Combining Knowledge Hunting and Neural Language Models to Solve the Winograd Schema Challenge

Ashok Prakash, Arpit Sharma, Arindam Mitra, Chitta Baral
Arizona State University
Tempe, USA
{apraka23,asharm73,amitra7,chitta}@asu.edu

Abstract

Winograd Schema Challenge (WSC) is a pronoun resolution task which seems to require reasoning with commonsense knowledge. The needed knowledge is not present in the given text. Automatic extraction of the needed knowledge is a bottleneck in solving the challenge. The existing state-of-the-art approach uses the knowledge embedded in their pre-trained language model. However, the language models only embed part of the knowledge, the ones related to frequently co-existing concepts. This limits the performance of such models on the WSC problems. In this work, we build-up on the language model based methods and augment them with a commonsense knowledge hunting (using automatic extraction from text) module and an explicit reasoning module. Our end-to-end system built in such a manner improves on the accuracy of two of the available language model based approaches by 5.53% and 7.7% respectively. Overall our system achieves the state-of-the-art accuracy of 71.06% on the WSC dataset, an improvement of 7.36% over the previous best.

1 Introduction

Reasoning with commonsense knowledge is an integral component of human behavior. It is due to this capability that people know that they should dodge a stone that is thrown towards them. It has been a long standing goal of the Artificial Intelligence community to simulate such commonsense reasoning abilities in machines. Over the years, many advances have been made and various challenges have been proposed to test their abilities (Clark et al., 2018; Mihaylov et al., 2018; Mishra et al., 2018). The Winograd Schema Challenge (WSC) (Levesque et al., 2011) is one such natural language understanding challenge. It is made up of pronoun resolution problems of a particular kind. The main part of each WSC problem is a set of sentences containing a pronoun. In addition, two definite noun phrases, called "answer choices" are also given. The answer choices are part of the input set of sentences. The goal is to determine which answer provides the most natural resolution for the pronoun. Below is an example problem from the WSC.

Sentences (S1): The fish ate the worm. It was tasty.
Pronoun to resolve: It
Answer Choices: a) fish b) worm

A WSC problem also specifies a "special word" that occurs in the sentences, and an "alternate word." Replacing the former by the latter changes the resolution of the pronoun. In the example above, the special word is *tasty* and the alternate word is *hungry*.

The resolution of the pronoun is difficult because the commonsense knowledge that is required to perform the resolution is not explicitly present in the input text. The above example requires the commonsense knowledge that *'something that is eaten may be tasty'*. There have been attempts (Sharma et al., 2015b; Emami et al., 2018a) to extract such knowledge from text repositories. Those approaches find the sentences which are similar to the sentences in a WSC problem but without the co-reference ambiguity. For example a sentence (which contains knowledge without ambiguity) corresponding to the above WSC problem is *'John ate a tasty apple'*. Such an approach to extract and use sentences which contain evidence for co-reference resolution is termed as Knowledge Hunting (Sharma et al., 2015b; Emami et al., 2018b). There are two main modules in the knowledge hunting approach, namely a knowledge extraction module and a reasoning module. To be able to use the extracted knowledge, the reasoning module puts several restrictions on the structure of the knowledge. If the knowledge extraction module could not find any knowledge pertaining to those restrictions, the extracted knowledge would probably be of no use.

Sometimes the needed knowledge are embedded in the pre-trained language models. Let us consider the WSC example mentioned below.

S2: The painting in Mark's living room shows an oak tree. It is to the right of a house.
Pronoun to resolve: It
Answer Choices: a) painting b) tree

Here, the knowledge that *'a tree is to the right of a house'* is more likely than *'a painting is to the right of a house'* is needed. With recent developments in neural network architectures for language modeling, it is evident that they are able to capture such knowledge by predicting that *'a tree is to the right of a house'* is a more probable phrase than *'a painting is to the right of a house'*. This is because language models are trained on huge amounts of text and they are able to learn the frequently co-occurring concepts from that text. Although the knowledge from language models is helpful in many examples, it is not suitable for several others. For example, the language models in (Trinh and Le, 2018) predict that *'fish is tasty'* is a more probable than *'worm is tasty'*. This is because the words *'fish'* and *'tasty'* occur in the same context more often than the words *'worm'* and *'tasty'*.

So, considering the benefits and limitations of the above mentioned approaches, in this work, we combine the knowledge hunting and neural language models to solve the Winograd Schema Challenge (WSC). The main contribution of this work is to tackle the WSC by:

- developing and utilizing an automated knowledge hunting approach to extract the needed knowledge and reason with it without relying on a strict formal representation,
- utilizing the knowledge that is embedded in the language models, and
- combining the knowledge extracted from knowledge hunting and the knowledge in language models.

As a result, our approach improves on the existing state-of-the-art accuracy by 7.36% and solves 71.06% of the WSC problems correctly.

2 Related Work

The Winograd Schema Challenge is a co-reference resolution problem. The problem of co-reference resolution has received large amount of attention in the field of Natural Language Processing (Raghunathan et al., 2010; Carbonell and Brown, 1988; Ng, 2017). However the requirement to use commonsense knowledge makes the Winograd Schema Challenge hard and the other approaches that are trained on their respective corpora do not perform well in the Winograd Schema problems.

The Winograd Schema Challenge was first proposed in 2011 and since then various works have been proposed to address it. These approaches can be broadly categorized into two types:

1. The approaches which use explicit commonsense knowledge and reasoning with the knowledge. Such approaches can further be divided into two types.

   (a) The approaches which provide a reasoning theory (Bailey et al., 2015; Schüller, 2014; Sharma et al., 2015b) with respect to a few specific types of commonsense knowledge and takes question specific knowledge while solving a Winograd Schema problem. One of the major shortcomings of such approaches is that they work only for the specific knowledge types and hence their coverage is restricted. Another shortcoming of such approaches is that they rely on strict formal representations of natural language text. The automatic development of such representations boils down to the well known complex problem of translating a natural language text into its formal meaning representation. Among these works, only the work of (Sharma et al., 2015b) accepts natural language knowledge sentences which it automatically converts into their required graphical representation (Sharma et al., 2015a). The remaining two (Bailey et al., 2015; Schüller, 2014) requires the knowledge to be provided in a logical form.

   (b) These approaches (Isaak and Michael, 2016) also answer a Winograd Schema problem with formal reasoning but use an existing knowledge base of facts and first-order rules to do that.

2. These approaches (Liu et al., 2017; Trinh and Le, 2018) utilize the recent advancement in the field of neural networks, particularly the benefits of word embedding and neural language model. The work of (Liu et al., 2017) uses ConceptNet and raw texts to train word embeddings which they later use to solve a Winograd Schema problem by a simple inference algorithm. The work of (Trinh and Le, 2018) on the other hand uses majority voting from several language models to resolve the co-reference. In layman terms, the system in (Trinh and Le, 2018) replaces the pronoun with the two answer choices to obtain two different sentences and then use the language models to find out which of the two replacement is more probable.

3 Our Method

In this section we first explain how our knowledge hunting approach and the neural language models are used to generate respective intermediate results. Then we explain the details of a Probabilistic Soft Logic (PSL) module which combines the intermediate results and predicts the confidence for each of the answer choices in a WSC example.

3.1 Knowledge Hunting Approach

There are two main modules in the Knowledge Hunting approach. The first module extracts a set of sentences corresponding to a WSC problem such that the extracted sentences may contain the needed commonsense knowledge. We call such a set of sentences, a *knowledge text*. The second module uses a *knowledge text* and generates a correspondence between the answer choices and the pronoun in a WSC text, and the entities in a *knowledge text*. We call such a correspondence as *entity alignment*. Such an *entity alignment* is an intermediate result from the knowledge hunting module. In the following we provide the details of *knowledge text* extraction and *entity alignment* modules.

3.1.1 Knowledge Extraction

The goal of the knowledge extraction module is to automatically extract a set of knowledge texts for a given WSC problem. Ideally, a *knowledge text* should be able to justify the answer of the associated WSC problem. In this vein, we aim to extract the texts that depict a scenario that is similar to that of the associated WSC problem. We roughly characterize a WSC scenario in terms of the events (verb phrases) and the properties of the entities that are associated with the scenario. The characterization of a scenario optionally includes the discourse connectives between the events and properties of the scenario. For example, in the WSC sentence *"The city councilmen refused the demonstrators a permit because they feared violence ."*, the scenario is mainly characterized by the verb phrases *"refused"* and *"feared"*, and the discourse connective *"because"*.

In this work, we use this abstract notion of a scenario to extract *knowledge texts* which depict similar scenarios. The following are the steps in the extraction module.

1. First, the module identifies the verb phrases, properties and discourse connectives in a given WSC *scenario*. For example the one-word verb phrases *"refused"* and *"feared"*, and the discourse connective *"because"* in the example mentioned above.

2. Secondly, the module automatically generates a set of search queries by using the keywords extracted in the previous step. The first query in the set is an ordered combination (as per the WSC sentence) of the keywords extracted in the previous step. For example the query *"* refused * because * feared * "* is the first query for the problem mentioned above. Afterwards the following set of modifications are performed with respect to the first query and the results are added to the set of queries.

   - The verb phrases are converted to their base form. For example, *" * refuse * because * fear * "*.
   - The discourse connectives are omitted. For example, *"* refuse * fear * "*.
   - The verbs in verb phrases and the adjectives are replaced with their synonyms from the WordNet KB (Miller, 1995). The top five synonyms from the top synset of the same part of speech are considered. An example query generated after this step is *"* decline * because * fear * "*.

3. Thirdly, the module uses the generated queries to search and extract text snippets, of length up to 30 words, from a search engine. The top 10 results (urls) from the search engine are retrieved for each query and text

snippets from those results are scraped. Out of the extracted texts, the 10 text snippets which are most similar to the WSC text are filtered and passed to the alignment module. We used a natural language inference model (Parikh et al., 2016) to find the most similar sentences. Since we also do not want to extract the snippets which contain the corresponding WSC sentences (because of ambiguity), this module removes the results with WSC sentences in them. We filtered out the *knowledge texts* which contained 80% or more words from the sentences in any of the WSC problems.

An example *knowledge text* extracted by using the query *“ * refused * because * feared * ”* via the steps mentioned above is, *“He also refused to give his full name because he feared for his safety.”*

3.1.2 Entity Alignment

A total of up to 10 *knowledge texts* are extracted with respect to each WSC problem. Each of them is processed individually along with the WSC problem to produce a corresponding intermediate result from the knowledge hunting module.

Let $W = \langle S, A_1, A_2, P, K \rangle$ be a modified WSC problem such that S be a set of WSC sentences, A_1 and A_2 be the answer choices one and two respectively, P be the pronoun to be resolved, and K be a *knowledge text*. The existing solvers (Sharma et al., 2015b) that use explicit knowledge to solve a WSC problem of the form W first convert K and S into a logical form and then use a set of axioms to compute the answer. However, it is a daunting task to convert free form text into a logical representation. Thus these methods often produce low recall. In this work, we take a detour from this approach and aim to build an “alignment” function. Informally, the task of the alignment function is to align the answer choices (A_1 and A_2) and the pronoun to be resolved (P) in S with the corresponding entities (noun/pronoun phrases) in K. These alignments are the intermediate results of the knowledge hunting module.

By the choice of knowledge extraction approach, the knowledge texts are similar to the WSC sentences in terms of events, i.e., they contain similar verb phrases, properties and discourse connectives. So, in an ideal situation we will have entities in K corresponding to each one of the concerned entities (A_1, A_2 and P) in W respectively. The goal of the alignment algorithm is to find that mapping. The mapping result is generated in the form of a *aligned_with* predicate of arity three. The first argument represents an entity (an answer choice or the pronoun) from S, the second argument represents an entity from K and the third argument is an identifier of the knowledge text used. We define an entity (noun phrase) E_j from a *knowledge text* K to be *aligned_with* to an entity A_j from a WSC text S if the following holds:

1. There exists a verb v in S and v' in K such that either $v = v'$ or v is a synonym of v'.
2. The “semantic role” of A_j with respect to v is same as the “semantic role” of E_j with respect to v'.

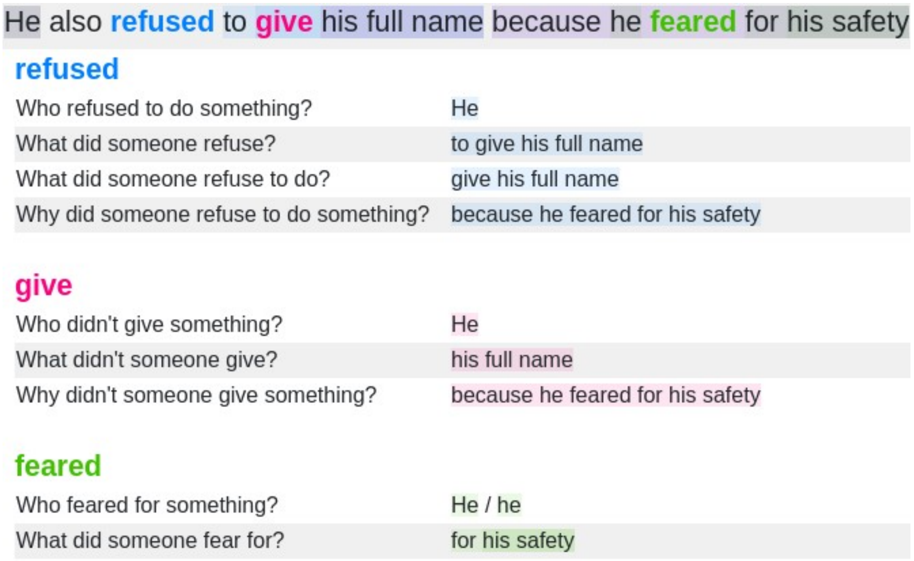
We use the semantic role labelling function, called QASRL (He et al., 2015) to compute the semantic roles of each entity. QASRL represents the semantic roles of an entity, in terms of question-answer pairs. Figure 1 shows the QASRL representation of the *knowledge text* “*He* also refused to give his full name because *he* feared for his safety.” It involves three verbs “refused”, “feared” and “give”. The questions represent the roles of the participating entities.

An example alignment generated for the WSC sentence,
S = *“The city councilmen refused the demonstrators a permit because they feared violence.”*
and the *knowledge text*,
K = *“He also refused to give his full name because he feared for his safety.”*
is,

aligned_with(city councilmen,He,K)
aligned_with(they,he,K)

There are three relevant entities in an input WSC problem, i.e., A_1, A_2 and P. Based on the existence of the entities corresponding to the entities in the WSC problem there are 2^8 possible cases. For example, the case $\{$*True True True*$\}$, abbreviated as $\{$*TTT*$\}$, represents that each of the entities A_1, A_2 and P are aligned with corresponding entities in a *knowledge text*.

The intuition behind the alignment approach is to find a common entity in a *knowledge text* such that it aligns with one of the answer choices (say A_i) and also with the pronoun to be resolved (P).



He also refused to give his full name because he feared for his safety

refused

Who refused to do something?	He
What did someone refuse?	to give his full name
What did someone refuse to do?	give his full name
Why did someone refuse to do something?	because he feared for his safety

give

Who didn't give something?	He
What didn't someone give?	his full name
Why didn't someone give something?	because he feared for his safety

feared

Who feared for something?	He / he
What did someone fear for?	for his safety

Figure 1: QASRL output for the sentence *"He also refused to give his full name because he feared for his safety."*

Case	Details	Example
TTT	Each entity (among A_1, A_2 and P) in the WSC sentences W have corresponding entities in the corresponding *knowledge text* K	**WSC Sentence:** ***Jim*** *comforted* ***Kevin*** *because* ***he*** *was so upset .* **Knowledge Text (K):** *She says* ***I*** *comforted* ***her***, *because* ***she*** *was so upset* **Alignments:** *aligns_with(Jim,I,K), aligns_with(Kevin,her,K), aligns_with(he,she,K)*
TFT	Only the entity representing the answer choice one (A_1) and the pronoun to be resolved (P) have corresponding entities in the *knowledge text* K	**WSC Sentence:** *The* ***trophy*** *does not fit into the brown* ***suitcase*** *because* ***it*** *is too large .* **Knowledge Text (K):** *installed CPU and* ***fan*** *would not fit in because the* ***fan*** *was too large* **Alignments:** *aligns_with(trophy,fan,K), aligns_with(it,fan,K)*
FTT	Only the entity representing the answer choice 2 (A_2) and the pronoun to be resolved (P) have corresponding entities in the *knowledge text* K	**WSC Sentence:** ***James*** *asked* ***Robert*** *for a favor but* ***he*** *refused .* **Knowledge Text (K):** *He asked the* ***LORD*** *what he should do, but the* ***LORD*** *refused to answer him, either by dreams or by sacred lots or by the prophets.* **Alignments:** *aligns_with(Robert,LORD,K) and aligns_with(he,LORD,K)*

Table 1: Alignment Cases in the Knowledge Hunting Approach. A_1 and A_2 are answer choices one and two, P is pronoun to resolve, E_{k1}, E_{k2} and E_{k3} are entities in a *knowledge text* (K)

Then we can say that both A_i and P refer to same entity and hence they refer to each other. An important aspect of such a scenario is the existence of the entities in a *knowledge text* which align with at least one of the answer choices and the pronoun to be resolved. In other words the cases $\{TTT\}$, $\{TFT\}$ and $\{FTT\}$. So we consider the alignments generated only with respect to these three cases as an output of the alignment module. The three cases and their details are shown in the Table 1 along with examples from the dataset.

3.2 Using the Knowledge from Language Models

In recent years, deep neural networks have achieved great success in the field of natural language processing (Liu et al., 2019; Chen et al., 2018). With the recent advancements in the neural network architectures and availability of powerful machine it is possible to train unsupervised language models and use them in various tasks (Devlin et al., 2018; Trinh and Le, 2018). Such language models are able to capture the knowledge which is helpful in solving many WSC problems. Let us consider the WSC problem shown below.

> **S3:** I put the heavy book on the table and it broke.
> **Pronoun to resolve:** it
> **Answer Choices:** a) table b) book

A knowledge that, "*table broke* is more likely than *book broke*" is sufficient to solve the above WSC problem. Such a knowledge is easily learned by the language models because they are trained on huge amounts of text snippets which are transcribed by people. Furthermore, these models are good at learning the frequently occurring patterns from data.

In this work, we aim to utilize such knowledge that is embedded in the neural language models. We replace the pronoun to be resolved in the WSC text with the two answer choices, one at a time, generating two possible texts. For example the two texts generated in the above WSC example are, S3(a) = *I put the heavy book on the table and table broke.*, S3(b) = *I put the heavy book on the table and book broke.* Then a pre-trained language model is used to predict the probability of each of the generated texts. Let P_a be the probability of S3(a) and P_b be the probability of S3(b). To be able to use the result of language models in Probabilistic Soft Logic (PSL) (Kimmig et al., 2012), the output of this step contains ***coref(**P**,**A_1**):**$PROB1$* and ***coref(**P**,**A_2**):**$PROB2$*, where P is the pronoun to be resolved, A_1 and A_2 are answer choices one and two respectively, and $PROB1$ and $PROB2$ are the probabilities of the texts generated by replacing P with A_1 and A_2 in the WSC text respectively, i.e., P_a and P_b in the example above.

3.3 Combining Knowledge Hunting and Language Models

In this step, the alignment results generated from the knowledge hunting module and the coreference probabilities generated from the language models are combined in a Probabilistic Soft Logic (PSL) (Kimmig et al., 2012) framework to infer the confidence for each of the answer choices in a WSC problem.

PSL is a probabilistic logic framework designed to have efficient inference. A key distinguishing feature of PSL is that ground atoms have soft, continuous truth values in the interval [0, 1] rather than binary truth values as used in Markov Logic Networks and most other kinds of probabilistic logic. Given a set of weighted logical formulas, PSL builds a graphical model defining a probability distribution over the continuous space of values of the random variables in the model. A PSL model is defined using a set of weighted if-then rules in first-order logic, as in the following example:

$$0.7 : \forall x, y, z. spouse(x, y) \land isChildOf(z, x) \rightarrow isChildOf(z, y) \tag{1}$$

Here, x, y and z represent variables. The above rule states that a person's child is also a child of his/her spouse. The weight (0.7) associated with the rule encodes the strength of the rule.

Each grounded atom, in a rule of a PSL model has a soft truth value in the interval [0, 1], which is denoted by I(a). Following formulas are used to compute soft truth values for the conjunctions ($\land$), disjunctions ($\lor$) and negations ($\neg$) in the logical formulas.

$$\begin{aligned} I(l1 \land l2) &= max\{0, I(l1) + I(l2) - 1\} \\ I(l1 \lor l2) &= min\{I(l1) + I(l2), 1\} \\ I(\neg l1) &= 1 - I(l1) \end{aligned} \tag{2}$$

Then, a given rule r $\equiv$ $rbody \rightarrow rhead$, it is said to be satisfied (i.e. I(r) = 1) iff I($rbody$) $\leq$ I($rhead$). Otherwise, PSL defines a distance to satisfaction d(r) which captures how far a rule r is from being satisfied: d(r) = max{0, I($rbody$) - I($rhead$)}. For example, assume we have the set of evidence: I($spouse(B, A)$) = 1, I($isChildOf(P, B)$) = 0.9, I($isChildOf(P, A)$) = 0.7, and that r is the resulting ground instance of rule (1). Then I($spouse(B, A)$ $\land$ $isChildOf(P, B)$)=max{0,1+0.9-1}=0.9, and d(r)=max{0,0.9-0.6}=0.3

PSL is primarily designed to support Most Probable Explanation (MPE) inference. MPE inference is the task of finding the overall interpretation (combination of grounded atoms) with the maximum probability given a set of evidence. Intuitively, the interpretation with the highest probability is the interpretation with the lowest distance to satisfaction. In other words, it is the interpretation that tries to satisfy all rules as much as possible.

We used the PSL framework to combine the results from the other modules in our approach and generate the confidence scores for each of the answer choices. The confidence scores are generated for the predicate *coref(p,*a_i*)* where p is the variable representing a pronoun to be resolved in a WSC problem and a_i is a variable representing an answer choice in the WSC problem.

To be able to use the alignment information from the knowledge hunting approach, following PSL rule was written. It is used to generate the $coref$ predicate and its truth value for the answer choices.

$$
\begin{aligned}
w : \{\forall a, e1, e2, k, p.&\\
aligned\_with(a, e1, k)\wedge&\\
aligned\_with(p, e2, k)\wedge&\\
similar(e1, e2)\wedge&\\
&\rightarrow coref(p, a)\}
\end{aligned}
\tag{3}
$$

Here w is the weight of the rule, a, p, $e1$, $e2$ and k are variables such that a is an answer choice in a WSC problem, p is the pronoun to be resolved in a WSC problem, and $e1$ and $e2$ are entities in a knowledge text k. The groundings of the $aligned\_with$ predicate are generated from the knowledge hunting module and the groundings of the $similar$ predicate encode the similar entities in k. The truth value of a grounding of $similar$ predicate is used to represent how similar the two entities, i.e., $e1$ and $e2$, are to each other. Although any kind of semantic similarity calculation algorithm may be used for producing the similar predicate, we used BERT (Devlin et al., 2018) to calculate the similarity between two entities. In case the values of $e1$ and $e2$ are same (say E) the truth value of the grounded atom $similar(E, E)$ becomes 1.

Intuitively, the above rule means that if an answer choice and the pronoun to be resolved in a WSC problem align with similar entities in a knowledge text corresponding to the WSC problem then the pronoun refers to the answer choice.

The above rule applies to all the three cases mentioned in the Table 1.

The neural language models approach produces two groundings of the atom defined by the binary predicate $coref$ as its result (see section 3.2). The two groundings refer to the co-reference between the pronoun to be resolved and the two answer choices respectively. The groundings are accompanied with their probabilities which we used as their truth values. These grounded $coref$ atoms are directly entered as input to the PSL framework along with the output from knowledge hunting approach to infer the truth values for the $coref$ atom with respect to each of the answer choices. Finally, the answer choice with higher truth value is considered as the correct co-referent of the pronoun to be resolved and hence the final answer.

4 Experiments

4.1 Dataset

The Winograd Schema Challenge corpus[1] consists of pronoun resolution problems where a set of sentences is given along with a pronoun in the sentences and two possible answer choices such that only one choice is correct. There are 285 problems in the WSC dataset. From this point onward, we will call this dataset as WSC_{285}. The generation of the original WSC dataset itself is an ongoing work. Hence the dataset keeps getting updated. This is why the works earlier than ours, used a smaller dataset containing 273 problems. All the problems in it are also present in WSC_{285}. From this point onward, we will call this subset of WSC_{285} as WSC_{273}. For a fair comparison between our work and others', we performed our experiments with respect to both WSC_{285} and WSC_{273}. The core to reproduce the results of this paper is available at https://github.com/Ashprakash/CKLM.

4.2 Experimental Setup and Results

First, we compared the results of our system with the previous works in terms of the number of correct predictions. The language models based component of our approach relies on pre-trained language models. Here we compared two different language models. First we used the ensemble of 14 pre-trained language models which are used in (Trinh and Le, 2018). Secondly, we used BERT (Devlin et al., 2018) pre-trained model. Based on the language model used, in the following experiments we use OUR_METHOD$_{T2018}$ to represent our approach which uses models from (Trinh and Le, 2018) and OUR_METHOD$_{BERT}$ to represent our approach which uses the BERT language model. We compared our method with five other methods (two language models based and three others). The comparison results are as shown in the Table 2. The first two, (Sharma et al., 2015b) and (Liu et al., 2017) hereafter called S2015 and L2017 respectively, address a subset of WSC problems (71 problems). Both of them are able to exploit only causal knowledge. This explains their low coverage over the entire corpus. We overcome this issue by using any form of knowledge text making predictions for each of

[1] Available at https://cs.nyu.edu/faculty/davise/papers/WinogradSchemas/WSCollection.xml

the problems in the dataset. More recently, two approaches on solving the WSC_{273} dataset have been proposed. The first work (Emami et al., 2018a) (hereafter called E2018) extract knowledge in form of sentences to find evidences to support each of the possible answer choices. A comparison between their results and our is present in the Table 2. Another work (Trinh and Le, 2018) (hereafter called T2018) uses a neural network architecture to learn language models from huge data sources to predict the probability of choosing one answer over the other is also compared as shown in the Table 2.

We performed a second set of experiments to further investigate the robustness of our method as compared to the state-of-the-art system (T2018). Each problem in the WSC has a sister problem in the WSC such that the texts in the two problems differ only by a word or two but the same pronoun refers to different entities. The two answer choices for both the problems in the pair are also same. For example, consider the following pair of problems.

S4: The firemen arrived **after** the police because they were coming from so far away.
Pronoun to resolve: they
Answer Choices: a) firemen b) police

S5: The firemen arrived **before** the police because they were coming from so far away .
Pronoun to resolve: they
Answer Choices: a) firemen b) police

In the above problems, only changing one word (*before*/*after*) in the sentence changes the answer to the problem. Due to this property of the dataset, a system can achieve an accuracy of 50% by just answering choice 1 as the correct answer for every problem. To make sure that this is not the case in our system, we performed the following two experiments.

1. **Experiment to Evaluate Pairwise Accuracy:** In this experiment we evaluate our method and the other methods to find out how many of the problem pairs were correctly solved. The table 3 shows the results of the experiment. It can be seen from the results that our best performing method(OUR_METHOD$_{BERT}$ on WSC$_{273}$) solves 57 pairs correctly, which is significantly more than its baseline 'BERT Only' method. Similar pattern for the other methods can be seen in the Table 3.

2. **Experiment to Evaluate System Bias:** In this experiment we evaluate our method and the others to find out if the methods are biased to chose the answer choice which is closer to the pronoun in a WSC sentence. We found that usually the answer choice 2 in the problem is closer to the pronoun to be resolved. Hence the experiments were performed to figure out how many times a method answers choice 2 as the final answer. The results of the experiments are as shown in the Table 3. As seen from the results, both, the language model based methods and our methods are not particularly biased towards one of the answer choices.

4.3 Remarks

Our best performing setting (OUR_METHOD$_{BERT}$ on WSC$_{273}$) correctly answers 26 problems which are incorrectly answered by the baseline language model (BERT Only on WSC$_{273}$). We found that the main reason for such a behavior is the addition of the suitable knowledge from the knowledge hunting module. It helps in generating the support for the correct answer to the extent that it overturns the decision of the language model. For example, we observed that for the WSC sentence *'The woman held the girl against her will'* the BERT language model predicted that *'her'* refers to *'The woman'* with the probability score of 0.513, which is incorrect, and to *'the girl'* with the probability score of 0.486. But the knowledge hunting approach alone within the PSL framework predicted the answer to be *'the girl'* with the probability score of 0.966, which is correct, and the answer *'the woman'* with the probability score of 0.034. Overall the PSL inference engine combined scores from both the approaches and corrected the decision made by the language model by predicting *'the girl'* as the correct answer with the probability score of 0.967.

On the other hand five problems were found to be incorrectly answered by our approach which were correctly answered by the language model. In all such cases the probabilities corresponding to the answer choices were found to be very close to each other and inclining towards the incor-

	#correct	**% Correct**
S2015	49	18.0
L2017	43	15.0
E2018	119	44.0
T2018 (WSC_{273})	174	63.70
T2018 (WSC_{285})	180	63.15
BERT Only (WSC_{273})	173	63.36
BERT Only (WSC_{285})	179	62.80
OUR_METHOD$_{T2018}$ (WSC_{273})	**189**	**69.23**
OUR_METHOD$_{T2018}$ (WSC_{285})	**195**	**68.42**
OUR_METHOD$_{BERT}$ (WSC_{273})	**194**	**71.06**
OUR_METHOD$_{BERT}$ (WSC_{285})	**200**	**70.17**

Table 2: Evaluation Results

	Correct Pairs	**Incorrect Pairs**	**#Times Choice2 is Chosen**
T2018 (WSC_{273})	42	89	142
T2018 (WSC_{285})	44	97	146
BERT Only (WSC_{273})	36	94	129
BERT Only (WSC_{285})	37	101	131
OUR_METHOD$_{T2018}$ (WSC_{273})	60	71	143
OUR_METHOD$_{T2018}$ (WSC_{285})	61	80	148
OUR_METHOD$_{BERT}$ (WSC_{273})	57	74	130
OUR_METHOD$_{BERT}$ (WSC_{285})	58	83	134

Table 3: Additional Experiments

rect answer. The difference between language model probabilities generally being very small, the combined approach answered incorrectly in such cases. The main reason for such a behavior is the availability of unsuitable *knowledge text*. For example the *knowledge text* for the WSC sentence *'The man lifted the boy onto his shoulders .'* was *'If she scores I'll feel really bad!' New documentary lifts the lid on life for female stars who are partners but line up for rival clubs'*. A similar pattern was found in the other settings as well.

5 Conclusion

Automatic extraction of the needed commonsense knowledge is a major obstacle in solving the Winograd Schema Challenge. We observed that sometimes the needed knowledge can be retrieved from the pre-trained neural language models. At other times a more involved knowledge about actions and properties is needed. So, in this work we utilized the knowledge embedded in the pre-trained language models and developed a technique to automatically extract the more involved commonsense knowledge from text repositories. Then we defined an approach to combine the two kinds of knowledge in a probabilistic soft logic based framework to solve the Winograd Schema Challenge (WSC). The experimental results show that the combined approach possesses the benefits of both the approaches and achieves the state-of-the-art accuracy on the WSC.

This work presents an approach to combine the ideas of knowledge hunting and language modeling to perform commonsense reasoning. It is a general approach may be applied to other commonsense reasoning tasks which require the both the knowledge embedded in the pre-trained language models and more involved knowledge about actions and properties.

Acknowledgement

Support from DARPA and NSF grant 1816039 is acknowledged.

References

Dan Bailey, Amelia Harrison, Yuliya Lierler, Vladimir Lifschitz, and Julian Michael. 2015. The winograd schema challenge and reasoning about correlation. In *In Working Notes of the Symposium on Logical Formalizations of Commonsense Reasoning*.

Jaime G Carbonell and Ralf D Brown. 1988. Anaphora resolution: a multi-strategy approach. In *Proceedings of the 12th conference on Computational linguistics-Volume 1*, pages 96–101. Association for Computational Linguistics.

Yongrui Chen, Huiying Li, and Zejian Xu. 2018. Convolutional neural network-based question answering over knowledge base with type constraint. In *China Conference on Knowledge Graph and Semantic Computing*, pages 28–39. Springer.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Ali Emami, Noelia De La Cruz, Adam Trischler, Kaheer Suleman, and Jackie Chi Kit Cheung. 2018a. A knowledge hunting framework for common sense reasoning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1949–1958.

Ali Emami, Adam Trischler, Kaheer Suleman, and Jackie Chi Kit Cheung. 2018b. A generalized knowledge hunting framework for the winograd schema challenge. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 25–31.

Luheng He, Mike Lewis, and Luke Zettlemoyer. 2015. Question-answer driven semantic role labeling: Using natural language to annotate natural language. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 643–653.

Nicos Isaak and Loizos Michael. 2016. Tackling the winograd schema challenge through machine logical inferences. In *STAIRS*, volume 284, pages 75–86.

Angelika Kimmig, Stephen H Bach, Matthias Broecheler, Bert Huang, and Lise Getoor. 2012. A short introduction to probabilistic soft logic. In *NIPS Workshop on probabilistic programming: Foundations and applications*, volume 1, page 3.

Hector J Levesque, Ernest Davis, and Leora Morgenstern. 2011. The winograd schema challenge. In *AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning*, volume 46, page 47.

Quan Liu, Hui Jiang, Andrew Evdokimov, Zhen-Hua Ling, Xiaodan Zhu, Si Wei, and Yu Hu. 2017. Cause-effect knowledge acquisition and neural association model for solving a set of winograd schema problems. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2344–2350.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. *arXiv preprint arXiv:1901.11504*.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. *arXiv preprint arXiv:1809.02789*.

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

Bhavana Dalvi Mishra, Lifu Huang, Niket Tandon, Wen-tau Yih, and Peter Clark. 2018. Tracking state changes in procedural text: A challenge dataset and models for process paragraph comprehension. *arXiv preprint arXiv:1805.06975*.

Vincent Ng. 2017. Machine learning for entity coreference resolution: A retrospective look at two decades of research. In *AAAI*, pages 4877–4884.

Ankur P Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. 2016. A decomposable attention model for natural language inference. *arXiv preprint arXiv:1606.01933*.

Karthik Raghunathan, Heeyoung Lee, Sudarshan Rangarajan, Nathanael Chambers, Mihai Surdeanu, Dan Jurafsky, and Christopher Manning. 2010. A multi-pass sieve for coreference resolution. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 492–501. Association for Computational Linguistics.

Peter Schüller. 2014. Tackling winograd schemas by formalizing relevance theory in knowledge graphs. In *Fourteenth International Conference on the Principles of Knowledge Representation and Reasoning*.

Arpit Sharma, Nguyen Vo, Somak Aditya, and Chitta Baral. 2015a. Identifying various kinds of event mentions in k-parser output. In *Proceedings of the The 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 82–88.

Arpit Sharma, Nguyen Ha Vo, Somak Aditya, and Chitta Baral. 2015b. Towards addressing the winograd schema challenge-building and using a semantic parser and a knowledge hunting module. In *IJCAI*, pages 1319–1325.

Trieu H Trinh and Quoc V Le. 2018. A simple method for commonsense reasoning. *arXiv preprint arXiv:1806.02847*.