Journal of Computer Science

Original Research Paper

The Stanford Dependency Relations for Commonsense Knowledge Representation of Winograd Schema Challenge (WSC)

[1]Nesreen Alsharman, [2]Raja Masadeh, [3]Ibrahim Ali Jawarneh and [2]Ahmad Al-Rababa'a

[1]Department of Computer Science, German Jordan University, Amman, Jordan
[2]Department of Computer Science, the World Islamic Sciences and Education University, Amman, Jordan
[3]Department of Mathematics, Al-Hussein Bin Talal University, Jordan

Article history
Received: 13-02-2024
Revised: 01-05-2024
Accepted: 14-05-2024

Corresponding Author:
Ibrahim Ali Jawarneh
Department of Mathematics,
Al-Hussein Bin Talal University, Jordan
Email: ibrahim.a.jawarneh@ahu.edu.jo

Abstract: An alternative to the Turing Test that could offer a more accurate measurement of artificial intelligence is the Winograd Schema Challenge (WSC). It presents a number of coreference resolution issues that cannot be resolved without the use of human behavior reasoning. A certain type of Commonsense Knowledge (CSK) is necessary for Winograd schema. In order to handle the representation of Winograd appropriately, this research used a Deep-learning Stanford dependency parser as a natural language processing tool created by the Stanford NLP Group. The purpose of this tool is to use dependency grammar to represent sentences based on their grammatical analysis which helps understand the connections between words in a sentence such as which words rely on other words for meaning or grammar which is the task of dependency parsing. In addition, Extracting these dependency relations reflects commonsense knowledge representation for WSC. Then, we integrate common sense knowledge with the Syntactic ontology graphical representation by substituting synonyms for the main events in each sentence. To assess the entire system, we employed Precision and Recall as natural language performance evaluation metrics. Precision and recall measures for Root and advc1 dependency types are 0.94 and 0.92 respectively. Precision and recall measures for the nsubj dependency type are 0.96 and 0.94 respectively. Precision and recall measures for dobj, idobj, and pobj dependency types are 0.92 and 0.83.

Keywords: Commonsense Knowledge (CSK), Winograd Schema Challenge, Referential Ambiguity, Stanford Dependency Relations

Introduction

A particular kind of pronoun disambiguation problem called the Winograd Schema (WS) is frequently used as a standard for assessing how well Artificial Intelligence (AI) and Machine Learning (ML) systems understand natural language Jurafsky and Martin (2023); Elazar *et al*. (2021); Takahashi *et al*. (2023); Hong *et al*. (2022). In order to answer a series of multiple-choice questions about ambiguous pronouns, candidates must comprehend contextual information. Inspired by Terry Winograd's work in artificial intelligence and language interpretation, Hector Levesque and his colleagues created these schemas Bennett (2022).

In general, Natural Language Processing (NLP) with AI solutions that simulate the coreference resolution intelligent behavior consists of three main components. First is semantically and syntactically parsing the input received from the form text (Constituency Parser, Dependency Parsing, Semantic Role Labeling). The second is extracting information that reflects all the Commonsense Reasoning or commonsense knowledge about the input text. The last one involves coming to a decision that demonstrates intelligent conduct in humans when resolving co-references. Computer science researchers are working to imitate this form of intelligent computer behavior in the field of natural language processing, which is a sub-field of artificial intelligence.

Example of Winograd Schema Challenge

Several AI competitions have been proposed in recent years to help evaluate machines' cognitive abilities

Levesque *et al.* (2012); Weston *et al.* (2015). WSC was introduced by Levesque *et al.* (2012) as an alternative to the Turing Test, which can provide a more accurate machine intelligence test. An annual competition based on this challenge has been declared by Nuance Communications, Inc. Winograd Schema's main aspect is a sentence containing a pronoun, for example, the city councilmen refused the demonstrators a permit because they feared violence. The test includes conflict of one form with the coreference resolution. In addition, there are two important noun words, named "answers," given; the responses are the demonstrators and the city council members in the above example Levesque *et al.* (2012). The objective is to identify the response that most naturally resolves the pronoun. First, the obvious response to the prior query: Who was violently feared? The second response, the city councilmen, is given. According to the WSC, the sentence has two words: A "distinctive word" and a "reciprocal word." When the former is substituted for the latter, the pronoun resolution changes. In the previous example, the distinctive word is feared, and the reciprocal word is advocated. As a result, each schema shows a set of two roughly similar but distinct coreference resolution issues. Levesque, Davis, and Morgenstern suggested constructing a series of "Google-proof" Winograd Schema, in the sense that properties of the special word alone and its statistical alternative would not explain changing the response when the words are exchanged. A framework would need to "think" in order to comprehend situations of this type, using pertinent previous information.

Related Work

There are numerous methods that have been suggested for solving the Winograd Schema Challenge. These strategies fall into three general categories:

1. Among the methods are those that concentrate on defining the theories of reasoning. Bailey *et al.* (2015); Schüller (2014); Sharma *et al*. (2015b); Wolff (2018). These strategies state a need for additional knowledge and justification, but they are suffering from the WSC Corpus problem of low coverage
2. An alternative set of techniques addresses the underlying theory of information retrieval in a cooperative way. These methods include fly knowledge extraction and knowledge extraction from a pre-populated knowledge base Sharma *et al.* (2015b); Emami *et al*. (2016); Isaak and Michael (2016). The heuristic techniques are a prerequisite for these strategies. More recently, the problem has been addressed with composition embedding techniques and statistical language modeling Wang and Sadrzadeh (2023); Radford *et al*. (2019); Lo *et al.* (2023). By embedding words and sentences, these methods try to capture the information and then utilize it to determine which phrase is most plausible. It is useful when the information sought is concentrated on the potential relationship between two terms, such as in the case of "a ball is kicked," when there is a relationship between kicked and ball. In the case of the Winograd Schema Challenge question "Fish eat the worm," however, it is not possible to deduce that "worm is tasty." However, since "fish" and "tasty" have a higher probability of occurring in the same corpus, it is more probable to consider "fish is tasty" than "it was tasty"
3. Other approaches that discuss the resolution the by finding the sentences that are like the sentences in a WSC problem but without the co-reference ambiguity Sharma *et al*. (2015b); Emami *et al.* (2016). For example, requires the commonsense knowledge that 'something that is eaten may be tasty' is the same 'The fish ate the worm. It was tasty'

In this research, we focused on Commonsense reasoning methods since a human-level accurate response cannot be generated statistically. The major focus of Commonsense reasoning WSC Levesque *et al*. (2012) is to clarify the referential ambiguity in a pair of statements so a binary question about these sentences is also included because WSC's goal is to provide an accurate answer to this binary question using.

For instance, a fish ate a worm. It was tasty. The binary question that requires a response is: What was tasty? Worms or fish? To accurately answer the question, we must determine what the pronoun "it" refers to in the second phrase. Different strategies, however, have been proposed to deal with this issue. The primary strategies suggested are as follows.

Based on commonsense reasoning, many works have been proposed as solutions to the issue Bailey *et al.* (2015); Schüller (2014); Richard-Bollans *et al.* (2018); Wolff (2018); Sharma *et al.* (2015a); Emami *et al.* (2016); Liu *et al.* (2017). These approaches have limitations due to the WSC corpus's low coverage, which necessitates additional knowledge and justifications.

The work of Sharma *et al.* (2015a) created the knowledge parser (K-Parser), a semantic parser, to extract knowledge from text collections. Their research only addressed a subset of Winograd schemas. Bailey *et al.* (2015) have suggested a method for dealing with the Winograd problems that is based on the correlation (positive and negative) between the sentences. A framework for reasoning sentence correlation has been introduced and it has been shown that this framework can be utilized to provide solutions to some Winograd Schema problems. Schüller (2014) used the Stanford dependency parser to convert the Winograd sentence to a dependency graph and combined it with manually created background knowledge to answer the question. Their method does not,

however, automatically extract commonsense knowledge. Emami *et al*. (2016) approached WSC by creating queries from the question, utilizing information retrieval to extract pertinent knowledge about the phrases, and then using that knowledge to reason. With the relevant works, they achieved a competitive performance.

The general semantic parser SemETAP, a knowledge-based semantic parser, was employed by Boguslavskiy Margolin *et al*. (2019) to tackle the WSC. SemETAP uses both standard and enhanced semantic structures. The former addresses the isolated sentence's semantics, while the latter adds inferences based on the knowledge at hand. They demonstrated that the WSC test can typically be passed if the background information is complete and accurate and the explanation of the outcome is simple enough for humans to understand.

Recent proposals for solutions to the issue include composition embedding Liu *et al*. (2017) and statistical language modeling Radford *et al*. (2019).

In order to better comprehend the relationships between words in a phrase, such as which words depend on other words for grammar or meaning, we applied a deep learning dependency parser in this study.

Deep Learning Based for Dependency Relations Extraction

Dependency structures and constituency structures are two main kinds of structures that are employed to show syntactic representations of texts. The dependency tree is one way to reflect dependency structures by presenting graphic arrows between words of a sentence that point from the head to the dependent. Usually, these dependencies that form a dependency tree are typed by grammatical or syntactical relations (subject, object, root, etc). For generating a dependency tree, the dependency parsing task is required.

There are many different implementations for dependency parsing progress such as Kübler *et al*. (2009) that utilized feature-based discriminative to attain. In these parsers, the subclass of transition-based dependency parsers has particularly attracted attention due to its speed in actual implementations that are needed in practical applications. But some applications need accuracy and commonsense knowledge such as the WSC challenge and these parsers aren't worth in these applications. Also, these parsers are not faultless as explained in Chen and Manning (2014). They are flawed statistically because they use millions of mostly inaccurate feature weights, so other methods for adding higher-support characteristics, including word class features, have also proved quite effective at enhancing parsing performance Koo *et al*. (2008). In addition, most new existing parsers are based on a manually created collection of feature templates, which are frequently imperfect and demand a high level of knowledge.

In this research, we focused on a deep learning dependency parse tree that reflects dependency structures and we reused the dependency parser using neural networks that are more fast and accurate Chen and Manning (2014). The goal of a deep learning parser is to predicate a sequence of transitions from the initial state to the terminal state. There are five facts about feed word neural networks:

- Consist of composed node layers
- Used a nonlinear function
- Data is passed between nodes feed word
- Rely on training data with feature selection
- Neural networks have different types such as CNN and RNN

The main findings of this study include demonstrating the value of dense structures learned for the parsing task, introducing an unusual activation function for the neural network that more effectively captures higher-order relationship features, and creating an accurate and quick neural network architecture. This study generated a greedy dependency parsing based on feedword neural networks that correctly anticipate the next transition M from SHIFT, LEFT ARC, and RIGHT ARC operations. To predicate the transition, the study depended on the feature selection that included some subsets: Sentence word, Sentence tag, and Sentence label where the sentence word represents some words of the sentence and their dependents at the top of the buffer and stack, sentence tag represent part of speech tagging for some words of a sentence such as{DT, NN, NNS, JJ, NNP ...} and the sentence label represents the some the arc labels for some words of sentence such as {tmod, nsubj, csubj, dobj, amod ...}. Figure 1 showed Feedforward neural network model for dependency relations.

Syntactic Graphical Representation of a WSC Challenge

To solve the Winograd Schema Challenge problem correctly, it is necessary to use commonsense knowledge so that it is worth extracting graphical syntax and semantic information together from input text. Graphical representation can differentiate between text environment and their events, it is capable of representing the same events or entities from different perspectives and uses a general collection of event-participant relationships. In recent years, there have been done many of works to convert English text into a semantic representation that may be utilized for tasks that require reasoning on either the semantics or syntax of the language such as Sharma *et al*. (2015b). The remainder of this section defines a graphical representation of a set of statements in WSC. In this study, non-linguists who wish to extract textual relations can benefit from a simple description of the grammatical relationships in a sentence provided by the Stanford Dependencies Representation (SDR) de Marneffe *et al*. (2006). The SDR

function converts each WSC sentence token into a dependency relationship. Stanford Parser was utilized to get tokenization and Stanford Dependency Relations. The Stanford Dependency Parser is a natural language processing tool developed at Stanford University de Marneffe *et al.* (2006) that is used to analyze sentence syntax. Dependency trees, which are constructed for individual sentences, show the grammatical relationships between words.

To ascertain a sentence's syntactic structure, the parser applies a set of grammatical rules and statistical models. It examines each word in the phrase and categorizes it into one of several roles, including object, modifier, subject, and so on.

Definition 1. (Tokens for a set of related sentences). Assume $N = (N1, N2, ..., Nn)$, $n >= 1$, be a set of related English sentences, T_i is the set of tokens in the sentence, and $TN = T1T2...T_n$ is the concatenation of the token sequences. Then the set of tokens $F(T)$ is defined as follows: $T(N) = \{t_i | t$ is the i^{th} token in WSC$\}$.

Example 4.1. "the city councilmen refused the demonstrators a permit because they feared violence."

Figure 2 shows the tokenization for Example 4.1 using the Stanford parser.

Definition 2. (SDR function) Let N be a group of connected English phrases., $T(N)$ be a set of tokens in N then the SDR Stanford Dependencies Relations function f SDR N maps each element in $T(N)$ to an element in set {dobj, idobj, nsubj, root, advc1, other}.

Figure 3 shows Stanford dependency relations of example 4.1 using the Stanford dependency parser tool.

Definition 3. (Mapping Class Function for Dependency) Let $T(N)$ be the set of tokens in N and let N be the collection of sentences linked to English. Next, the function for mapping class f_N^{DC} maps a token of $T(N)$ to a related dependency relation in a set DC, i.e., $f_N^{DC}\ T(N) \rightarrow C$ where the set DC is a union of sets $DC1$, $DC2$ and {0} such that,

$DC1$ = {Root, advc1} where Root represents the sentence's main idea that is indicated by the sentence's root grammatical relationship, and advc1 is a clause that modifies the verb (temporal clause, consequence, conditional clause, purpose clause, etc.) and knowns an adverbial clause modifier of a *S* or *VP*.

$DC2$ = {nusbj, dobj, idobj, pobj} where nsubj is a noun phrase that serves as the syntactic subject of a clause is known as a nominal subject, dobj is a noun phrase that serves as the verb's (accusative) object is the direct object of a VP, idobj is the noun phrase that serves as the verb's (dative) object is the indirect object of a VP and pobj is the head of a noun phrase that follows the preposition, or the adverbs "here" and "there," and serves as the object of a preposition.

Let us consider the set of related English sentences shown in Example 4.1, then the tokens in the set of related sentences are shown in Fig. 2 and the mapping produced by the f_N^{DC} function is:

f_N^{DC} (The) = {other}
f_N^{DC} (City) = {other}
f_N^{DC} (Councilmen) = {nsubj}
f_N^{DC} (refused) = {Root}
f_N^{DC} (the) = {other}
f_N^{DC} (demonstrators) = {idobj}
f_N^{DC} (a) = {other}
f_N^{DC} (permit) = {other}
f_N^{DC} (because) = {other}
f_N^{DC} (they) = {nsubj}
f_N^{DC} (feared) = {advc1}
f_N^{DC} (violence) = {obj}

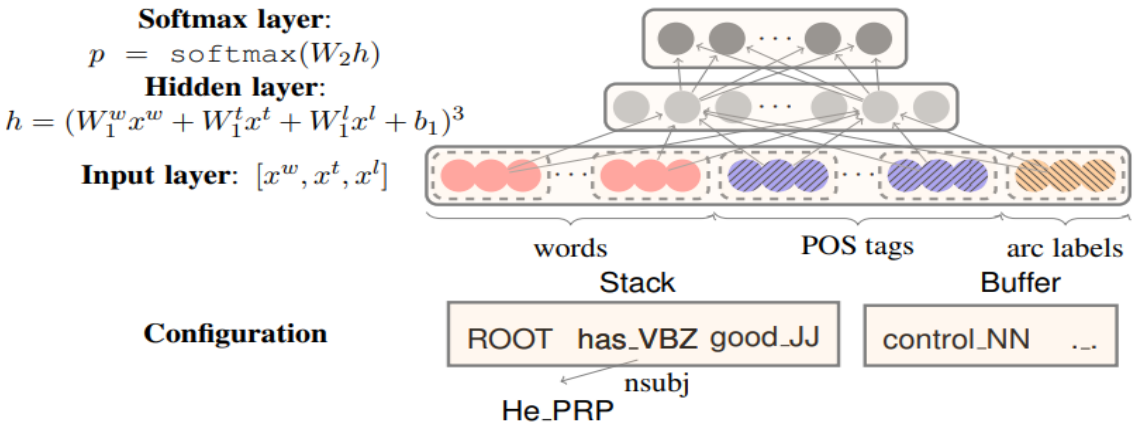

Fig. 1: Feedforward neural network model for dependency relations

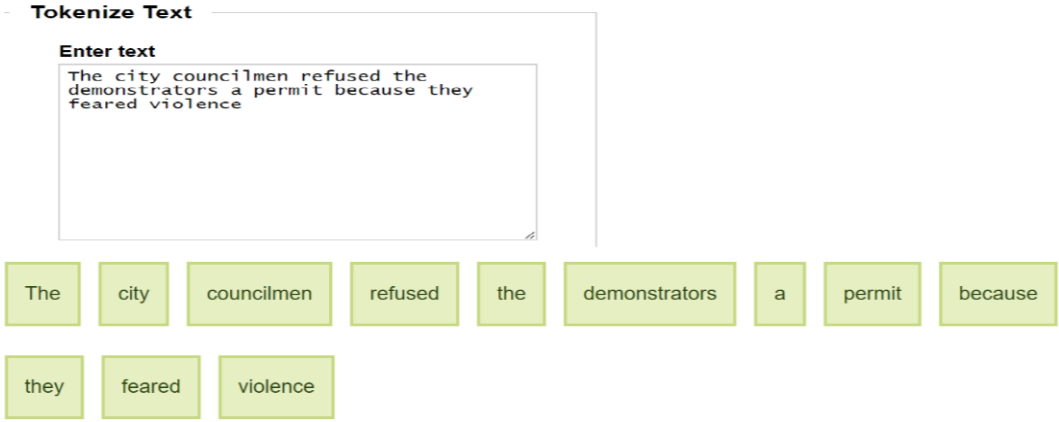

Fig. 2: Tokenization for Example 4.1 using NLTK tool

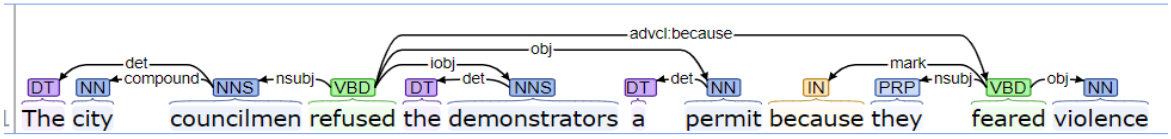

Fig. 3: Stanford dependency relations for Example 4.1 using Stanford dependency parser tool

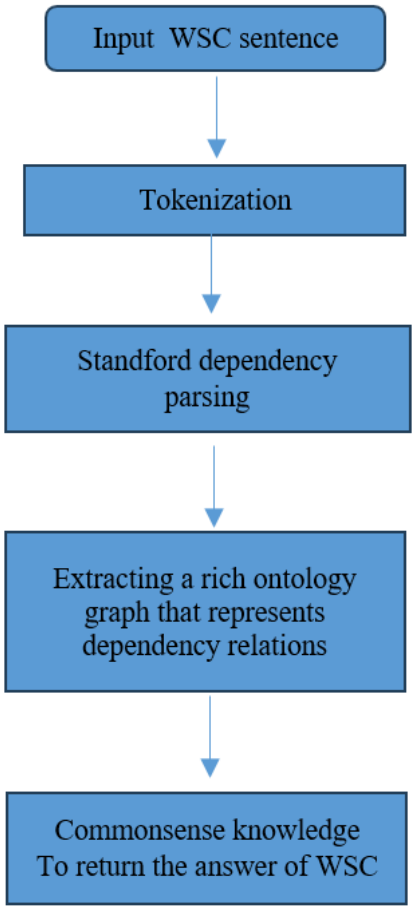

Fig. 4: Proposed method

Materials and Methods

The proposed method is based on four basic tasks: Tokenization, Stanford dependency parsing, extracting a rich ontology graph that represents dependency relations, and the commonsense knowledge to return the answer of WSC schema. Figure 4 shows the main steps of the proposed method.

A good dependency relations representation of a text uses a general set of relations between the events and the participants, can express the text's structure, and can represent the same events or entities from various views. To construct a syntactic representation of the input text, we employed an ontology graph-based syntactic parser (Stanford Dependency Parser). Figure 3 shows an en example that extracting the dependency relations representation of a WSC sentence.

The Stanford dependency parser is a good parser because it has graphical relations that are familiar to read and it has a rich ontology graph that reflects dependency relations to represent the universal and existential entities and events.

Within the WSC corpus, there are 282 sentence and question pairings. A portion of the WSC corpus is used to assess the suggested technique. The subset includes a sizeable portion of Winograd schema, which covers two distinct categories of commonsense knowledge that are described by Sharma *et al*. (2015a):

- Causal Attributive: In this category, the necessary commonsense knowledge has an event and each participant entity's associated characteristic is causally tied to the event. For example, "The man could not leave his son because he is weak." and the

question ”Who is weak?", the anticipated response is “man”. The sort of commonsense knowledge needed to arrive at this conclusion is that nsubj could not leave obj maybe due to nsubj being weak. Here nsubj is the entity and could not leave is the main event in the structure of the sentence

- Direct Causal Events: The commonsense knowledge needed in this category has two mutually causal events where a pronoun participates in one and its candidate co-referent participates in another. For example, “Lindsey wanted to write a letter to Betty even though she knew would never send it.” To reach this conclusion, one must possess a commonsense understanding: IF nsubj1 wanted S to obj but nsubj2 never sent something. THEN nsubj1= nsubj2
- In order to recover commonsense Knowledge into a given sentence and the related question for finding the conclusion that the answer question, we combine the Syntactic ontology graphical representation as described in the previous section with commonsense knowledge by replacing the main events in a sentence with their synonyms. For example, “wanted to write a letter" was replaced with “required to write a letter”

Results and Discussion

For evaluating WSC, first, we evaluated syntactic ontology representations extracted by the Stanford Dependency parser and they reflected the entire system that it relates to the WSC as the previous related work by Sharma *et al*. (2015a). This related work used precision and recall measures to evaluate K-parser manually to extract dependency relations. However, we used precision and recall measures to automatically evaluate the entire system that depends on the Stanford Dependency Parser.

Two crucial measures for assessing classification and natural language processing models are precision and recall Fränti and Mariescu-Istodor (2023), especially when dealing with binary classification issues. These metrics consider many characteristics of a model’s predictions in order to evaluate the model’s performance. Table 1 showed Recall and precision contingency table.

Table 1: Recall and precision contingency table

	Non relevant	Relevant
Retrieved	False positive(fp)	True positives (tp)
Non-retrieved	True negative (TN)	False negatives (fn)

Table 2: Precision and recall for Stanford dependency parser

	Precision-Recall	
Root and advc1dependency type	0.94	0.92
nsubj Dependency type	0.96	0.94
dobj, idobj, pobj dependency type	0.92	0.83

In information retrieval precision and recall could be defined as the following.

Precision is calculated as the total number of recovered items divided by the number of relevant retrieved items.

The number of relevant items retrieved divided by the total number of relevant elements equals recall.

Table 2 shows the precision and recall for the fast deep-learning Stanford parser with highly Accurate results Chen and Manning (2014).

As future work to improve the results, we will use Answer Set Programming (ASP) Baral (2003) because we want the process of adding additional restrictions to be as simple as possible. It is crucial to the algorithm’s isomorphism detection stage, which pairs the nodes of two graphs according to a set of restrictions.

Conclusion

Pronominal anaphora resolution challenges that call for the use of cognitive inference in conjunction with domain knowledge are the focus of the Winograd Schema Challenge. While these issues are relatively simple for people to solve, they are quite challenging for machines to tackle. The Stanford NLP Group's Deep-learning Stanford dependency parser was employed in this study as a natural language processing tool to handle the Winograd representation suitably. The precision and recall for the quick deep learning Stanford parser with extremely accurate results are displayed in Table 2 by Chen and Manning (2014).

Acknowledgment

We extend our heartfelt gratitude to the editor for his invaluable assistance in getting this research report published. Furthermore, we express our gratitude to the editorial team for their painstaking work in examining and editing our work.

Funding Information

Funding or other financial assistance has not been given to the authors.

Author’s Contributions

Nesreen Alsharman: Make considerable contributions to the conception and design and/or acquisition of data and/or analysis and interpretation of data, contribute to drafting the article or reviewing it critically for significant intellectual content. Give final approval of the version to be submitted and any revised version.

Raja Masadeh: Contributed to drafting the article or reviewing it critically for significant intellectual content. Give final approval of the version to be submitted and any revised version.

Ibrahim Ali Jawarneh: Give final approval of the version to be submitted and any revised mathematical version.

Ahmad Al-Rababa'a: Revise the final approval of the version to be submitted.

Ethics

This research paper is unique and includes content that hasn't been published before. The corresponding author attests to the fact that there are no ethical concerns and that all other authors have read and approved the paper.

References

Bailey, D., Harrison, A., Lierler, Y., Lifschitz, V., & Michael, J. (2015). The Winograd schema challenge and reasoning about correlation. *2015 AAAI Spring Symposium Series*, 17–24. https://cdn.aaai.org/ocs/10295/10295-45254-1-PB.pdf

Baral, C. (2003). *Knowledge Representation, Reasoning and Declarative Problem Solving*. Cambridge University Press. ISBN-10: 9781139436441.

Bennett, B. (2022). Semantic analysis of Winograd Schema no. 1. *Formal Ontology in Information Systems*, *344*, 33–47. https://doi.org/10.3233/FAIA210369

Boguslavskiy Margolin, I., Frolova, T. I., Iomdin, L. L., Lazursky, A. V., Rygaev, I. P., & Timoshenko, S. P. (2019). Knowledge-based approach to Winograd Schema Challenge. *Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialogue 2019"*. https://oa.upm.es/65046/

Chen, D., & Manning, C. (2014). A Fast and Accurate Dependency Parser using Neural Networks. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 740–750. https://doi.org/10.3115/v1/d14-1082

de Marneffe, M.-C., MacCartney, Bill, & Manning, C. D. (2006). Generating typed dependency parses from phrase structure parses. *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, Genoa, Italy. https://aclanthology.org/L06-1260/

Elazar, Y., Zhang, H., Goldberg, Y., & Roth, D. (2021). Back to Square One: Artifact Detection, Training and Commonsense Disentanglement in the Winograd Schema. *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 10486–10500. https://doi.org/10.18653/v1/2021.emnlp-main.819

Emami, A., La Cruz, N. D., Trischler, A., Suleman, K., & Cheung, J. C. K. (2016). A knowledge hunting framework for common sense reasoning. *ArXiv:1810.01375*. https://doi.org/10.48550/arXiv.1810.01375

Fränti, P., & Mariescu-Istodor, R. (2023). Soft precision and recall. *Pattern Recognition Letters*, *167*, 115–121. https://doi.org/10.1016/j.patrec.2023.02.005

Hong, S. J., Bennett, B., Clymo, J., & Álvarez, L. G. (2022). Karaml: Integrating knowledge-based and machine learning approaches to solve the win grad schema challenge. *Proceedings of the AAAI 2022 Spring Symposium on Machine Learning and Knowledge Engineering for Hybrid Intelligence (AAAI-MAKE 2022)*, 1–16. https://iccl.inf.tu-dresden.de/w/images/7/78/KARAML.pdf

Jurafsky, D., & Martin, J. H. (2023). Speech andLanguage Processing. https://web.stanford.edu/~jurafsky/slp3/.

Koo, T., Carreras, X., & Collins, M. (2008). Simple semi-supervised dependency parsing. *Proceedings of ACL-08: HLT*, 595–603. https://aclanthology.org/P08-1068.pdf

Kübler, S., McDonald, R., & Nivre, J. (2009). *Dependency Parsing* (pp. 11–20). Springer International Publishing. https://doi.org/10.1007/978-3-031-02131-2_2

Levesque, H. J., Davis, E., & Morgenstern, L. (2012). The winograd schema challenge. *13th International Conference on the Principles of Knowledge Representation and Reasoning, KR 2012*, 552–561. https://nyuscholars.nyu.edu/en/publications/the-winograd-schema-challenge-2

Liu, Q., Jiang, H., Evdokimov, A., Ling, Z.-H., Zhu, X., Wei, S., & Hu, Y. (2017). Cause-Effect Knowledge Acquisition and Neural Association Model for Solving A Set of Winograd Schema Problems. *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, 2344–2350. https://doi.org/10.24963/ijcai.2017/326

Lo, K. I., Sadrzadeh, M., & Mansfield, S. (2023). Generalised Winograd Schema and its Contextuality. *ArXiv:2308.16498*, *384*, 187–202. https://doi.org/10.4204/eptcs.384.11

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. *OpenAI Blog*, *1*(8), 9. https://insightcivic.s3.us-east-1.amazonaws.com/language-models.pdf

Richard-Bollans, A. L., Gomez Alvarez, L., & Cohn, A. G. (2018, January). The role of pragmatics in solving the Winograd Schema Challenge. In *Proceedings of the Thirteenth International Symposium on Commonsense Reasoning (Commonsense 2017)*. CEUR Workshop Proceedings. https://eprints.whiterose.ac.uk/122937/

Schüller, P. (2014). Tackling winograd schemas by formalizing relevance theory in knowledge graphs. In *Proceedings of the Fourteenth International Conference on Principles of Knowledge Representation and Reasoning* (pp. 358-367). https://dl.acm.org/doi/abs/10.5555/3031929.3031973

Sharma, A., Vo, N. H., Aditya, S., & Baral, C. (2015a). Towards Addressing the Winograd Schema Challenge — Building and Using a Semantic Parser and a Knowledge Hunting Module. *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)*, 1319–1325. https://teaching.bb-ai.net/Student-Projects/Winograd-Challenge-Papers/Sharma-semantic-parser.pdf

Sharma, A., Vo, N., Aditya, S., & Baral, C. (2015b). Identifying Various Kinds of Event Mentions in K-Parser Output. *Proceedings of the The 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, 82–88. https://doi.org/10.3115/v1/w15-0811

Takahashi, K., Oka, T., & Komachi, M. (2023). Effectiveness of Pre-Trained Language Models for the Japanese Winograd Schema Challenge. *Journal of Advanced Computational Intelligence and Intelligent Informatics*, *27*(3), 511–521. https://doi.org/10.20965/jaciii.2023.p0511

Wang, D., & Sadrzadeh, M. (2023). The Causal Structure of Semantic Ambiguities. *ArXiv:2206.06807v3*, *394*, 208–220. https://doi.org/10.4204/eptcs.394.12

Weston, J., Bordes, A., Chopra, S., Rush, A. M., Merriënboer, B. van, Joulin, Armand, & Mikolov, T. (2015). Towards ai-complete question answering: A set of prerequisite toy tasks. *ArXiv:1502.05698*. https://doi.org/10.48550/arXiv.1502.05698

Wolff, J. G. (2018). Interpreting Winograd Schemas Via the SP Theory of Intelligence and Its Realisation in the SP Computer Model. *ArXiv:1810.04554*. https://doi.org/10.48550/arXiv.1810.04554