Precise Task Formalization Matters in Winograd Schema Evaluations

Haokun Liu[1*] **William Huang**[2*] **Dhara A. Mungra**[1] **Samuel R. Bowman**[1,2,3]
[1]Center for Data Science, New York University
[2]Courant Institute of Mathematical Sciences, New York University
[3]Department of Linguistics, New York University
{haokunliu, wh629, dam797, bowman}@nyu.edu

Abstract

Performance on the Winograd Schema Challenge (WSC), a respected English commonsense reasoning benchmark, recently rocketed from chance accuracy to 89% on the SuperGLUE leaderboard, with relatively little corroborating evidence of a correspondingly large improvement in reasoning ability. We hypothesize that much of this improvement comes from recent changes in task formalization—the combination of input specification, loss function, and reuse of pretrained parameters—by users of the dataset, rather than improvements in the pretrained model's reasoning ability. We perform an ablation on two Winograd Schema datasets that interpolates between the formalizations used before and after this surge, and find (i) framing the task as multiple choice improves performance by 2-6 points and (ii) several additional techniques, including the reuse of a pretrained language modeling head, can mitigate the model's extreme sensitivity to hyperparameters. We urge future benchmark creators to impose additional structure to minimize the impact of formalization decisions on reported results.

1 Introduction

Over the last couple of years, large pretrained models have achieved human performance on a large share of established natural language understanding benchmark datasets (Devlin et al., 2019). Recent results report a surge in performance to near-human levels on the Winograd Schema Challenge (WSC; Levesque, 2011) in particular(Liu et al., 2019). However, variations in task formulation across papers and evaluations makes it hard to understand the true degree of recent progress.

The WSC is an English commonsense reasoning evaluation that requires a model to resolve carefully-constructed ambiguous pronouns. For example, in the sentence "*Jim yelled at Kevin because **he** was so upset.*" the reader will likely have to consider the motivation of the query noun phrase (NP) to recognize whether the pronoun ***he*** refers to *Jim*.

The accuracy of WSC has seen an abrupt increase from 64% to 89% on the SuperGLUE (Wang et al., 2019) leaderboard upon the release of RoBERTa (Liu et al., 2019). While many works (Kocijan et al., 2019; Liu et al., 2019; Raffel et al., 2020) attribute such improvements to improved pretraining and the use of auxiliary training datasets, the impact of the task formalization—the combination of input specification, task specific layer design, and loss function—has not yet been seriously studied.

The SuperGLUE WSC baseline with BERT (64%) resolves pronoun references for individual examples by concatenating the pronoun and query NP embeddings and making a binary prediction for the NP span. Meanwhile, RoBERTa (89%) uses a pretrained masked language modeling (MLM) head as part of the output layer and treats the task as a multiple-choice (MC) decision between candidate NPs. We refer to these two task formalizations as pointwise span (**P-Span**) and multiple choice masked language modeling (**MC-MLM**).

In our work, we interpolate between P-Span and MC-MLM using both BERT and RoBERTa to understand tasks' sensitivity to formalization and the components contributing to MC-MLM's improvement. We find MC-MLM outperforms P-Span and reduces sensitivity to hyperparameters and random restarts. We also see large variances of scores spanning random guessing to state-of-the-art (SotA) performance. The biggest gain comes from including MC inference. Further, paired training with query and candidate NPs, using a softmax over candidates, and using a pretrained MLM head all

[*] Equal contribution.

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 8275–8280,
November 16–20, 2020. ©2020 Association for Computational Linguistics

Ablation	Formalization	Input Example	Emb	Loss	MC	Label
Strong Baseline (RoBERTa method)	MC-MLM	a) `[CLS]` Jim yelled at Kevin because `[MASK]` was so upset. `[SEP]` b) `[CLS]` Jim yelled at Kevin because `[MASK]` was so upset. `[SEP]`	`[MASK]`	$-\sum_{i}^{n} \delta_{y,i} \log \frac{\mathbb{P}(\text{NP}_i \mid \text{s})}{\sum_j^n \mathbb{P}(\text{NP}_j \mid \text{s})}$	✓	Index
No MLM (WinoGrande method)	MC-Sent		`[CLS]`	$-\sum_{i}^{n} \delta_{y,i} \log \mathbb{P}(\text{NP}_i \mid \text{s}_1, \ldots, \text{s}_n)$	✓	Index
No Softmax Scaling	MC-Sent-NoSoftmax	a) `[CLS]` Jim yelled at Kevin because Jim `[SEP]` was so upset. `[SEP]`		$-\sum_{i}^{n} \big[y_i \log \mathbb{P}(\text{True} \mid \text{s}_i) + (1-y_i) \log \mathbb{P}(\text{False} \mid \text{s}_i)\big]$	✓	Binary
No Paired Training	MC-Sent-NoPairLoss	b) `[CLS]` Jim yelled at Kevin because Kevin `[SEP]` was so upset. `[SEP]`		$-\big[y \log \mathbb{P}(\text{True} \mid \text{s}) + (1-y) \log \mathbb{P}(\text{False} \mid \text{s})\big]$	✓	Binary
No MC Evaluation	P-Sent			$-\big[y \log \mathbb{P}(\text{True} \mid \text{s}) + (1-y) \log \mathbb{P}(\text{False} \mid \text{s})\big]$	✗	Binary
Predict from Span (SuperGLUE method)	P-Span	`[CLS]` Jim yelled at Kevin because he was so upset. `[SEP]`	`[CLS]`, PRON, NP	$-\big[y \log \mathbb{P}(\text{True} \mid \text{s}) + (1-y) \log \mathbb{P}(\text{False} \mid \text{s})\big]$	✗	Binary

Table 1: Overview of the formalizations. When *MC* (multiple choice) is ✓, the model predicts positive if the query NP $\mathbb{P}(\cdot)$ is highest among all candidates; when *MC* is ✗, the model predicts positive if $\mathbb{P}(\cdot) > 0.5$. *Emb* indicates which RoBERTa output layer embeddings are used. In the loss function, y is the index of the correct input when *Label* is *Index* and 0 or 1 when *Label* is *Binary*. *s* is a sequence of input tokens, we use subscript to indicate multiple input sequences. $\delta_{y,i}$ is 1 when $y = i$ i.e. the i-th input is correct and 0 otherwise. In the MC-MLM input example, the underline marks NPs to predict. For P-Span, the underline marks the NP and PRON spans.

lead to reductions in variance. We show that these formalization choices impact performance differences between the BERT and RoBERTa approaches on SuperGLUE WSC, with validation accuracy increasing between P-Span and MC-MLM by 21.1% using RoBERTa and 10.5% using BERT.

The effect of task formalization may incentivize gains from supplemental MC options and aggressive hyperparameter tuning. To avoid this, we suggest future benchmarks impose more structure, such as in this case either explicitly distributing gold candidate NPs or enforcing rules against their use. For system developers, this result highlights the value of fine-tuning pretrained language modeling heads to target tasks in low-resource settings (Raffel et al., 2020), at least where the task format makes this an option.

2 Related Work

WSC Datasets Levesque et al. (2012) launch the WSC with 108 handbuilt question-answer pairs, which has since grown to 273 examples, often called WSC273. Since then, several similar or derived datasets have emerged (Kocijan et al., 2020). The SuperGLUE version of the task recasts examples from WSC273 and the Pronoun Disambiguation Problems dataset (Morgenstern et al., 2016) into 554 training and 104 validation binary classification problems. 146 test examples are derived from fiction books and handcrafted by the original WSC authors. Sakaguchi et al. (2020) collect a larger dataset of fill-in-the-blank-format problems, called WinoGrande, via crowdsourcing with adversarial filtering. The dataset has five training sets, ranging from 160 to 41k examples, and shared validation and test sets with 1.3k and 1.8k examples, respectively.

Approaches Trinh and Le (2018) generate inputs for their recurrent neural network language model by replacing the PRON with either the query or candidate NP and compare the probability of the two sentences, yielding 64% accuracy on WSC273. Radford et al. (2019) use this method with a transformer language model, boosting accuracy to 71%. Ruan et al. (2019) fine-tune BERT in a similar way, reaching 71% as well. Kocijan et al. (2019) also use BERT, but include additional Winograd-like training data and use the model's pretrained MLM head to achieve 74% accuracy. In another style of approach, Klein and Nabi (2019) experiment with BERT pretrained *attention* weights without fine-tuning, and achieve an accuracy of 60%.

For SuperGLUE WSC, the official baseline uses BERT and a linear classifier on BERT's output embeddings for the `[CLS]` token, pronoun token, and query NP span representations but fail to exceed the majority-class baseline, only matching it at 64%. Liu et al. (2019) use the newer RoBERTa and adapt the Kocijan et al. (2019) approach with cross entropy loss to raise this accuracy to 89%. T5 (Raffel et al., 2020) marks the pronoun in the input and fine-tune a transformer encoder-decoder model to generate the target NP, achieving the current state of the art at 94%.

Looking to WinoGrande, Sakaguchi et al. (2020) adapt Ruan et al. (2019)'s method with RoBERTa as the baseline model, achieving 68% accuracy on WinoGrande-Medium and 79% accuracy on the full test set.

3 Methods under Study

We evaluate six formalizations—three existing ones and three that we introduce—to interpolate between P-Span and MC-MLM. These all use an output layer on top of an MLM pretrained transformer model, but differ in the input specification, loss function, prediction method, contextual embeddings used by the output layer, and label type. Table 1 presents an overview.

MC-MLM This approach follows that of Liu et al. (2019) in the introduction of RoBERTa. Here, the pronoun in the input is replaced by `[MASK]`. The model then uses its pretrained MLM head to evaluate the probability NP_i should replace `[MASK]` and uses a softmax over the log probabilities. For multi-token NPs the model compares the *geometric mean* of these probabilities.

MC-Sent This approach follows the WinoGrande baselines. Here, we specify the inputs by replacing the pronoun with an NP candidate and marking it with an additional `[SEP]` token. The output head feeds each option's `[CLS]` embedding into a linear layer and applies a softmax over the outputs. MC-Sent trains a linear layer from scratch, while MC-MLM may take advantage of the embedding model's MLM pretraining.

MC-Sent-NoSoftmax MC-Sent-NoSoftmax only differs from MC-Sent by replacing the final softmax with a sigmoid and computes the probabilities of whether each input sequence is correct. Without softmax, MC-Sent-NoSoftmax is unable to provide larger gradients for examples with smaller margins between candidates. We refer to this as softmax scaling.

MC-Sent-NoPairLoss MC-Sent-NoPairLoss and MC-Sent-NoSoftmax differ by loss function, where MC-Sent-NoPairLoss only considers the query input. MC-Sent-NoPairLoss is unable to use gradients from multiple candidates to neutralize signals from shared words and focus on NP options. We refer to this as paired training.

P-Sent In P-Sent, we further remove *MC evaluation* by restricting the model to a single binary classification question. This forces P-Sent to resolve pronoun references without implicitly learning to detect and eliminate NPs.

P-Span Instead of replacing PRON with NPs to determine the validity of the input sentence, P-Span follows the SuperGLUE baseline to determine whether the NP reference is correct. It first averages over the representations from the PRON and NP spans to create span representations. The span representations are then concatenated with the `[CLS]` token embedding and used by a logistic regression classifier.

4 Experiments

Implementation Our code[1] builds on Huggingface Transformers (Wolf et al., 2019) and fairseq (Ott et al., 2019). All our experiments use either pretrained RoBERTa-large or BERT-large-cased models. We evaluate on the validation set every epoch with early stopping. We conduct a random hyperparameter search of 60 trials over the space of learning rate {1e-5, 2e-5, 3e-5}, epoch limit {10, 20, 40}, batch size {8, 16, 32, 64}, and random seed.

Datasets We run experiments on SuperGLUE WSC and WinoGrande-Medium. We do not cover larger WinoGrande sizes due to computation constraints. Each WSC example includes a sentence, a PRON span, and a *single* marked NP span. Following Liu et al., for MC-based formalizations, we mine candidate NPs with spaCy[2] and only keep one example from the group of examples that only differ by query NP to avoid redundancy.

[1] https://github.com/nyu-mll/wsc-formalizations/tree/code_release

[2] https://spacy.io/

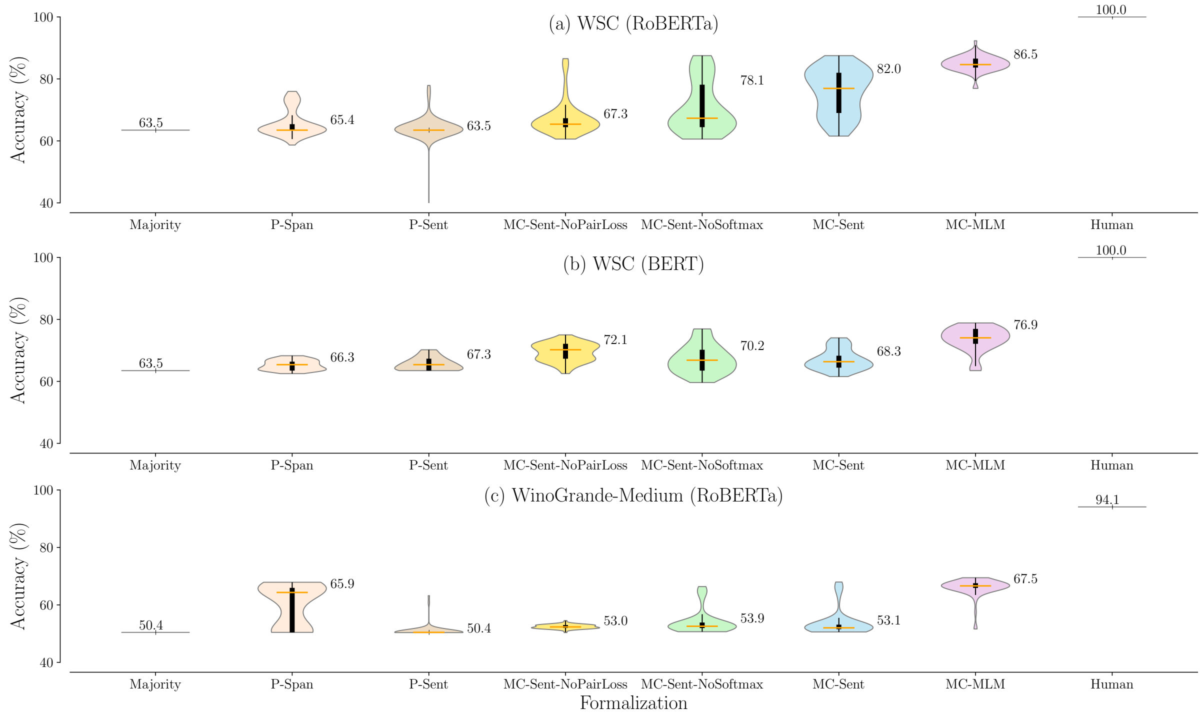

Figure 1: Plots of validation accuracy from 60 runs on each corpus. The orange line marks the median number and label marks 75th percentiles.

		WSC		WinoGrande	
Formalization	Test	Std	Kurt	Std	Kurt
MC-MLM	86	2	3	3	13
MC-Sent	77	7	-1	5	4
MC-Sent-NoSoftmax	77	8	-1	4	3
MC-Sent-NoPairLoss	86	6	5	1	1
P-Sent	67	5	24	2	28
P-Span	80	4	0	7	-2

Table 2: Validation accuracy standard deviation (*Std*) and excess kurtosis (*Kurt*) for WSC and WinoGrande and test accuracy for WSC using RoBERTa. Test results are from an ensemble of the top five models.

For WinoGrande, each example provides a sentence with two marked NP spans and a fill-in-the-blank to represent the PRON. When using asymmetric formalizations like P-Sent, we duplicate each example, making one option the query and the other the candidate. For P-Span, we use the first appearance of query or candidate NP in the sentence as the NP span and use the blank as PRON span.

RoBERTa Results Figures 1a and 1c and Table 2 show the distribution over validation accuracies from 60 runs with each formalization using RoBERTa. We do not report WinoGrande test results since submissions require test set predictions from all five training sets and we only train using WinoGrande-Medium. We also include the majority-class baseline and human performance. From the WSC test results, we find MC-MLM outperforms P-Span. The 6% gain between P-Sent and MC-Sent-NoPairLoss indicates MC evaluation alone may improve accuracy. However, we also find most formalizations are sensitive to hyperparameter choices and random seeds. Given the small size of the SuperGLUE WSC test set at 146 examples, we find it more informative to focus on the distribution of validation results.

In both datasets, we see three main changes. First, including paired training with MC-Sent-NoSoftmax increases performance variance by adding more weight to the tail of higher scores. Second, we see the weight of higher performances increase even more with softmax scaling in MC-Sent. In WSC, the higher scores become the body of the distribution with smaller variance. In WinoGrande, the distribution of MC-Sent has an increased excess kurtosis indicating the tail of higher scores occur more frequently. Finally, the model achieves higher scores with significantly lower variance using MLM in MC-MLM. This may be a result of fine-tuning the pretrained MLM head rather than a new initialization.

We see two main differences between WSC and

WinoGrande results. First, P-Sent performs significantly worse than P-Span on WinoGrande. We suspect this is due to WinoGrande's adversarial filtering removing examples that are easy to classify from sentence representations. Second, MC-Sent-NoPairLoss does not benefit WinoGrande and may indicate the benefit from MC evaluation may not extend to other Winograd like corpora.

BERT Results Figure 1b shows ablation results using BERT and WSC. We find that RoBERTa outperforms BERT with both models using the same MC-MLM formalization, which is in line with leaderboard performances. We also find similar trends across task formalizations in Figure 1a, further highlighting the impact of formalization decisions on performance gains. Most formalizations are still sensitive to hyperparameter choices and random seed, MC evaluation alone provides a benefit over P-Span at the 75$^{\text{th}}$ percentile of roughly 6%, and incorporating MLM provides additional benefits in performance.

However, we also find that using BERT's pretrained MLM head does not provide the lower variance displayed with RoBERTa. Comparing the performances of intermediate formalizations, we see that BERT generally performs worse than RoBERTa. This is consistent with the findings from Tenney et al. (2019) that show BERT embeddings encode information less suited for coreference resolution during pretraining. Consequently, BERT's pretrained MLM head would be less optimized for a coreference resolution task like WSC than RoBERTa's and may not provide the same stability benefits.

5 Conclusion

By only varying task formalization, we observe a wide range of results among reasonable task formalizations on WSC and WinoGrande evaluations. Having access to candidate NPs during inference alone improves the performance on SuperGLUE WSC. However, models with MC evaluation are highly sensitive to hyperparameters and fail to perform better on WinoGrande. We find training with paired inputs, using a softmax over candidates, and reusing a pretrained MLM head all help to learn commonsense reasoning and reduce this sensitivity. While we find evidence that these formalization choices can largely influence WSC performance, we do not see obvious evidence of similar occurrences on other task comparisons with RoBERTa.

For MC formalizations, we follow Liu et al. for WSC and use spaCy to mine candidate NPs. This extrinsic preprocessing step yields dramatic gains without significantly changing the reasoning ability of the model. We view such gains as orthogonal to the intent of the task and urge benchmark creators to minimize the opportunity for these insubstantial improvements by imposing as much structure as is possible in the released data, for example, by providing candidate NPs explicitly.

We also encourage future reports of system performances to use the same task formalization whenever possible. At a minimum, greater emphasis should be given to task formalization decisions when they deviate from the prevailing standard. We believe this will help disentangle gains due to models' reasoning abilities, especially in situations where these decisions significantly impact performance, such as in WSC.

Finally, we find that differences between reasonable formalizations can have big impacts on performance with our case study using WSC. For example, using a pretrained MLM task head as the basis for a downstream task classifier yields strong results with very little hyperparameter sensitivity. This echoes the strong results seen with T5 and offers further motivation to explore these kinds of design decisions in other tasks.

Acknowledgements

We thank Vid Kocijan, Ethan Perez, Jason Phang, Aishwarya Kamath, Iacer Calixto, Clara Vania, Rasika Bhalerao, Alex Warstadt, Richard Pang, He He, and Ernie Davis for their helpful feedback. This project has benefited from financial support to SB by Eric and Wendy Schmidt (made by recommendation of the Schmidt Futures program), by Samsung Research (under the project *Improving Deep Learning using Latent Structure*), by Intuit, Inc., and in-kind support by the NYU High-Performance Computing Center and by NVIDIA Corporation (with the donation of a Titan V GPU). This material is based upon work supported by the National Science Foundation under Grant No. 1922658. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Tassilo Klein and Moin Nabi. 2019. Attention is (not) all you need for commonsense reasoning. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4831–4836. Association for Computational Linguistics.

Vid Kocijan, Ana-Maria Cretu, Oana-Maria Camburu, Yordan Yordanov, and Thomas Lukasiewicz. 2019. A surprisingly robust trick for the Winograd Schema Challenge. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4837–4842. Association for Computational Linguistics.

Vid Kocijan, Thomas Lukasiewicz, Ernest Davis, Gary Marcus, and Leora Morgenstern. 2020. A review of Winograd Schema Challenge datasets and approaches. *CoRR*, abs/2004.13831.

Hector J. Levesque. 2011. The Winograd Schema Challenge. *AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning*.

Hector J. Levesque, Ernest Davis, and Leora Morgenstern. 2012. 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*. AAAI Press.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Leora Morgenstern, Ernest Davis, and Charles L. Ortiz Jr. 2016. Planning, executing, and evaluating the Winograd Schema Challenge. *AI Magazine*, 37(1):50–54.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Demonstrations*, pages 48–53. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. Unpublished manuscript available at `https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf`.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Yu-Ping Ruan, Xiaodan Zhu, Zhen-Hua Ling, Zhan Shi, Quan Liu, and Si Wei. 2019. Exploring unsupervised pretraining and sentence structure modelling for Winograd Schema Challenge. *CoRR*, abs/1904.09705.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Winogrande: An adversarial Winograd Schema Challenge at scale. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8732–8740. AAAI Press.

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R. Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R. Bowman, Dipanjan Das, and Ellie Pavlick. 2019. What do you learn from context? probing for sentence structure in contextualized word representations. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

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

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 3261–3275.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. HuggingFace's Transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.