18]. In this example are the ASP encoding of the sentence and question from *Example 3.1.1*. For extraction of additional knowledge the sentence *"She could not lift it because she is a weak girl"* was used. The extracted knowledge from this sentence is *"weak y prevent y lifts"* and this corresponds to the first knowledge type *"Property prevents Action"*. The ASP encoding of this knowledge is shown in Code 3.1. The predicate name **has_k** indicates that these rules are extracted from the graph representation of the additional knowledge. Similarly, there are predicates **has_s** and **has_q** for rules extracted from the graphs of the input sentence and question. Line 4 in Code 3.1 is the rule which characterizes the corresponding knowledge type for this example. Running the RA with the provided ASP rules for the sentence, question and knowledge type, retrieves as answers: **ans(q_1,he_9)** and **ans(q_1,man_2)**. Hence, the two nodes **he_9** and **man_2** from the merged representation correspond to the unknown node **q_1** from the question. The conclusion is that the correct referent for the ambiguous pronoun **he_9** is the noun **man_2**.

```
1 has_k(weak_1,is_trait_of,y_2).
2 has_k(weak_1,instance_of,weak).
3 has_k(y_2,instance_of,entity).
4 %has_k(weak_1,prevents,lifts_5).
5 has_k(lifts_5,instance_of,lift).
```

```
6 has_k(lifts_5,agent,y_2).
```

Code 3.1: “weak y prevents y lifts"

Observations After trying the provided worked out example and ensuring that the RA retrieved the correct answer, we decided to try out some more WSC problems(#2, #3, #5, #10, #11, #14, #20, #52 in Appendix B). In order to do this, we followed the same steps as for the worked out example. For each of the chosen WSC problems, as a first step we used the KParser to get the graph representation of the input Winograd sentence and question. Next, we encoded the relevant information from the graphs in ASP format. Since the knowledge types for the different WSC problems are not publicly available, we could not know which problem from which knowledge type is. Therefore, we assumed that all the chosen problems correspond to the first knowledge type. For all of the chosen WSC problems, as additional knowledge we used the same rules as in Code 3.1. In these rules, we only changed the traits and the events according to the Winograd sentences. For example, when considering the first half of the WS #20, the trait is *successful* and the event is *envies*. To our surprise, for all of the WSs that we tried the correct answer was retrieved. This lead us to suspect that the rule for the identified knowledge type was not contributing at all to the reasoning procedure. To confirm this, on the worked out example we commented out the rule that defines the knowledge type (Line 4 in Code 3.1) and again the correct answer was retrieved. After trying both halves for all the chosen WSC problems, we came to conclusion that for extracting the answer, the RA relies on hard coded information about the agent and the recipient. For instance, if in Code 3.1, in Line 6 we substitute *agent* with *recipient*, the retrieved answer is then **ans(q_1,he_9)** and **ans(q_1,son_7)**. Therefore, the conclusion of our observation is that the RA does not consider the rule for the knowledge type in the reasoning process. Rather, it relies on the hard coded information about the agent and the recipient of the main event for retrieving an answer.

4 Categorization of Commonsense Knowledge

Motivated by the found shortcoming in the usage of the identified knowledge types with the RA in Sharma and Baral [SB18], we present a conceptual analysis which focuses on different categorization of the WSC problems. As a result of our analysis we present six categories based on the domain of commonsense knowledge in which the required additional knowledge belongs. The idea is to use these categories when defining the additional knowledge such that it will contribute to the reasoning process when extracting the possible correct answer. In this chapter, we first discuss the process of identifying the proposed categories and give short description for each of them. Next, we discuss in more details the semantic representation and ASP encoding of two WSs from one of the categories. After that, we apply additional knowledge defined according to one of the identified categories to the RA from Sharma and Baral [SB18]. Finally, we end this chapter with a general discussion.

4.1 Commonsense domains

As discussed before, for correctly resolving the problems from the WSC corpus, a system needs to be able to mimic commonsense reasoning. To do this, additional background knowledge from different domains is required. In order to determine these domains of knowledge in the WSC corpus, we approached the problems inductively and we identified six categories. Two annotators thoroughly analyzed the entire WSC corpus (Appendix B) and agreed on six different categories of knowledge. Five of these categories are associated with knowledge from a specific domain, whereas the sixth category requires knowledge from multiple domains. In a second process, all WSs were annotated with these previously identified categories. The results of the annotation are presented at the end of Appendix B. The annotation was based on the additional knowledge that is applied in order to answer the question form the WS. To decide which WS belongs to which category, we rely on the context of the sentence and on the adjective or the special word. The identified categories are specific to the WSC corpus and they are related to what was discovered during the analysis done by the annotators. We now explain the six categories in more detail.

Physical To resolve the WSs from this category additional knowledge about some physical feature which is a trait of the ambiguous pronoun is needed.

Emotional This category captures all WSs that include some emotion. In other words, resolving these WSs requires some knowledge about emotions and their characteristics.

Interactions These WSs are related to abstract forms of interaction between the subject and the object in the sentence. In contrast to the WSs from the *Physical* and *Emotional* categories, resolving these WSs does not include knowledge about emotions or characteristics of physical objects.

Comparison In this category are all the WSs where the special words are usually antonyms. The comparison is mostly between the subject and the object in the sentence.

Causal Although many of the WSs express causality, we categorize here only those for which the causality is expressed more explicitly. These sentences contain linking words such as *so* and *because*.

Multiple knowledge Lastly, in this category are all the WSs which do not belong to any of the other categories. For resolving these WSs knowledge from different domains is required.

The annotation of the WSs in some cases was very straightforward and easy to decide. For instance, if we analyze the example for the *Emotional* category in Table 4.1 the special word can be *vindicated* or *crushed*, which we know both are emotions. In addition, the verb *felt*, which is a past form of the verb *to feel*, is an emotion indicator. To answer the question “Who was the winner of the competition?", a deeper understanding of both emotions is required. Furthermore, knowledge about which emotions are related to winning is needed. Overall, the reasoning process for answering the questions in this category involves knowledge about emotions.

In comparison, the annotation of the WSs from the *Interactions* category was challenging. The WSs in this category are about inter-human relationships, often between the subject and the object in the sentence. To solve a WS from this category, applying knowledge about more abstract activities is necessary. Some examples for such activities are: to cooperate, to promise, to convince, to refuse, etc. That is to say, when resolving these WSs there is no reasoning about emotions, comparison or exchange of any physical objects involved. The sentence given as an example from this category in Table 4.1 is annotated with category *Interactions* because of the special word. *Giving/Receiving* help is an abstract form of

Category	Example
1. Physical	**S:** John couldn't see the stage with Billy in front of him because he is so **[short/tall]**. **Q:** Who is so [short/tall]?
2. Emotional	**S:** Frank felt **[vindicated/crushed]** when his longtime rival Bill revealed that he was the winner of the competition. **Q:** Who was the winner of the competition?
3. Interactions	**S:** Joan made sure to thank Susan for all the help she had **[given/received]**. **Q:** Who had [given/received] help?
4. Comparison	**S:** Joe's uncle can still beat him at tennis, even though he is 30 years **[older/younger]**. **Q:** Who is [older/younger]?
5. Causal	**S:** Pete envies Martin **[because/although]** he is very successful. **Q:** Who is very successful?
6. Multiple knowledge	**S:** Sam and Amy are passionately in love, but Amy's parents are unhappy about it, because they are **[snobs/fifteen]**. **Q:** Who are [snobs/fifteen]?

Table 4.1: Types of knowledge

interaction between two different entities. This activity cannot be categorized as physical, emotional, causal or as comparison.

Additionally, these categories can be divided further. Since in the *Physical* category we have the highest number of WSs, we will analyze this category in more details. The WSs from this category, contain different physical features which characterize the ambiguous pronoun. After carefully analyzing these features we identified five subcategories: *size, height/width, weight, space* and *time*.

In the *size* subcategory is the WS from *Example 2.1.1*, which was discussed in chapter 2. The activity in the sentence from this WS depends on the physical trait of the pronoun *it*.

S: The trophy does not fit into the brown suitcase because it is too small.

This physical trait is also the special word which can be *small* or *large*. For resolving these WSs, knowledge about the characteristics of an object of a certain size is required. For *Example 2.1.1*, this knowledge should capture that a small object cannot fit into a large one, while a large one can fit small object.

Similar to this subcategory are the *height/width* and the *weight* subcategories. The WSs in these subcategories have a special word which characterize the ambiguous pronoun with certain height, width or

weight such as *short/tall*, *wide/narrow*. Since these are all similar traits, we assume that their characterizations can be formalized in a similar way.

For solving the WSs from the *space* subcategory, spatial reasoning and in some cases spatial-temporal reasoning is needed. Since the knowledge required for resolving the WSs from this subcategory has to include rules for navigating and understanding time and space, we consider the knowledge needed for this category to be the hardest one to formalize among the subcategories of the *Physical* category. For example, consider the following WS:

Example 3.1

S: Tom threw his schoolbag down to Ray after he reached the [top/bottom] of the stairs.

Q: Who reached the [top/bottom] of the stairs?

A: Tom/Ray

The sentence in *Example 3.1* contains information about a specific location in space which can be *top of the stairs* or *bottom of the stairs*. In addition, it states something about an event which happened *after* a certain event. Therefore, answering the question from this example requires applying reasoning about both space and time. Other WSs are simpler, where only knowledge about space is needed. These WSs have a special words such as: *above/bellow, in/out*.

The WSs from the *time* subcategory require temporal reasoning. Formalization of knowledge about change over time and chronology of events is needed for solving these WSs. For annotating these WSs we relied on the context and we identified events which happened at different points in time. The following is an example of WS that belongs to this subcategory:

Example 3.2

S: Thomson visited Cooper's grave in 1765. At that date he had been [dead/travelling] for five years.

Q: Who had been [dead/travelling] for five years?

A: Cooper/Thomson

Evaluation To evaluate the agreement of the annotation we used Cohen's kappa statistic [Coh60], which is a measure for an inter-rater agreement. Namely, Cohen's kappa measures the agreement between two annotators regarding the annotation of N items with C mutually exclusive categories. It is defined as follows:

$$\kappa = (p_o - p_e)/(1 - p_e) \tag{4.1}$$

where p_o is the observed accuracy among the annotators and p_e is the expected accuracy when both annotators assign categories randomly. Each of the two annotators analyzed all the problems from the WSC corpus and annotated them with one of the identified six categories. For the calculation of the Cohen's kappa score, we used the sklearn metrics package[1] that includes an implementation of Cohen's kappa.

Results The calculated Cohen's kappa score for the annotation of the WSs is 0.66. According to the description for relative strength of agreement from Landis and Koch [LGK77], this score is interpreted as substantial agreement. In our case this means, that both annotators largely agree on the annotation of WSs with the provided six categories.

Category	**Annotator 1** # - %	**Annotator 2** # - %
Physical	36 - %24	39 - %26
Emotions	7 - %5	9 - %6
Interactions	44 - %29	24 - %36
Comparison	19 - %13	26 - %17
Causal	16 - %11	18 - %12
Multiple knowledge	28 - %19	34 - %23

Table 4.2: Annotation results

The numbers of WSs and the percentage for each category, as annotated by the two annotators, are shown in Table 4.2. The result of the evaluation of the WSs annotation indicate that all of the identified categories were recognized by the two annotators. Although not a large number of WSs are in the *Emotions* and *Causal* categories, there is a strong agreement between the annotators about the WSs belonging in these two categories. For the *Causal* category, this indicates that even though many of the WSs seem causal, it is not so difficult to discriminate the sentences in which the cause and effect are explicit. The *Physical* category contains largest number of WSs for which both annotators agree. Indeed, the observation of the semantic graphs for WSs from this category showed that there is a similarity in the knowledge that is represented with these WSs. As discussed earlier, the identification of the WSs which belong in the *Interactions* category was quite difficult. This is clearly shown in the large difference between the annotators in the annotation results. While one annotator placed 44 WSs in this category, the

[1] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.cohen_kappa_score

other annotator 20 of those 44 WSs distributed in all other five categories. Therefore, we can conclude that the identified categories are not exclusive, i.e. some of the WSs can belong to more than one category.

4.2 Semantic representation and ASP encoding

As the *Physical* category was the one on which both annotators agreed on, we will further investigate two WSs that belong to this category. Namely, we observe the WSs from *Example 2.1.1* and *Example 3.1.1*. We use the KParser for translating the input sentence and question into a graph representation. The two semantic graphs[2], are shown in Figure 4.1 and Figure 4.2. We can observe that the semantic representations of the sentences are indeed very similar. Figure 4.1 represents the semantic graph for the sentence:

S1: The trophy does not fit into the brown suitcase because it is too small.

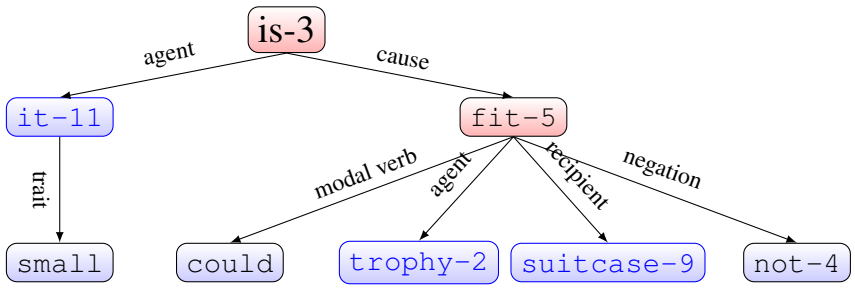

Figure 4.1: “The trophy doesn’t fit into the brown suitcase because it’s too small."

Figure 4.2 shows the semantic graph for the sentence:

S2: The man couldn’t lift his son because he was so weak.

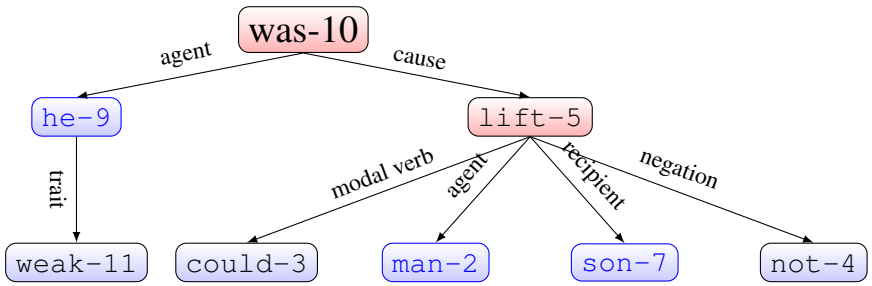

Figure 4.2: “The man couldn’t lift his son because he was so weak."

In both representations, the main event (*fit, lift*) is connected with the agent and the recipient of the action as well as with the modifiers of the action. Additionally, the ambiguous pronoun (*it, he*), associated with a physical trait (*small, weak*), is represented as the cause of the main action.

[2]Slightly simplified such that only relevant information is represented.

We are interested to find out whether we can improve the reasoning procedure from Sharma and Baral [SB18] by applying rules based on the category of the WS. For this purpose, for both WSs from *Example 2.1.1* and *Example 3.1.1*, the input sentences, questions and additional knowledge were encoded in ASP. Because we use the RA, we used the same predicate names as in the worked-out example which was discussed in section 3.2. As an example of these ASP rules, part of the extracted rules for the WS from *Example 2.1.1* are presented in Code 4.1 and Code 4.2. Since both of the WSs that we are testing are from the *Physical* category, as additional knowledge we provide a rule that characterizes the physical trait from the sentences.

In Code 4.1 the rules[3] extracted from the graph in Figure 4.1 are shown. For extracting these rules we followed the steps provided for the worked-out example. However, it is not difficult to notice that the information encoded in the ASP rules comes straightforward from observing the graph representation of the sentence. In Code 4.2 the rules which are used as additional knowledge for this example are shown. The rule in Line 1 is the rule which characterizes the physical trait of the ambiguous pronoun. We encode the trait of being small as trait of an object and we have no knowledge that this object can fit something else. To explain this further, as additional knowledge we encode the sentence: *"Small object(s) could not fit"* . In the same manner the ASP rules for the WSs from *Example 3.1.1* are defined. As additional knowledge for this example we encode the sentence: *"Weak entitie(s) could not lift"*.

```
1 has_s(it_11,instance_of,object).
2 has_s(small_14,is_trait_of,it_9).
3 has_s(small_14,modifier,too_13).
4 has_s(fit_5,agent,trophy_2).
5 has_s(fit_5,recipient,suitcase_9).
6 has_s(suitcase_9,instance_of,object).
7 has_s(trophy_2,instance_of,object).
```

Code 4.1: Knowledge from S2

```
1 has_k(small_1,is_trait_of,y_2) :- has_k(fits_5, recipient, y_2), not has_k(fits_5,modifier,could_3).
2 has_k(y_2,instance_of,entity).
3 has_k(fits_5,recipient,y_2).
```

Code 4.2: Additional knowledge

When running the RA with our encoding of the additional knowledge, for both examples the correct answer was retrieved. We also considered the twin sentences with the switched special word, and the RA again retrieved the correct answer. Additionally, when the rule describing the background knowledge is removed, there is no answer was retrieved. Furthermore, in Line 4 when we substitute *recipient* with

[3]Simplified such that only rules with relevant information are shown.

agent, again no answer is retrieved. This means that when the additional knowledge is formalized as in Code 4.2, it prevents the RA to rely on hard coded information.

4.3 Discussion

An initial objective of this project was to identify the main challenge that needs to be resolved in order to tackle the WSC. From studying the existing approaches, we recognized the need for having a formal representation of a specific background knowledge during the reasoning process in the knowledge-based approaches. To this end, we proposed a categorization of the sentences for which similar knowledge is required for sentences belonging in the same category.

The identified categories differ from the ones presented by Sharma and Baral [SB18] in the way they are defined. We rely on the context of the WS sentences, whereas in [SB18] the categorization is based on the structure of the sentence. Namely, we analyzed the entire WSC corpus and identified what is the commonsense knowledge that is needed for answering the question and from which category it would be. In Sharma and Baral [SB18] the categories are identified using a more general approach that is, they all have the same structure *X prevents/follows/causes Y*.

One unanticipated finding was that the rule-based RA presented in Sharma and Baral [SB18] does not work as expected. What came as a surprise was that the correct answer was retrieved even when rule describing the knoledge type was removed. Intrigued by this result, we analyzed more closely the Reasoning Algorithm. What we discovered is that the answer returned depends on whether in the background knowledge there is information about the *agent* or the *recipient* of the action.

Since the Reasoning Algorithm is well encoded, we decided to change only the encoding of the background knowledge so it would capture the characteristics to one of our identified categories. Having these rule in the background knowledge and then running the RA returns the correct answer. In contrast to the rule from the example in Sharma and Baral [SB18], when the rule that formalizes the physical trait is removed from the background knowledge, no answer is retrieved.

A good semantic graph representation of the WS is of high importance for correctly formalizing the available knowledge. To understand better the choice of *agent* and the *recipient* for the WS sentences, we analyzed more closely the work of the KParser. Although in most of the sentences that we parsed a correct graph representation was returned, in some of them there were many inconsistency. For example consider the following sentence:

S: Joan made sure to thank Susan for all the help she had given.

When parsed with the KParser, the semantic graph representation of this sentence consists of three disconnected graphs. Moreover, in the case when there are two consequent sentences in one WS the result from the KParser is too complicated for analyzing, let alone for formalizing in ASP.

5 Conclusion and Future Work

Conclusion In this report we presented a detail analysis of the WSC. We introduced the main components of a WS and identified the need for additional, background knowledge as a main challenge for correctly resolving one. We presented an extensive literature review on the existing different proposals and their contributions towards solving the WSC. Looking into the different approaches, we highlighted that neither the achine learning based nor the knowledge-based approaches alone are sufficient for achieving high accuracy on the WSC. The detailed analysis of one of the promising knowledge-based approaches ([SB18]), reviled a flaw in the formalization of the background knowledge used in the reasoning procedure. Therefore, with the intention to improve the process of extracting and formalizing additional relevant knowledge, we analyzed in more details the WSC problems and identified different categories of WSs. The categories that we presented can enhance the work on both the machine learning and the knowledge-based approaches or can be used as a bridge between them. We found the process of identification of the categories and deciding the right category for the WSs to be quite challenging. It is not surprising that so far too little attention has been paid to the content of the WSs. In addition, we used an existing reasoning algorithm, which when adapted as for one of the identified categories retrieves correct answers for WSs. It is important to note that our work presented here is an analysis of the WSC and not an attempt towards solving it. Moreover, due to the limited number of WSC problems, the presented categories were identified in a “backward-engineering" manner, that is by considering directly the test set.

Future Work To support the assumption that identifying the category of a WS can improve the extraction of relevant additional knowledge, the characteristics of the meta-concepts from each category ought to be formalized. Analyzing the semantic graphs for the WS sentences could be a good starting point for recognizing these characteristics. Afterwards, these formalized concepts need to be implemented and tested in an appropriate reasoning algorithm. Another interesting extension would be to explore an implementation of knowledge-enhanced neural networks with commonsense knowledge from the different categories and test their performance. Recently, as proposed by Ma et al. [MPC18], applying knowledge injection during the training of deep neural networks can lead to improvement of the result of the neural network. Using commonsense knowledge databases, different neural networks can be pre-trained with

knowledge from a specific category. After analyzing the Winograd input sentence, a network trained with the knowledge from the identified category can be used in the process of extracting relevant background knowledge. Additionally, formalized rules describing the characteristics of a category can be fed to neural networks before training [RDG18]. In this way, the rules would be a guidance for the network during the training phase. Not only could this speed up the training of the network by reducing the amount of required data, but might also support explanations for the predicted answer. An approach that exploits the advantage of formalized rules, capturing characteristics of different areas of background knowledge, can possibly merge together the strengths of both categories of approaches, machine learning and knowledge-based Systems.

References

[AW72] Terry Allen Winograd. Understanding natural language. *Cognitive Psychology*, 3:1–191, 01 1972.

[Ben15] David Bender. Establishing a human baseline for the winograd schema challenge. In *Proceedings of the 26th Modern AI and Cognitive Science Conference 2015, Greensboro, NC, USA, April 25-26, 2015.*, pages 39–45, 2015.

[Coh60] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46, 1960.

[Cry11] David Crystal. A dictionary of linguistics and phonetics, sixth edition. *The Modern Language Journal*, 76:25–26, 01 2011.

[Dav16] Ernest Davis. Winograd schemas and machine translation. *CoRR*, abs/1608.01884, 2016.

[ECT+18] Ali Emami, Noelia De La Cruz, Adam Trischler, Kaheer Suleman, and Jackie Chi Kit Cheung. A knowledge hunting framework for common sense reasoning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 1949–1958, 2018.

[JA18] Opitz Juri and Frank Anette. Addressing the winograd schema challenge as a sequence ranking task. In *Proceedings of the First International Workshop on Language Cognition and Computational Models*, pages 41–52. Association for Computational Linguistics, 2018.

[Joa02] Thorsten Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, July 23-26, 2002, Edmonton, Alberta, Canada*, pages 133–142, 2002.

[LDM12] Hector J. Levesque, Ernest Davis, and Leora Morgenstern. The winograd schema challenge. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Thirteenth International Conference, KR 2012, Rome, Italy, June 10-14, 2012*, 2012.

[Len95] Douglas B. Lenat. CYC: A large-scale investment in knowledge infrastructure. *Commun. ACM*, 38(11):32–38, 1995.

[LGK77] J Landis and Gary G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33:159–74, 04 1977.

[Lif08] Vladimir Lifschitz. What is answer set programming? In *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence, AAAI 2008, Chicago, Illinois, USA, July 13-17, 2008*, pages 1594–1597, 2008.

[LJL+16a] Quan Liu, Hui Jiang, Zhen-Hua Ling, Si Wei, and Yu Hu. Probabilistic reasoning via deep learning: Neural association models. *CoRR*, abs/1603.07704, 2016.

[LJL+16b] Quan Liu, Hui Jiang, Zhen-Hua Ling, Xiaodan Zhu, Si Wei, and Yu Hu. Combing context and commonsense knowledge through neural networks for solving winograd schema problems. *CoRR*, abs/1611.04146, 2016.

[LJL+17] Quan Liu, Hui Jiang, Zhen-Hua Ling, Xiaodan Zhu, Si Wei, and Yu Hu. Combing context and commonsense knowledge through neural networks for solving winograd schema problems. In *2017 AAAI Spring Symposia, Stanford University, Palo Alto, California, USA, March 27-29, 2017*, 2017.

[LPC+11] Heeyoung Lee, Yves Peirsman, Angel Chang, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. Stanford's multi-pass sieve coreference resolution system at the conll-2011 shared task. In *Conference on Natural Language Learning (CoNLL) Shared Task*, 2011.

[LS04] Hugo Liu and Push Singh. Conceptnet—a practical commonsense reasoning tool-kit. *BT technology journal*, 22, 06 2004.

[McC60] John McCarthy. Programs with common sense. Technical report, Cambridge, MA, USA, 1960.

[MCCD13] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013.

[MDJ16] Leora Morgenstern, Ernest Davis, and Charles L. Ortiz Jr. Planning, executing, and evaluating the winograd schema challenge. *AI Magazine*, 37(1):50–54, 2016.

[Mil95] George A. Miller. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, 1995.

[MJ15] Leora Morgenstern and Charles L. Ortiz Jr. The winograd schema challenge: Evaluating progress in commonsense reasoning. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA.*, pages 4024–4026, 2015.

[MMS+02] John McCarthy, Marvin Minsky, Aaron Sloman, Leiguang Gong, Tessa A. Lau, Leora Morgenstern, Erik T. Mueller, Doug Riecken, Moninder Singh, and Push Singh. An architecture of diversity for commonsense reasoning. *IBM Systems Journal*, 41(3), 2002.

[MPC18] Yukun Ma, Haiyun Peng, and Erik Cambria. Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive LSTM. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5876–5883, 2018.

[RDG18] Soumali Roychowdhury, Michelangelo Diligenti, and Marco Gori. Image classification using deep learning and prior knowledge. In *The Workshops of the The Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, Louisiana, USA, February 2-7, 2018.*, pages 336–343, 2018.

[RN12] Altaf Rahman and Vincent Ng. 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, EMNLP-CoNLL 2012, July 12-14, 2012, Jeju Island, Korea*, pages 777–789, 2012.

[RWC+19] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. `https://github.com/openai/gpt-2`, 2019. [Online; accessed 04-03-2019].

[SB16] Arpit Sharma and Chitta Baral. Automatic extraction of events-based conditional commonsense knowledge. In *Knowledge Extraction from Text, Papers from the 2016 AAAI Workshop, Phoenix, Arizona, USA, February 12, 2016.*, 2016.

[SB18] Arpit Sharma and Chitta Baral. Commonsense knowledge types identification and reasoning for the winograd schema challenge. `https://www.semanticscholar.org/paper/Commonsense-Knowledge-Types-Identification-and-for/1151e8ceafdd292fbf70db5cbca20a805a3ecacb`, 2018. [Online; accessed 25-02-2019].

[Sch72] Roger C. Schank. Conceptual dependency: A theory of natural language understanding. *Cognitive Psychology*, 3(4):552 – 631, 1972.

[Sch14] Peter Schüller. Tackling winograd schemas by formalizing relevance theory in knowledge graphs. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Fourteenth International Conference, KR 2014, Vienna, Austria, July 20-24, 2014*, 2014.

[SVAB15] Arpit Sharma, Nguyen Ha Vo, Somak Aditya, and Chitta Baral. Towards addressing the winograd schema challenge - building and using a semantic parser and a knowledge hunting module. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, pages 1319–1325, 2015.

[TEC+18] Paul Trichelair, Ali Emami, Jackie Chi Kit Cheung, Adam Trischler, Kaheer Suleman, and Fernando Diaz. On the evaluation of common-sense reasoning in natural language understanding. *CoRR*, abs/1811.01778, 2018.

[TL18] Trieu H. Trinh and Quoc V. Le. A simple method for commonsense reasoning. *CoRR*, abs/1806.02847, 2018.

[VSP+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 6000–6010, 2017.

[WS02] Deirdre Wilson and Dan Sperber. Relevance theory. In L. Horn and G. Ward, editors, *The Handbook of Pragmatics*, pages 607–632. Blackwell, 2002.

A Reasoning Algorithm

Implementation in ASP of the reasoning algorithm provided by Sharma and Baral [SB18].

```
%Phase 1: Generating merged representation

%Defining domain of variables
k_val(X) :- has_k(X,R,Y).
k_val(Y) :- has_k(X,R,Y).

%Extracting constant nodes from graphical representation of
%sentence, question and knowledge (Step 1)
s_const(X) :- has_s(X,instance_of,I).
q_const(X) :- has_q(X,instance_of,I).
k_class(X) :- has_k(A,instance_of,X).
k_const(X) :- not k_class(X), k_val(X).

%Extracting constant nodes which has constant parent nodes from
%graphical representations of sentence, question and knowledge (Step 2)
s_has_par(X) :- has_s(P,R,X), s_const(X), s_const(P).
q_has_par(X) :- has_q(P,R,X), q_const(X), q_const(P).
k_has_par(X) :- has_k(P,R,X), k_const(X), k_const(P).

%Extracting constant nodes which has constant children nodes from
%graphical representations of sentence, question and knowledge. (Step 2)
s_has_child(X) :- has_s(X,R,C), s_const(X), s_const(C).
q_has_child(X) :- has_q(X,R,C), q_const(X), q_const(C).
k_has_child(X) :- has_k(X,R,C), k_const(X), k_const(C).

%Extracting cross-domain siblings from a knowledge representation
%to a sentence representation. (Step 3)
not_k_s_crossdom_sib(X,Y) :- has_k(X,instance_of,I1),
                             has_k(X,instance_of,I2),
                             has_s(Y,instance_of,I1),
                             not has_s(Y,instance_of,I2),
                             I1!=I2.
not_k_s_crossdom_sib(X,Y) :- has_k(X,instance_of,I1),
                             has_k(X,instance_of,I2),
                             I1!=I2,
                             not has_s(Y,instance_of,I1),
                             has_s(Y,instance_of,I2).
```

```
not_k_s_crossdom_sib(X,Y) :- has_k(X,instance_of,I1),
                             has_k(X,instance_of,I2),
                             I1!=I2,
                             not has_s(Y,instance_of,I1),
                             not has_s(Y,instance_of,I2),
                             s_const(Y).
k_s_crossdom_sib(X,Y) :- has_s(Y,instance_of,I),
                         has_k(X,instance_of,I),
                         s_const(Y),
                         k_const(X),
                         not not_k_s_crossdom_sib(X,Y).

%Extracting cross-domain clones from a knowledge representation
%to a sentence representation. (Step 4)
k_s_crossdom_clone(X,Y) :- k_s_crossdom_sib(X,Y),
                           has_k(Pj,Rj,X),
                           has_s(Pj_prime,Rj,Y),
                           k_s_crossdom_clone(Pj,Pj_prime),
                           k_const(Pj),
                           has_k(X,Rk,Cj),
                           has_s(Y,Rk,Cj_prime),
                           k_s_crossdom_sib(Cj,Cj_prime),
                           k_const(Cj).
k_s_crossdom_clone(X,Y) :- k_s_crossdom_sib(X,Y),
                           not k_has_par(X),
                           has_k(X,Rk,Cj),has_s(Y,Rk,Cj_prime),
                           k_s_crossdom_sib(Cj,Cj_prime),
                           k_const(Cj).
k_s_crossdom_clone(X,Y) :- k_s_crossdom_sib(X,Y),
                           has_k(Pj,Rj,X),
                           has_s(Pj_prime,Rj,Y),
                           k_s_crossdom_clone(Pj,Pj_prime),
                           k_const(Pj),
                           not k_has_child(X).
k_s_crossdom_clone(X,Y) :- k_s_crossdom_sib(X,Y),
                           not k_has_par(X),
                           not k_has_child(X).

%Generating a merged representation of a sentence and a knowledge. (Step 5)
has_m(X,R,Y) :- has_s(X,R,Y).
k_covered(X) :- k_const(X),
                s_const(Y),
                k_s_crossdom_clone(X,Y).
k_not_all_covered :- k_const(X),
                     not k_covered(X).
k_all_covered :- not k_not_all_covered.
                 has_m(X,R,Y) :- has_s(X,R,Y1),
                 has_s(X2,R2,Y),
                 Y1!=Y,
```

```
                        k_s_crossdom_clone(Y_k,Y1),
                        k_s_crossdom_clone(Y_k,Y),
                        k_all_covered.
has_m(X,R,Y) :- has_s(X1,R,Y),
                        has_s(X,R2,Y2),
                        X1!=X,
                        k_s_crossdom_clone(X_k,X1),
                        k_s_crossdom_clone(X_k,X),
                        k_all_covered.

%Phase 2: Extracting possible answers

%Extracting constant nodes from the merged representation. (Step 1)
has_const(X) :- has_m(X,instance_of,I).

%Extracting constant nodes which has constant parent nodes
%from a merged representation. (Step 2)
m_has_par(X) :- has_m(P,R,X), m_const(X), m_const(P).

%Extracting constant nodes which has constant children nodes
%from a merged representation. (Step 2)
m_has_child(X) :- has_m(X,R,C), m_const(X), m_const(C).

%Extracting cross-domain siblings from a question representation
%to a merged representation. (Step 3)
not_q_m_crossdom_sib(X,Y) :- has_q(X,instance_of,I1),
                        has_q(X,instance_of,I2),
                        has_m(Y,instance_of,I1),
                        not has_m(Y,instance_of,I2),
                        I1!=I2, I1!=q, I2!=q.
not_q_m_crossdom_sib(X,Y) :- has_q(X,instance_of,I1),
                        has_q(X,instance_of,I2),
                        not has_m(Y,instance_of,I1),
                        has_m(Y,instance_of,I2),
                        I1!=I2, I1!=q, I2!=q.
not_q_m_crossdom_sib(X,Y) :- has_q(X,instance_of,I1),
                        has_q(X,instance_of,I2),
                        not has_m(Y,instance_of,I1),
                        not has_m(Y,instance_of,I2),
                        m_const(Y),
                        I1!=I2, I1!=q, I2!=q.
q_m_crossdom_sib(X,Y) :- has_m(Y,instance_of,I),
                        has_q(X,instance_of,I),
                        not not_q_m_crossdom_sib(X,Y),
                        m_const(Y),
                        q_const(X).

%Extracting cross-domain clones from a question representation
%to a merged representation. (Step 4)
```

```
q_m_crossdom_clone(X,Y) :- q_m_crossdom_sib(X,Y),
                           has_q(Pj,Rj,X),
                           has_m(Pj_prime,Rj,Y),
                           q_m_crossdom_clone(Pj,Pj_prime),
                           q_const(Pj),has_q(X,Rk,Cj),
                           has_m(Y,Rk,Cj_prime),
                           q_m_crossdom_sib(Cj,Cj_prime),
                           q_const(Cj).
q_m_crossdom_clone(X,Y) :- q_m_crossdom_sib(X,Y),
                           not q_has_par(X),
                           has_q(X,Rk,Cj),
                           has_m(Y,Rk,Cj_prime),
                           q_m_crossdom_sib(Cj,Cj_prime),
                           q_const(Cj).
q_m_crossdom_clone(X,Y) :- q_m_crossdom_sib(X,Y),
                           has_q(Pj,Rj,X),
                           has_m(Pj_prime,Rj,Y),
                           q_m_crossdom_clone(Pj,Pj_prime),
                           q_const(Pj),
                           not q_has_child(X).
q_m_crossdom_clone(X,Y) :- q_m_crossdom_sib(X,Y),
                           not q_has_par(X),
                           not q_has_child(X).

%Extracting the answers to the input question. (Step 5)
q_covered(X) :- q_const(X),
                           m_const(Y),
                           q_m_crossdom_clone(X,Y).
                           q_not_all_covered :-
                           not q_covered(X),
                           q_const(X).
q_all_covered :- not q_not_all_covered.
ans(Q,X) :- q_m_crossdom_clone(Q,X),
                           has_q(Q,instance_of,q),
                           q_all_covered.

%Making answers visible in the terminal
 #show ans/2.
```

B Collection of Winograd Schemas and Annotation Results

1. The city councilmen refused the demonstrators a permit because they [feared/advocated] violence. Who [feared/advocated] violence? Answers: The city councilmen/the demonstrators.
2. The trophy doesn't fit into the brown suitcase because it's too [small/large]. What is too [small-/large]? Answers:The suitcase/the trophy.
3. Joan made sure to thank Susan for all the help she had [given/received]. Who had [given/received] help? Answers: Susan/Joan.
4. Paul tried to call George on the phone, but he wasn't [successful/available]. Who was not [successful/available]? Answers: Paul/George
5. The lawyer asked the witness a question, but he was reluctant to [answer/repeat] it. Who was reluctant to [answer/repeat] the question? Answers: The witness/the lawyer.
6. The delivery truck zoomed by the school bus because it was going so [fast/slow]. What was going so [fast/slow]? Answers: The truck/the bus
7. Frank felt [vindicated/crushed] when his longtime rival Bill revealed that he was the winner of the competition. Who was the winner of the competition? Answers: Frank/Bill
8. The man couldn't lift his son because he was so [weak/heavy]. Who was [weak/heavy]? Answers: The man/the son.
9. The large ball crashed right through the table because it was made of [steel/styrofoam]. What was made of [steel/styrofoam]? Answers: The ball/the table.
10. John couldn't see the stage with Billy in front of him because he is so [short/tall]. Who is so [short/tall]? Answers: John/Billy.
11. Tom threw his schoolbag down to Ray after he reached the [top/bottom] of the stairs. Who reached the [top/bottom] of the stairs? Answers:Tom/Ray.

12. Although they ran at about the same speed, Sue beat Sally because she had such a [good/bad] start. Who had a [good/bad] start? Answers: Sue/Sally.

13. The sculpture rolled off the shelf because it wasn't [anchored/level]. What wasn't [anchored/level]? Answers: The sculpture/the shelf.

14. Sam's drawing was hung just above Tina's and it did look much better with another one [below/above] it. Which looked better? Answers: Sam's drawing/Tina's drawing.

15. Anna did a lot [better/worse] than her good friend Lucy on the test because she had studied so hard. Who studied hard? Answers: Anna/Lucy

16. The firemen arrived [after/before] the police because they were coming from so far away. Who came from far away? Answers: The firemen/the police.

17. Frank was upset with Tom because the toaster he had [bought from/sold to] him didn't work. Who had [bought/sold] the toaster? Answers: Frank/Tom.

18. Jim [yelled at/comforted] Kevin because he was so upset. Who was upset?

19. Answer : Jim/Kevin.The sack of potatoes had been placed [above/below] the bag of flour, so it had to be moved first. What had to be moved first? Answers: The sack of potatoes/the bag of flour.

20. Pete envies Martin [because/although] he is very successful. Who is very successful? Answers: Martin/Pete.

21. I was trying to balance the bottle upside down on the table, but I couldn't do it because it was so [top-heavy/uneven]. What was [top-heavy/uneven]? Answers: the bottle/the table.

22. I spread the cloth on the table in order to [protect/display] it. To [protect/display] what? Answers: the table/the cloth.

23. The older students were bullying the younger ones, so we [rescued/punished] them. Whom did we [rescue/punish]? Answers: The younger students/the older students.

24. I poured water from the bottle into the cup until it was [full/empty]. What was [full/empty]? Answers: The cup/the bottle.

25. Susan knows all about Ann's personal problems because she is [nosy/indiscreet]. Who is [nosy/indiscreet]? Answers: Susan/Anne.

26. Sid explained his theory to Mark but he couldn't [convince/understand] him. Who did not [convince/understand] whom? Answer Pair A: Sid did not convince Mark/Mark did not convince Sid. Answer Pair B: Sid did not understand Mark/Mark did not understand Sid.

27. Susan knew that Ann's son had been in a car accident, [so/because] she told her about it. Who told the other about the accident? Answers: Susan/Ann.

28. Joe's uncle can still beat him at tennis, even though he is 30 years [older/younger]. Who is [older/younger]? Answers:Joe's uncle/Joe.

29. The police left the house and went into the garage, [where/after] they found the murder weapon. Where did they find the murder weapon? Answers: In the garage/in the house.

30. The painting in Mark's living room shows an oak tree. It is to the right of [the bookcase/a house]. What is to the right of [the bookcase/a house]? Answers: The painting/the tree.

31. There is a gap in the wall. You can see the garden [through/behind] it. You can see the garden [through/behind] what? Answers: The gap/the wall.

32. The drain is clogged with hair. It has to be [cleaned/removed]. What has to be [cleaned/removed]? Answers: The drain/the hair.

33. My meeting started at 4:00 and I needed to catch the train at 4:30, so there wasn't much time. Luckily, it was [short/delayed], so it worked out. What was [short/delayed]? Answers: The meeting/the train.

34. There is a pillar between me and the stage, and I can't [see/see around] it. What can't I [see/see around]? Answers: The stage/the pillar.

35. They broadcast an announcement, but a subway came into the station and I couldn't [hear/hear over] it. What couldn't I [hear/hear over]? Answers: The announcement/the subway

36. In the middle of the outdoor concert, the rain started falling, [and/but] it continued until 10. What continued until 10? Answers: The rain/the concert.

37. I used an old rag to clean the knife, and then I put it in the [drawer/trash]. What did I put in the [drawer/trash]? Answers: The knife/the rag.

38. Ann asked Mary what time the library closes, [but/because] she had forgotten. Who had forgotten? Answers:Mary/Ann.

39. I took the water bottle out of the backpack so that it would be [lighter/handy]. What would be [lighter/handy]? Answers: The backpack/the bottle.

40. I couldn't put the pot on the shelf because it was too [high/tall]. What was too [high/tall]? Answers: The shelf/the pot.

41. I'm sure that my map will show this building; it is very [famous/good]. What is [famous/good]? Answers: The building/the map.

42. Bob paid for Charlie's college education. He is very [generous/grateful]. Who is [generous/grateful]? Answers: Bob/Charlie.

43. Bob paid for Charlie's college education, but now Charlie acts as though it never happened. He is very [hurt/ungrateful]. Who is [hurt/ungrateful]? Answers: Bob/Charlie

44. Bob was playing cards with Adam and was way ahead. If Adam hadn't had a sudden run of good luck, he would have [won/lost]. Who would have [won/lost]? Answers: Bob/Adam.

45. Adam can't leave work here until Bob arrives to replace him. If Bob had left home for work on time, he would be [here/gone] by this time. Who would be [here/gone]? Answers: Bob/Adam

46. If the con artist has succeeded in fooling Sam, he would have [gotten/lost] a lot of money. Who would have [gotten/lost] the money? Answers: The con artist/Sam.

47. It was a summer afternoon, and the dog was sitting in the middle of the lawn. After a while, it got up and moved to a spot under the tree, because it was [hot/cooler]. What was [hot/cooler]? Answers: The dog/The spot under the tree.

48. The cat was lying by the mouse hole waiting for the mouse, but it was too [cautious/impatient]. What was too [cautious/impatient]? Answers: The mouse/the cat.

49. Anne gave birth to a daughter last month. She is a very charming [woman/baby]. Who is a very charming [woman/baby]? Answers: Anne/Anne's daughter.

50. Alice tried frantically to stop her daughter from [chatting/barking] at the party, leaving us to wonder why she was behaving so strangely. Who was behaving strangely? Answers: Alice/Alice's daughter.

51. I saw Jim yelling at some guy in a military uniform with a huge red beard. I don't know [who/why] he was, but he looked very unhappy. Who looked very unhappy? Answers: The guy in the uniform/Jim.

52. The fish ate the worm. It was [tasty/hungry]. What was [tasty/hungry]? Answers: The worm/the fish.

53. I was trying to open the lock with the key, but someone had filled the keyhole with chewing gum, and I couldn't get it [in/out]. What couldn't I get [in/out]? Answers: The key/the chewing gum.

54. The dog chased the cat, which ran up a tree. It waited at the [top/bottom]. Which waited at the [top/bottom]? Answers: The cat/the do g.

55. In the storm, the tree fell down and crashed through the roof of my house. Now, I have to get it [removed/repaired]. What has to be [removed/repaired]? Answers: The tree/the roof.

56. The customer walked into the bank and stabbed one of the tellers. He was immediately taken to the [emergency room/police station]. Who was taken to the [emergency room/police station]? Answers: The teller/the customer.

57. John was doing research in the library when he heard a man humming and whistling. He was very [annoyed/annoying]. Who was [annoyed/annoying]? Answers:John/the hummer.

58. John was jogging through the park when he saw a man juggling watermelons. He was very [impressed/impressive]. Who was [impressed/impressive]? Answers: John/the juggler.

59. Bob collapsed on the sidewalk. Soon he saw Carl coming to help. He was very [ill/concerned]. Who was [ill/concerned]? Answers: Bob/Carl.

60. Sam and Amy are passionately in love, but Amy's parents are unhappy about it, because they are [snobs/fifteen]. Who are [snobs/fifteen]? Answers: Amy's parents/Sam and Amy.

61. Mark told Pete many lies about himself, which Pete included in his book. He should have been more [truthful/skeptical]. Who should have been more [truthful/skeptical]? Answers: Mark/Pete.

62. Joe has sold his house and bought a new one a few miles away. He will be moving [out of/into] it on Thursday. Which house will he be moving [out of/into]? Answers: The old house/the new house.

63. Many people start to read Paul's books and can't put them down. They are [gripped/popular] because Paul writes so well. Who or what are [gripped/popular]? Answers: The readers/the books.

64. Mary took out her flute and played one of her favorite pieces. She has [loved/had] it since she was a child. What has Mary [loved/had] since she was a child? Answers: The piece/the flute.

65. Sam pulled up a chair to the piano, but it was broken, so he had to [stand/sing] instead. What was broken? Answers: The chair/the piano.

66. Since it was raining, I carried the newspaper [over/in] my backpack to keep it dry. What was I trying to keep dry? Answers: The backpack /the newspaper.

67. Sara borrowed the book from the library because she needs it for an article she is working on. She [reads/writes] it when she gets home from work. What does Sara [read/write] when she gets home from work? Answers: The book/the article.

68. This morning, Joey built a sand castle on the beach, and put a toy flag in the highest tower, but this afternoon [a breeze/the tide] knocked it down. What did the [breeze/tide] knock down? Answers: The flag/the sand castle.

69. Jane knocked on Susan's door, but there was no answer. She was [out/disappointed]. Who was [out/disappointed]? Answers: Susan/Jane.

70. Jane knocked on the door, and Susan answered it. She invited her to come [out/in]. Who invited whom? Answers: Jane invited Susan an/Susan invited Jane.

71. Sam took French classes from Adam, because he was [eager/known] to speak it fluently. Who was [eager/known] to speak French fluently? Answers: Sam/Adam.

72. The path to the lake was blocked, so we couldn't [reach/use] it. What couldn't we [reach/use]? Answers: The lake/the path.

73. The sun was covered by a thick cloud all morning, but luckily, by the time the picnic started, it was [gone/out]. What was [gone/out]? Answers: The cloud/the sun.

74. We went to the lake, because a shark had been seen at the ocean beach, so it was a [dangerous/safer] place to swim. Which was a [dangerous/safer] place to swim? Answers: The beach/t he lake.

75. Sam tried to paint a picture of shepherds with sheep, but they ended up looking more like [dogs/-golfers]. What looked like [dogs/golfers]? Answers:the sheep/the shepherds.

76. Mary tucked her daughter Anne into bed, so that she could [sleep/work]. Who is going to [sleep/-work]? Answers: Anne/Mary.

77. Fred and Alice had very warm down coats, but they were not [enough/prepared] for the cold in Alaska. Who or what were not [enough/prepared] for the cold? Answers: The coats/Fred and Alice.

78. Thomson visited Cooper's grave in 1765. At that date he had been [dead/travelling] for five years. Who had been [dead/travelling] for five years? Answers: Cooper/Thomson

79. Jackson was greatly influenced by Arnold, though he lived two centuries [earlier/later]. Who lived [earlier/later]? Answers: Arnold/Jackson.

80. Tom's daughter Eva is engaged to Dr. Stewart, who is his partner. The two [doctors/lovers] have known one another for ten years. Which two people have known one another for ten years? Answers: Tom and Dr. Stewart/Eva and Dr. Stewart.

81. I can't cut that tree down with that axe; it is too [thick/small]. What is too [thick/small]? Answers: The tree/the axe.

82. The foxes are getting in at night and attacking the chickens. I shall have to [guard/kill] them. What do I have to [guard/kill]? Answers: The chickens/the foxes.

83. The foxes are getting in at night and attacking the chickens. They have gotten very [bold/nervous]. What has gotten [bold/nervous]? Answers: The foxes/the chickens.

84. Fred covered his eyes with his hands, because the wind was blowing sand around. He [opened/lowered] them when the wind stopped. What did Fred [open/lower]? Answers: His eyes/his hands.

85. The actress used to be named Terpsichore, but she changed it to Tina a few years ago, because she figured it was [easier/too hard] to pronounce. Which name was [easier/too hard] to pronounce? Answers: Tina/Terpsichore.

86. Fred watched TV while George went out to buy groceries. After an hour he got [up/back]. Who got [up/back]? Answers: Fred/George.

87. Fred was supposed to run the dishwasher, but he put it off, because he wanted to watch TV. But the show turned out to be boring, so he changed his mind and turned it [on/off]. What did Fred turn [on/off]? Answers: The dishwasher/the television.

88. Fred is the only man still alive who remembers my great-grandfather. He [is/was] a remarkable man. Who [is/was] a remarkable man? Answers: Fred/my great-grandfather.

89. Fred is the only man alive who still remembers my father as an infant. When Fred first saw my father, he was twelve [years/months] old. Who was twelve [years/months] old? Answers: Fred/my father.

90. In July, Kamtchatka declared war on Yakutsk. Since Yakutsk's army was much better equipped and ten times larger, they were [victorious/defeated] within weeks. Who was [victorious/defeated]Answers: Yakutsk/Kamchatka.

91. Elizabeth moved her company from Sparta to Troy to save money on taxes; the taxes are much [higher/lower] there. Where are the taxes [higher/lower]? Answers: In Sparta/In Troy

92. Esther figures that she will save shipping costs if she builds her factory in Springfield instead of Franklin, because [most/none] of her customers live there. In which town do [most/none] of Esther's customers live? Answers: Springfield/Franklin.

93. Look! There is a [shark/minnow] swimming right below that duck! It had better get away to safety fast! What needs to get away to safety? Answer Pair A: The shark/The duck. Answer Pair B: The minnow/the duck.

94. There are too many deer in the park, so the park service brought in a small pack of wolves. The population should [increase/decrease] over the next few years. Which population will [increase/decrease]? Answers: The wolves/the deer.

95. Archaeologists have concluded that humans lived in Laputa 20,000 years ago. They hunted for [deer/evidence] on the river banks. Who hunted for [deer/evidence]? Answers: The prehistoric humans/the archaeologists.

96. The scientists are studying three species of fish that have recently been found living in the Indian Ocean. They [appeared/began] two years ago. Who or what [appeared/began] two years ago? Answers: The fish/the scientists.

97. The journalists interviewed the stars of the new movie. They were very [cooperative/persistent], so the interview lasted for a long time. Who was [cooperative/persistent]? Answers: The stars/the journalists

98. The police arrested all of the gang members. They were trying to [run/stop] the drug trade in the neighborhood. Who was trying to [run/stop] the drug trade? Answers: The gang/the police.

99. I put the cake away in the refrigerator. It has a lot of [butter/leftovers] in it. What has a lot of [butter/leftovers]? Answers: The cake/the refrigerator.

100. Sam broke both his ankles and he's walking with crutches. But a month or so from now they should be [better/unnecessary]. What should be [better/unnecessary]? Answers: The ankles/the crutches.

101. When the sponsors of the bill got to the town hall, they were surprised to find that the room was full of opponents. They were very much in the [majority/minority]. Who were in the [majority/minority]? Answers: The opponents /the sponsors.

102. Everyone really loved the oatmeal cookies; only a few people liked the chocolate chip cookies. Next time, we should make [more/fewer] of them. Which cookie should we make [more/fewer] of, next time? Answers: The oatmeal cookies/the chocolate chip.

103. We had hoped to place copies of our newsletter on all the chairs in the auditorium, but there were simply [not enough/too many] of them. There are [too many/not enough] of what? Answers: chairs/copies of the newsletter.

104. I stuck a pin through a carrot. When I pulled the pin out, it [left/had] a hole. What [left/had] a hole? Answers: The pin/the carrot.

105. I couldn't find a spoon, so I tried using a pen to stir my coffee. But that turned out to be a bad idea, because it got full of [ink/coffee]. What got full of [ink/coffee]? Answers:The coffee/the pen.

106. Steve follows Fred's example in everything. He [admires/influences] him hugely. Who [admires/influences] whom? Answers: Steve admires Fred/Fred influences Steve.

107. The table won't fit through the doorway because it is too [wide/narrow]. What is too [wide/narrow]? Answers: The table/the doorway.

108. Grace was happy to trade me her sweater for my jacket. She thinks it looks [great/dowdy] on her. What looks [great/dowdy] on Grace? Answers: The jacket/the sweater.

109. Bill thinks that calling attention to himself was rude [to/of] Bert. Who called attention to himself? Answers: Bill/Bert.

110. John [hired/hired himself out to] Bill to take care of him. Who is taking care of whom? Answers: Bill is taking care of John/John is taking care of Bill.

111. John [promised/ordered] Bill to leave, so an hour later he left. Who left? Answers: John/Bill.

112. Sam Goodman's biography of the Spartan general Xenophanes conveys a vivid sense of the difficulties he faced in his [childhood/research]. Who faced difficulties? Answers: Xenophanes/Sam.

113. Emma's mother had died long ago, and her [place/education] had been [taken/managed] by an excellent woman as governess. Whose [place/education] had been [taken/managed]? Answers: Emma's mother/Emma.

114. Jane knocked on Susan's door but she did not [answer/get an answer]. Who did not [answer/get an answer]? Answers: Susan/Jane

115. Joe paid the detective after he [received/delivered] the final report on the case. Who [received/delivered] the final report? Answers: Joe/the detective.

116. Beth didn't get angry with Sally, who had cut her off, because she stopped and [counted to ten/apologized]. Who [counted to ten/apologized]? Answers: Beth/Sally

117. Jim signaled the barman and gestured toward his [empty glass/bathroom key]. Whose [empty glass/bathroom key]? Answers: Jim/the barman.

118. Dan took the rear seat while Bill claimed the front because his "Dibs!" was [quicker/slow]. Whose "Dibs" was [quicker/slow]? Answers: Bill/Dan

119. Tom said "Check" to Ralph as he [took/moved] his bishop. Whose bishop did Tom [take/move]? Answers: Ralph's /Tom's

120. As Andrea in the crop duster passed over Susan, she could see the landing [strip/gear]. Who could see the landing [strip/gear]? Answers: Andrea/Susan

121. Tom gave Ralph a lift to school so he wouldn't have to [walk/drive alone]. Who wouldn't have to [walk/drive alone]? Answers: Ralph/Tom

122. Bill passed the half-empty plate to John because he was [full/hungry]. Who was [full/hungry]? Answers: Bill/John

123. Bill passed the gameboy to John because his turn was [over/next]. Whose turn was [over/next]? Answers: Bill/John

124. The man lifted the boy onto his [bunk bed/shoulders]. Whose [bunk bed/shoulders]? Answers: The boy's/the man's.

125. Patting/Stretching her back, the woman smiled at the girl. Whose back did the woman [pat/stretch]? Answers: The girl's/ the woman's

126. Billy cried because Toby wouldn't [share/accept] his toy. Who owned the toy? Answers: Toby/-Billy

127. Lily spoke to Donna, breaking her [concentration/silence]. Whose [concentration/silence]? Answers: Donna/ Lily

128. When Tommy dropped his ice cream, Timmy giggled, so father gave him a [stern/sympathetic] look. Who got the look from father? Answers: Timmy/Tommy

129. As Ollie carried Tommy up the long winding steps, his legs [dangled/ached]. Whose legs [dangled/ached]? Answers: Tommy/Ollie

130. The father carried the sleeping boy in his [arms/bassinet]. Whose [arms/bassinet]? Answers: The father/the boy

131. The woman held the girl against her [chest/will]. Whose [chest/will]? Answers: The woman's/the girl's

132. Pam's parents came home and found her having sex with her boyfriend, Paul. They were [embarrassed/furious] about it. Who were [embarrassed/furious]? Answers: Pam and Paul/Pam's parents.

133. Dr. Adams informed Kate that she had [cancer/retired] and presented several options for future treatment. Who had [cancer/retired]? Answers: Kate/Dr. Adams

134. Dan had to stop Bill from toying with the injured bird. He is very [compassionate/cruel]. Who is [compassionate/cruel]? Answers: Dan/Bill

135. George got free tickets to the play, but he gave them to Eric [because/even though] he was [particularly/not particularly] eager to see it. Who [was/was not] eager to see the play?
Answers:"because " & "particularly": Eric.
"because" & "not particularly": George
"even though" & "particularly": George
"even though" & "not particularly": Eric

136. Jane gave Joan candy because she [was/wasn't] hungry. Who [was/wasn't] hungry? Answers: Joan/Jane.

137. I tried to paint a picture of an orchard, with lemons in the lemon trees, but they came out looking more like [light bulbs /telephone poles]. What looked like [light bulbs/telephone poles]? Answers: The lemons/the trees.

138. James asked Robert for a favor but he [refused/was refused]. Who [refused/was refused]? Answers: Robert/James

139. Kirilov ceded the presidency to Shatov because he was [more/less] popular. Who was [more/less] popular? Answers: Shatov/Kirilov

140. Emma did not pass the ball to Janie although she [was open/saw that she was open]. Who [was open/saw that the other player was open]? Answers: Janie/Emma

141. Joe saw his brother skiing on TV last night but the fool didn't [recognize him/have a coat on] Who is the fool? Answers: Joe/Joe's brother.

142. I put the [heavy book/butterfly wing] on the table and it broke. What broke? Answer Pair A: The table/The book Answer Pair B: The butterfly wing/The table

143. Madonna fired her trainer because she [slept with/couldn't stand] her boyfriend. Who [slept with/-couldn't stand] whose boyfriend? Answer: The trainer slept with Madonna 's boyfriend/Madonna couldn't stand the train er's boyfriend.

144. Carol believed that Rebecca [suspected/regretted] that she had stolen the watch. Who is suspected of stealing the watch?/Who stole the watch? Answer: Carol/Rebecca

145. This book introduced Shakespeare to [Ovid/Goethe]; it was a major influence on his writing. Whose writing was influenced? Answer: Shakespeare/Goethe

146. This book introduced Shakespeare to [Ovid/Goethe]; it was a fine selection of his writing. A fine selection of whose writing? Answer: Ovid/ Shakespeare

147. Alice looked for her friend Jade in the crowd. Since she always [has good luck/wears a red turban], Alice spotted her quickly. Who always [has good luck/wears a red turban]Answer: Alice/Jade

148. During a game of tag, Ethan [chased/ran from] Luke because he was "it". Who was "it"? Answer: Ethan/Luke

149. At the Loebner competition the judges couldn't figure out which respondents were the chatbots because they were so [advanced/stupid]. Who were so [advanced/stupid]? Answer: the chatbots/the judges.

150. The user changed his password from "GrWQWu8JyC" to "willow-towered Canopy Huntertropic wrestles" as it was easy to [remember/forget]. What was easy to [remember/forget]? Answer: the password "GrWQWu8JyC"/ the password " willow-towered Canopy Huntertropic wrestles")

# WS	**Physical**	**Emotional**	**Interactions**	**Comparison**	**Causal**	**Multiple knowledge**
1		✓				✓
2	✓			✓		
3			✓✓			
4			✓✓			
5			✓✓			
6	✓			✓		
7		✓✓				
8	✓				✓	
9	✓✓					
10	✓✓					
11	✓✓					
12					✓	
13	✓✓					
14	✓			✓		
15				✓✓		
16	✓				✓	
17			✓✓			
18		✓✓				
19	✓✓					
20					✓✓	
21	✓✓					

22					✓	✓
23						✓✓
24	✓			✓		
25						✓✓
26			✓✓			
27					✓✓	
28				✓✓		
29	✓✓					
30	✓✓					
31	✓✓					
32						✓✓
33	✓✓					
34	✓✓					
35	✓✓					
36					✓✓	
37			✓			✓
38			✓		✓	
39	✓				✓	
40	✓✓					
41						✓✓
42			✓✓			
43			✓✓			
44				✓✓		
45				✓	✓	
46				✓	✓	
47				✓✓		
48			✓			✓
49					✓	✓
50			✓✓			
51		✓✓				
52				✓✓		
53	✓✓					

54	✓✓					
55	✓✓					
56	✓✓					
57		✓				✓
58						✓✓
59			✓			✓
60						✓✓
61			✓✓			
62	✓✓					
63					✓✓	
64						✓✓
65					✓	✓
66	✓✓					
67			✓	✓		
68	✓✓					
69			✓			✓
70	✓✓					
71			✓		✓	
72			✓		✓	
73						✓✓
74					✓✓	
75				✓✓		
76			✓	✓		
77						✓✓
78			✓			✓
79				✓		✓
80						✓✓
81	✓✓					
82			✓			✓
83		✓✓				
84	✓		✓			
85				✓✓		

86	✓		✓			
87					✓✓	
88	✓✓					
89	✓✓					
90			✓✓			
91				✓✓		
92					✓✓	
93	✓✓					
94				✓✓		
95						✓✓
96						✓✓
97			✓✓			
98			✓✓			
99	✓					✓
100					✓✓	
101				✓✓		
102				✓✓		
103				✓✓		
104	✓✓					
105						✓✓
106			✓✓			
107	✓			✓		
108			✓	✓		
109			✓✓			
110			✓✓			
111			✓✓			
112						✓✓
113						✓✓
114			✓✓			
115			✓✓			
116					✓	✓
117	✓					✓

118				✓ ✓		
119	✓		✓			
120	✓ ✓					
121	✓		✓			
122				✓ ✓		
123			✓ ✓			
124	✓ ✓					
125	✓					✓
126			✓			✓
127			✓ ✓			
128		✓ ✓				
129						✓ ✓
130	✓ ✓					
131			✓ ✓			
132		✓ ✓				
133						✓ ✓
134		✓ ✓				
135					✓ ✓	
136					✓ ✓	
137						✓ ✓
138			✓ ✓			
139				✓ ✓		
140			✓		✓	
141						✓ ✓
142	✓ ✓					
143			✓			✓
144			✓ ✓			
145						✓ ✓
146						✓ ✓
147						✓ ✓
148			✓ ✓			
149			✓	✓		

150				✓✓		

Table B.1: Annotation results

Legend

- ✓ Annotator 1
- ✓ Annotator 2