Analyzing Semantic Faithfulness of Language Models via Input Intervention on Conversational Question Answering

Akshay Chaturvedi, Swarnadeep Bhar, Soumadeep Saha, Nicholas Asher, Utpal Garain

Analyzing Semantic Faithfulness of Language Models via Input Intervention on Conversational Question Answering

Akshay Chaturvedi*
IRIT, Université Paul Sabatier, Toulouse, France

Swarnadeep Bhar
IRIT, Université Paul Sabatier, Toulouse, France

Soumadeep Saha
Indian Statistical Institute, Kolkata, India

Utpal Garain
Indian Statistical Institute, Kolkata, India

Nicholas Asher
IRIT, Université Paul Sabatier, Toulouse, France

Transformer-based language models have been shown to be highly effective for several NLP tasks. In this paper, we consider three transformer models, BERT, RoBERTa, and XLNet, in both small and large versions, and investigate how faithful their representations are with respect to the semantic content of texts. We formalize a notion of semantic faithfulness, in which the semantic content of a text should causally figure in a model's inferences in question answering. We then test this notion by observing a model's behavior on answering questions about a story after performing two novel semantic interventions—deletion intervention and negation intervention. While transformer models achieve high performance on standard question answering tasks, we show that they fail to be semantically faithful once we perform these interventions for a significant number of cases ($\sim 50\%$ for deletion intervention, and $\sim 20\%$ drop in accuracy for negation intervention). We then propose an intervention-based training regime that can mitigate the undesirable effects for deletion intervention by a significant margin (from $\sim 50\%$ to $\sim 6\%$). We analyze the inner-workings of the models to better understand the effectiveness of intervention-based training for deletion intervention. But we show that this training does not attenuate other aspects of semantic unfaithfulness such as the models' inability to deal with negation intervention or to capture the predicate-argument structure of texts. We also test InstructGPT, via prompting, for its ability to handle the two interventions and to capture predicate-argument structure. While InstructGPT models do achieve very high performance on predicate-argument structure task, they fail to respond adequately to our deletion and negation interventions.

* E-mail: akshay91.isi@gmail.com.

Story	Once upon a time, in a barn near a farm house, there lived a little white kitten named Cotton. Cotton lived high up [...] farmer's horses slept. **But Cotton wasn't alone in her little home above the barn, oh no.**
Conversation History	What color was Cotton? white Where did she live? in a barn
Question	Did she live alone?
Prediction	no

Table 1: An example from CoQA data set (Reddy, Chen, and Manning 2019). XL-Net (Yang et al. 2019) correctly predicts *no* for the question *"Did she live alone?"*. However, it still predicts *no* when the rationale (i.e., text marked in bold) is removed from the story (i.e., deletion intervention).

1. Introduction

Transformer-based language models such as BERT (Devlin et al. 2019), RoBERTa (Liu et al. 2019b), etc. have revolutionized natural language processing (NLP) research, generating contextualized representations that provide state of the art performance for various tasks like part of speech (POS) tagging, semantic role labelling etc. The transfer learning ability of these models has discarded the need for designing task-specific NLP systems. The latest incarnation of language models have now excited both the imagination and the fears of researchers (Black et al. 2022; Castelvecchi 2022) and journalists in the popular press; the models and chatbots based on them seem to be able to do code, argue and tell stories, but they also have trouble distinguishing fact from fiction.[1]

Given their successes and their hold on the public imagination, researchers are increasingly interested in understanding the *inner workings* of these models (Liu et al. 2019a; Tenney et al. 2019; Talmor et al. 2020). In this paper, we look at how a fundamental property of linguistic meaning we call *semantic faithfulness* is encoded in the contextualized representations of transformer-based language models and how that information is used in inferences by the models when answering questions. A semantically faithful model will accurately track the semantic content of questions and texts on which the answers to those questions are based. It is a crucial property for a model to have, if it is to distinguish facts about what is expressed in a text or conversation from fiction or hallucination. We will show that current, popular transformer models are not semantically faithful.

This lack of semantic faithfulness highlights potential problems with popular language models trained with transformer architectures. If these models are not semantically faithful, then they will fail to capture the actual semantic content of texts. Operations that we develop in the body of the paper can be used to dramatically alter text content that these language models would not find, leading to errors with potentially important, negative socio-economic consequences. Even more worrisome is the instability that we have observed in these models and their occasional failure to

1 Here is a sample of stories from the *New York Times*: 'The New Chatbots Could Change the World. Can You Trust Them?' (*NYT* Dec. 10, 2022; 'Meet GPT-3. It Has Learned to Code (and Blog and Argue)', *NYT*, Nov 24,2020; 'The brilliance and the weirdness of ChatGPT', *NYT*, Dec. 5, 2022.

keep predicate argument structure straight; if these models cannot reliably return information semantically entailed by textual content, then we can't rely on their predictions in many sensitive areas. Yet such systems are being deployed rapidly in these areas.

In the next section, we discuss the virtues of semantic faithfulness and preview results of experiments that shed light on a model's semantic faithfulness. In Section 3, we discuss the related work. In Section 4, we turn to the data set and the transformer models that we will use in examining semantic faithfulness. In Sections 5 and 7, we introduce two types of interventions on texts, deletion and negation interventions, that show that the machine learned models we investigated lack semantic faithfulness. In Section 6, we discuss a kind of training that can help models acquire semantic faithfulness at least with respect to deletion intervention. In Section 8, we look at how models deal with predicate argument structure and with inferences involving semantically equivalent questions. Once again we find that models lack semantic faithfulness. In Section 9, we analyse semantic faithfulness of InstructGPT (Ouyang et al. 2022) via prompting. Finally, we conclude in Sections 10 and 11.

2. The fundamentals: semantic faithfulness

The property of interest is *semantic faithfulness*. It relies on a basic theorem of all formal models of meaning in linguistics: the substitution of semantically equivalent expressions within a larger context should make no difference to the meaning of that context.

2.1 The definition of semantic faithfulness

Let $||\models$ represent the intuitive answerhood relation between a question Q and answers ϕ, ψ to Q, where those answers follow from the semantic content of a story or text T or model of its meaning M_T. Let $\models$ represent semantic entailment as defined in formal semantics—e.g., in (Dowty, Wall, and Peters 1981).

Definition [Semantic faithfulnesss]: If $T \models \phi \leftrightarrow \psi$, and $T \models Q \leftrightarrow Q'$, then M_T is a semantically faithful model of T iff:

$$M_T, Q \;||\models\; \phi \text{ iff } M_T, Q \;||\models\; \psi \tag{1}$$

and

$$M_T, Q \;||\models\; \psi \text{ iff } M_T, Q' \;||\models\; \psi \tag{2}$$

Note that if $T \models Q \leftrightarrow Q'$ and $T \models \phi \leftrightarrow \psi$, then by the substitution of equal semantic values, it follows in formal semantics that $T, Q \;||\models\; \phi$ iff $T, Q' \;||\models\; \psi$.

A semantically faithful machine learning model of meaning and question answering bases its answers to questions about T on the intuitive, semantic content of T and should mirror the inferences based on semantic consequence: if T's semantic content doesn't support an answer ϕ to question Q, then the model shouldn't provide ϕ in response to Q; if T's semantic content supports an answer ϕ to question Q, then the model should provide ϕ in response to Q. Furthermore, if T is altered to T' so that while $T, Q \;||\models\; \psi$, $T', Q \;|\!\not\models\; \psi$, a semantically faithful model should replicate this pattern: $M_T, Q \;||\models\; \psi$, but $M_{T'}, Q \;|\!\not\models\; \psi$. Thus, semantic faithfulness is a normative criterion that tells us how machine learning models of meaning should track human linguistic judgments, when textual input is altered in ways that are relevant to semantic meaning and semantic structure.

Linguistic meaning and the semantic consequence relation $\models$ are defined recursively over semantic structure. Thus, semantic faithfulness provides an important window into machine learning models' grasp of semantic structure and its exploitation during inference. If a model is not semantically faithful, then it doesn't respect semantic entailment. This in turn means that the model is not capturing correctly at least some aspects of semantic structure. Semantic structure includes predicate argument structure (i.e. which object described in T has which property) but also defines the scope of operators like negation over other components in the structure. Semantic structure also links semantic faithfulness with inference, as we exploit that structure to define valid inference. The lack of semantic structure can cause the model to perform invalid inferences.

2.2 A remark on language models and formal semantics

Semantic faithfulness makes use of a traditional notion of semantic consequence, which itself seems divorced *a priori* from the distributional view of semantics present in language models. However, the two are not far apart. In fact they are complementary. (Asher 2011) argues for a complementary level of type-theoretic meaning that roughly corresponds to distributional semantics. In a similar vein, (Fernando 2004) provides a semantics of temporal expressions. But the relation between the distributional view and that of formal semantics is more than complementary. Inspired *inter alia* by (Reynolds 1974), (Asher, Paul, and Venant 2017) provides a model of language in terms of a space of finite and infinite strings. Many of these strings are just jumbles of words but the set also includes coherent and consistent strings that form meaningful texts and conversations. This subset of coherent and consistent texts and conversations allows us to define the semantics and strategic consequences of a conversational move in terms of its possible continuations. LMs find their place rather naturally (Fernando 2022). As LMs are transformer based, trained language models, they provide a probability distribution over possible continuations. Thus, they are sensitive to and can predict possible continuations of a given text or discourse.

When defined over the appropriate set of strings, a semantic consequence relation for continuation semantics subsumes $\models$, as defined in denotational, truth conditional semantics under certain mild assumptions (Reynolds 1974). (De Groote 2006; Asher and Pogodalla 2010) extend this result to modern, so called "dynamic" formal semantics for texts and conversations (Kamp and Reyle 1993; Asher 1993). Thus, continuation semantics is at least with respect to semantic consequence a natural and non-conservative extension of truth conditional semantics,

Continuation semantics also provides a more notion of meaning that is more refined than that provided by truth conditional semantics. Consider, for instance, the set of most probable continuations for (1-a). They are not the most probable continuations for (1-b).

(1) a. If we do this, 2/3 of the population will be saved.
b. If we do this, 1/3 of the population will die.

(1-a)'s most likely continuations would focus perhaps on implementation of the plan; (1-b)'s most likely continuations would focus on how to mitigate the effect of the action or to search for other alternatives. Thus, while (1-a) and (1-b) are semantically equivalent with respect to denotational, truth conditional semantics, they do not generate the same probability distribution over possible continuations and so have arguably distinct

meanings in a continuation semantics that takes the future evolution of a conversation or text into account. Thus in principle continuation semantics as practiced by LMs can in principle capture a finer grained semantics than denotational truth conditional semantics, as well as pragmatic and strategic elements of language.

For an LM to be semantically faithful and to produce coherent texts and conversations, it must learn the right probability distribution over the right set of strings. That is, it has to distinguish sense from nonsense, and it has then to recognize inconsistent from consistent strings, incoherent from coherent ones. If it does so, then the LM will have mastered both $\models$ and the more difficult to capture notion of semantic coherence that underlies well-formed texts and conversations. If it does not do so, it will not be semantically faithful. The fact that continuation semantics subsumes the logical consequence relation $\models$ of formal and discourse semantics reinforces our contention that semantic faithfulness based on such a semantics should be a necessary constraint for adequate meanings based on continuations or meanings based on distributions. Semantic faithfulness is not only a test for an adequate notion of meaning but it also offers a road towards training LMs to better reflect an intuitive notion of semantic consequence and coherence.

In designing experiments to test semantic faithfulness and LM model inference then, we need to pay attention to continuation semantics and how possible interventions can affect discourse continuations. Our interventions exploit the and the semantics of continuations. We need to do this, because LMs are sensitive to continuations and can detect low probablity continuations. Simple insertions of materials to affect semantic content threaten to not end up testing the inferences we want to test but rather signal an LM's sensitivity to low probability continuations. Continuation semantics provides a rationale for human in the loop constructions of interventions that respect or shift semantic content and continuations (Kaushik, Hovy, and Lipton 2019; Gardner et al. 2020).

2.3 A summary of our contributions

We show that transformer representations of meanings are not semantically faithful, and this calls into question their grasp of semantic structure. We detail three types of experiments in which we show large language models fail to be semantically faithful: in the first case, transformers "hallucinate" responses to questions about texts that are not grounded in their semantic content; in the second, models fail to observe modifications of a text that renders it inconsistent with the model's answer to a question about the original text; in the third, we show that models don't reliably capture predicate argument structure. These are serious problems, as it means that we cannot offer guarantees that these sophisticated text understanding systems capture basic textual, semantic content. Hence, simple semantic inferences cannot be fully trusted. We analyze the reasons for this and suggest some ways of remedying this defect.

To investigate semantic faithfulness of a model M, we look at inferences M must perform to answer a question, given a story or conversation T and a conversation history containing other questions. Table 1 shows an example. We look at question answering before and after performing two new operations, *deletion intervention* and *negation intervention*, that affect the semantic content of T.

Deletion intervention removes from T a text span conveying semantic information necessary and sufficient given T for answering a question with answer ψ. We call the text conveying the targeted semantic information the *rationale*. T itself supports ψ as an answer to Q—in the formalism of equation 1, $T, Q \mid\models \psi$. But post intervention T, call

it $d(T)$, does not: $d(T), Q \not\models \psi$. The semantic content of $d(T)$ no longer semantically supports ψ. A semantically faithful model M_T should mirror this shift: $M_T, Q \models \psi$ but $M_{d(T)} \not\models \psi$—which accords with human practice and intuition.

Negation intervention modifies a text T into a text $n(T)$ such that $n(T)$ is inconsistent with ψ, where ψ was an answer to a question supported by the original text. In the formal terms we have used to defined semantic faithfulness, $T, Q \models \psi$ but $n(T), Q \models \neg\psi$. One simple instance of negation intervention would insert a negation with scope over the Q targeted semantic information. But this is not the only or even the primary way; in fact such simple cases of negation intervention amount to only 10% of our interventions. To preserve S's discourse coherence and style, changing the content of a text so as to flip the answer in a yes/no questions typically requires other changes to S. To consider a simple example, suppose that in Table 1, we consider as our question Q: *was Cotton white?* Performing negation intervention on the rationale, *there lived a little white kitten named Cotton*, led us to replace the rationale with two sentences: *there lived a little kitten named Cotton. Cotton was not white.*

In general, negation intervention tests whether an ML model is sensitive to semantic consistency. A semantically faithful model should no longer answer Q with *yes* post negation intervention. Once again negation intervention exploits the notion of semantic faithfulness. We should observe a shift in the ML model's behavior after negation intervention on a text T, $n(T)$: supposing that on $T, Q \models \psi$, a semantically faithful model M_T should be such that $M_T, Q \models \psi$ but $M_{n(T)} \not\models \psi$.

Deletion and negation interventions allow us to study the models' behavior in a counterfactual scenario. Such counterfactual scenarios are crucial to understanding the causal efficacy of the rationale in the models' inferring of the ground truth answer for a given question (Schölkopf 2019; Kusner et al. 2017; Asher, Paul, and Russell 2021). Scientific experiments establish or refute causal links between A and B by seeing what happens when A holds and what happens when $\neg A$ holds. Generally, A causes B only if both A and B hold and the counterfactual claim, that if $\neg A$ were true then $\neg B$ would also be true, also holds. So if we can show that a model M_T is such that $M_T, Q \models \psi$ and $T, Q \models \psi$ but also such that $M_{\mathbb{B}(T)}, Q \models \psi$ and $i(T), Q \not\models \psi$—i.e., $i(T)$ no longer contains information α (originally in T) that linguistically supports ψ as an answer to Q—then we have shown that α is not causally involved in the inference to ψ.

We perform our experiments on CoQA (Reddy, Chen, and Manning 2019), a conversational question answering data set. The CoQA data set includes for each question an annotated rationale that human annotators determined to provide the ground truth answers to questions and from which ideally the answer should be computed in a text understanding system. The **bold text** in Table 1 is an example of a rationale. We exploited these annotated rationales to study the language models' behavior under our semantic interventions. More precisely, we ask the following question: *Do language models predict the ground truth answer even when the rationale is removed from the story under deletion intervention or negated under negation intervention?*

The surprising answer to our question is "yes"; popular language models based on state of the art transformer architectures continue to predict the same answers after deletion intervention and negation intervention as they did on the original text. Such models are not semantically faithful. Intuitively, a model should not make such a prediction post deletion or post negation intervention, since the content on which the ground truth answer should be computed, i.e. the rationale, is no longer present in the story. Our interventions show that the rationale is not a cause of the model's computing the ground truth answer; at least they are not necessary for computing the answer.

This strongly suggests that such language models are not guaranteed to be semantically faithful, something we establish in greater detail in Sections 5 and 7.

In a third set of experiments in Section 8, we query the models directly for their knowledge of predicate argument structure in texts. We construct sentences with two objects that each have a different property. We then perform two experiments. In the first, simple experiment, we simply query the model about the properties those objects have. In some cases, some models had trouble even with this simple task. In a second set of experiments, we query the model with two distinct but semantically equivalent yes/ no questions. This experiment produces some surprising results where models have trouble answering semantically equivalent questions in the same way, once again indicating a lack of semantic faithfulness. Formally we have:

- two questions, Q, Q',
- $\models Q \leftrightarrow Q'$ and
- $T, Q \mid\!\models \psi$ *iff* $T, Q' \mid\!\models \psi$
- but it's **not** the case that
  $M_T, Q \mid\!\models \psi$ *iff* $M_T, Q' \mid\!\models \psi$.

Working with *base* and *large* variants of three language models, BERT (Devlin et al. 2019), RoBERTa (Liu et al. 2019b), and XLNet (Yang et al. 2019), on the CoQA data set (Reddy, Chen, and Manning 2019), we make the following five contributions:

1. We show that, despite the models' high performance on the CoQA data set, they wrongly predict the ground truth answer post deletion intervention for a large number of cases ($\sim 50\%$).
2. We show that a simple intervention-based training strategy is extremely effective in making these models sensitive to deletion intervention without sacrificing high performance on the original data set.
3. We quantitatively analyze the *inner-workings* of these models by comparing the embeddings of common words under the two training strategies. We find that under intervention based training, the embeddings are more contextualized with regards to the rationale.
4. For negation intervention, we show that all the models suffer a $\sim 20\%$ drop in accuracy when the textual support is negated in the story.
5. We show that, in general, the models have difficulty in capturing predicate argument structure by examining their behavior on paraphrased questions.
6. We also test the ability of InstructGPT (Ouyang et al. 2022) (i.e. *text-davinci-002* and *text-davinci-003*) to tackle the two interventions and capture predicate-argument structure via prompting. For the two interventions, InstructGPT models also displays similar behavior as the other models. With regards to predicate argument structure, the models achieves very high performance. However, for certain cases, the models do exhibit inconsistent behavior as detailed in Section 9.

3. Related work

There has been a significant amount of research analyzing language models' behavior across different NLP tasks (Rogers, Kovaleva, and Rumshisky 2020). *Probing* has been a popular technique to investigate linguistic structures encoded in the contextualized representations of these models (Pimentel et al. 2020; Hewitt and Liang 2019; Hewitt and Manning 2019; Chi, Hewitt, and Manning 2020). In probing, one trains a model (known as a *probe*) which takes the frozen representations of the language model as input, for a particular linguistic task. The high performance of the probe implies that the contextualized representations have encoded the required linguistic information.

In particular, predicate argument structure has been a subject of probing (Conia and Navigli 2020, 2022). Though most, if not all, of the effort is devoted to finding arguments of verbal predicates denoting actions or events using semantic role labeling formalisms (Chi, Hewitt, and Manning 2020; Conia and Navigli 2020). Little effort to our knowledge has been made in the literature to investigate the grasp of predicate argument structure at the level of the formal semantic translations of natural language text—which includes the arguments of verbal predicates but also things like adjectival modification.

One major disadvantage of probing methods is that they fail to address how this information is used during inference (Tenney, Das, and Pavlick 2019; Rogers, Kovaleva, and Rumshisky 2020). Probing only shows that there are enough clues in the representation so that a probe model can learn to find, say the predicate argument from the language model's representation. It tells us little as to whether the model leverages that implicit information in reasoning about textual content. Our experiments are designed to do the latter.

Another approach to understanding the inner workings of language models studies their behavior at inference time. Elazar et al. (2021) explores an intervention-based model analysis, called *amnesic probing*. Amnesic probing performs interventions on the hidden representations of the model in order to remove specific morphological information. In principle one could extend this approach to other kinds of linguistic information. Amnesic probing is unlike our work, in which the interventions are performed on the input linguistic content and form. Balasubramanian et al. (2020) showed in related work that BERT is *surprisingly brittle* when one named entity is replaced by another. Sun et al. (2020) showed the lack of robustness of BERT to commonly occurring misspellings.

For the task of question answering, a Transformer-based language model with multiple output heads is typically used (Hu et al. 2019). An output head caters to a particular *answer type*. Thus, the usage of multiple output heads allows the model to generate different answer types such as span, yes/no, number, etc. Geva et al. (2021) studied the behavior of *non-target* heads, i.e., output heads not being used for prediction. They showed that, in some cases, non-target heads are able to explain the models' prediction generated by the *target head*. Schuff, Adel, and Vu (2020) analyzed the question answering models which predict answer as well as an explanation. For such models, they manually analyzed the predicted answer and explanation to show that the explanation is often not suitable for the predicted answer. Their methodology is in contrast to our work, since we simply argue that the model uses the rationale for predicting the answer if it is sensitive to *deletion intervention*.

Researchers in prior work have also studied the behavior of the model on manipulated input texts (Balasubramanian et al. 2020; Sun et al. 2020; Jia and Liang 2017; Song et al. 2021; Belinkov and Bisk 2018; Zhang et al. 2020). However, they usually frame the task in an *adversarial scenario* and rely either on an attack algorithm or complex heuristics

for generating manipulated text. The objective in such work is to fool the model with the manipulated text so that the model changes its predictions whereas a human would not change the prediction in the face of the manipulated data.

In contrast, deletion intervention is a simple *content deletion* strategy; it is not designed to get the model to shift its predictions in cases where a human would not. It's not designed to trick or fool ML models. Deletion intervention manipulates the text to test how the deletion of content affects inference; ideally both humans and the ML model should shift their predictions in a similar way given a deletion intervention. Nevertheless, it is also reasonable to expect a model that was successfully attacked in an adversarial setting to be sensitive to deletion intervention.

With respect to negation intervention, researchers have examined the effects of negation and inference at the sentential level on synthetic datasets (Naik et al. 2018; Kassner and Schütze 2020; Hossain et al. 2020; Hosseini et al. 2021). Our aim is more ambitious; we study how transformer models encode both the content C in a text and content C' in a negation-intervened text that is inconsistent with C. Using negation intervention, we test how replacing C with C' affects inference in natural settings. As with deletion intervention, we offer another way of changing the meaning of texts that should make both humans and semantically faithful models change their predictions. There is similar work relevant to negation intervention—on contrast set data and also counterfactual data (Kaushik, Hovy, and Lipton 2019; Gardner et al. 2020). The datasets on which Kaushik, Hovy, and Lipton (2019); Gardner et al. (2020) operate are less complex discursively than the CoQA dataset. The CoQA dataset also allows us to look more closely at what the models are actually sensitive to in a longer text or story. We return to this issue in more detail in Section 7. In general, interventions are an important mechanism to build counterfactual models as Kaushik, Hovy, and Lipton (2019) also argue. These are important for understanding causal structure (Schölkopf 2019; Kusner et al. 2017; Barocas, Hardt, and Narayanan 2019).

4. Specifications of the data set and models

We now describe the CoQA data set and the architecture of the three language models used for this work along with implementation details.

4.1 The CoQA data set

The CoQA data set consists of a set of stories paired with a sequence of questions based on the story. To answer a particular question, the model has access to the story and previous questions with their ground truth answers—this is the conversation history. The data set contains questions of five types: *yes/no* questions, questions whose direct answer is a *number*, alternative or *option* questions (e.g., *do you want tea or coffee?*, questions with an *unknown* answer, and questions whose answer is contained in a *span* of text. The *span* answer type accounts for majority of the questions ($> 75\%$). The data set also contains a *human annotated rationale* for each question. The training set contains $\sim 8K$ stories and $\sim 0.1M$ questions; the development set contains 500 stories and $7,983$ questions. Since the test set is not publicly available, we report the performance of the three models across different experimental settings on the development set.

4.2 Models

We conducted experiments on *base* and *large* variants of three Transformer-based language models— BERT (Devlin et al. 2019), RoBERTa (Liu et al. 2019b) and XLNet (Yang et al. 2019). To predict the answer for the i^{th} question, Q_i, for a given story S, the three models use previous questions along and their ground truth answers from the conversation history. The input for the three models for the story, S, and question, Q_i, is as follows.

$$\begin{aligned}\text{XLNet} : [S \text{ <sep> } Q_{i-2}A_{i-2}Q_{i-1}A_{i-1} \\ Q_i \text{ <sep> } \text{ <cls> }] \\ \text{BERT/RoBERTa} : [\text{ <cls> } Q_{i-2}A_{i-2}Q_{i-1}A_{i-1}Q_i \\ \text{<sep> } S \text{ <sep> }]\end{aligned}$$

where <sep> token is used to demarcate the story and the question history, <cls> is a special token, and A_j denotes the ground truth answer for the question, Q_j . In the rest of the paper, we refer to the string $Q_{i-2}A_{i-2}Q_{i-1}A_{i-1}Q_i$ as *question context*.

We adopted the publicly available XLNet model for this paper[2]. The model contains output heads for *unknown*, *yes*, *no*, *number*, *option*, and *span*. Each output head is fed with a concatenation of the CLS embedding and contextualized embeddings of the story weighted by the predicted start probabilities to predict a score.

For BERT and RoBERTa, we implemented the rationale tagging multi-task model described in Ju et al. (2019). Unlike XLNet, the two models are trained on question answering as well as on the rationale tagging task. Furthermore, for a question, the two models can predict *yes*, *no*, *unknown*, and *span*. As a result, the two models predict span for 78.9% of the questions in the development set, whereas XLNet predicts span for 75.8%. For span prediction, the start and end logits for the answer are predicted by applying a fully connected layer to the contextualized representation of the story obtained from the last layer of the model.

The rationale tagging task requires predicting whether a token $t \in S$ belongs to the rationale. Let $h_t \in \mathbb{R}^d$ denote the contextualized embedding obtained from the last layer for token t. The model assigns a probability p_t for t to be in the rationale as follows.

$$p_t = \sigma(uReLU(Vh_t)) \tag{3}$$

where $u \in \mathbb{R}^{1\times d}$, $V \in \mathbb{R}^{d\times d}$, $ReLU$ is the rectified linear unit activation function, and $\sigma(.)$ denotes the sigmoid function. An attention mechanism is then used to generate a representation, q^L, as shown below.

2 https://github.com/stevezheng23/mrc_tf

Story	Characters: Sandy, Rose, Jane, Justin, Mrs. Lin [...] Jane: Sandy, I called you yesterday. Your mother told me [...] This year is very important to us. **Sandy:(Crying) My father has lost his job**, and we have no money to pay all the spending. [...] *Jane: Eh...I hear that Sandy's father has lost his job*, and Sandy has a part-time job.
Question	Who was unemployed?
Prediction	Sandy's father

Table 2: An example from CoQA data set where the rationale (shown in bold) is not necessary to answer the question. The question can be answered using the italicised text.

$$p'_t = p_t \times h_t \tag{4}$$

$$a_t = softmax(w_1 ReLU(W_2 p'_t)) \tag{5}$$

$$q^L = \sum_t a_t \times p'_t \tag{6}$$

where $w_1 \in \mathbb{R}^{1\times d}$, $W_2 \in \mathbb{R}^{d\times d}$. Let $h_{CLS} \in \mathbb{R}^d$ denote the CLS embedding obtained from the last layer. h_{CLS} is concatenated with the embedding q^L. The concatenated embedding is then used in BERT and RoBERTa to generate a score for *yes*, *no*, and *unknown* respectively.

4.3 Implementation details

We implemented the three language models in PyTorch using the Huggingface library (Wolf et al. 2020). The models were finetuned on the CoQA data set for 1 epoch. The *base* variant of the three models was trained on a single 11 GB GTX 1080 Ti GPU, whereas the *large* variant was trained on a single 24 GB Quadro RTX 6000 GPU. The code for this work along with the additional dataset created as part of studying negation intervention and predicate-argument structure will be made publicly available.

5. Deletion Intervention and Results

In this section, we explain the operation of *deletion intervention*. Deletion intervention is an operation that removes the *rationale* of a question Q from the *story*. For a few instances in the CoQA data set, we found that the annotated *rationale* for Q was not necessary for answering Q, because the sentences following the rationale contained the relevant information for supplying an answer to Q. One such instance is shown in Table 2. In our experiments, we did not find any instance where the sentences preceding the rationale contained the necessary information for answering the question. To avoid problems with such examples containing redundancies, given an original story (OS), we created two additional data sets:

Model	Data set	F1	EM	unk%
BERT-base	OS	76.1	66.3	1.97
	TS	77.2	67.1	2.18
	TS-R	55.6	48.2	1.98
BERT-large	OS	80.7	71.1	2.01
	TS	81.6	72.1	2.32
	TS-R	63.6	57.8	3.79
RoBERTa-base	OS	80.3	70.8	1.95
	TS	80.8	71.1	2.64
	TS-R	55.5	51.1	16.92
RoBERTa-large	OS	87.0	77.7	1.74
	TS	86.8	77.3	2.72
	TS-R	59.9	55.7	22.36
XLNet-base	OS	82.5	74.8	1.08
	TS	82.1	74.2	1.11
	TS-R	53.5	48.0	14.0
XLNet-large	OS	86.3	78.9	0.86
	TS	85.6	78.5	2.58
	TS-R	48.1	44.3	31.68

Table 3: EM, and F1 score of the models when trained solely on the original story (OT training strategy).

1. TS: In this data set, we truncate the original story (OS) so that the statement containing the rationale is the last statement. We refer to this data set as TS (short for *truncated story*). The stories in TS do not reduplicate elsewhere information in the rationale.
2. TS-R: Given TS, we perform *deletion intervention* by removing all the sentences containing the rationale. The cases where the rationale begins from the first sentence itself are discarded. For questions where the model predicts a *span*, we add the ground truth answer (if not already present) post deletion intervention. This is necessary since for the *span* type questions, the model can only predict the ground truth answer if it is present in the story. As an example, consider the question "Where does Alan go after work?" and the story "Alan works in an office. **He goes to a nearby park after work.**" (rationale shown in bold). In this case, TS-R will be "Alan works in an office. park." Since TS-R doesn't contain the information necessary for answering the question, the model should predict *unknown* for such instances.

We trained the models on the OS data set and evaluated them on the three aforementioned data sets. We refer to this training strategy as OT (short for original training). Table 3 shows EM (exact match), F1, and the percentage of unknown predictions (unk $\%$) of the models on the three data sets. As we can see from the table, for the data sets OS and TS, the performance of all the models is pretty similar. The performance drops for TS-R which shows some sensitivity to *deletion intervention*. However, all the models still achieve an EM of $\sim 50\%$ which is intuitively way too high for a semantically faithful model. We believe this shows that the models rely on superficial cues for predicting the answer; for example, in the presence of a question like "What color was X?" it searches for a color word not too far from a mention of X. We also find that, for TS-R, the unk $\%$ for RoBERTa, XLNet is significantly higher than BERT.

Model	Data set	F1	EM	unk%
BERT-base	OS	76.4	67.2	3.82
	TS	77.7	68.0	7.93
	TS-R	5.7	5.4	93.08
BERT-large	OS	78.8	69.8	4.20
	TS	80.1	70.7	7.34
	TS-R	5.4	5.1	94.25
RoBERTa-base	OS	81.2	71.6	2.86
	TS	81.9	72.0	5.20
	TS-R	5.5	5.3	94.25
RoBERTa-large	OS	86.2	76.9	2.66
	TS	86.3	76.7	4.01
	TS-R	5.1	5.0	95.34
XLNet-base	OS	81.3	74.2	4.63
	TS	79.6	72.4	10.87
	TS-R	6.6	6.4	93.86
XLNet-large	OS	83.1	75.8	5.10
	TS	81.0	74.1	10.69
	TS-R	5.6	5.5	95.42

Table 4: EM, and F1 score of the models under intervention-based training (IBT training strategy).

Model	Strategy	F1	EM
BERT-base	OT	38.0	31.5
	IBT	42.7	38.1
BERT-large	OT	40.9	34.9
	IBT	47.4	42.5
RoBERTa-base	OT	41.1	35.1
	IBT	45.9	40.9
RoBERTa-large	OT	46.4	40.7
	IBT	55.4	50.5
XLNet-base	OT	43.7	38.7
	IBT	52.8	48.6
XLNet-large	OT	48.3	43.7
	IBT	59.2	55.1

Table 5: Off the shelf performance (EM, and F1 score) of the models on SQUAD dataset. IBT performs significantly better than OT as this dataset contains $\sim 50\%$ unanswerable questions.

6. Intervention-based Training

To enhance the sensitivity of the language models on TS-R, we propose a simple *intervention-based training* (IBT). In this training strategy, we train the model on the three data sets simultaneously. For OS and TS, the model is trained to predict the ground truth answer, whereas, for TS-R, the model is trained to predict *unknown*. Note that the models are trained for same number of epochs under both the training strategies.

Table 4 shows the performance of the models on the three data sets. First, we observe that for the data sets OS and TS, the training strategy IBT is at par with the strategy OT. From the table, we can see that the performance of the models drops significantly on TS-R. Furthermore, all the models have very high unk $\%$ ($> 90\%$) on TS-R. Thus, IBT is able to make the models highly sensitive to *deletion intervention*. We also found that, for span type questions in TS-R, when the ground truth answer is not added

at the end, the models trained under IBT still have a higher unk % ($> 70\%$) compared to models trained under OT where the unk % varies from $\sim 20\%$ to $\sim 45\%$. Hence, the models trained under IBT do not solely rely on this cue to predict *unknown*.

To further substantiate this claim, we look at off-the-shelf performance of the models trained under both strategies on SQUAD dataset (Rajpurkar, Jia, and Liang 2018). Table 5 shows the off-the-shelf performance on *SQUAD development set* of all the models under the two training strategies. From the table, we can see that IBT performs significantly better than OT. This is because SQUAD contains $\sim 50\%$ unanswerable questions. For such questions, the models trained under IBT predict *unknown* more often than their OT counterpart.

6.1 In-depth Analysis of Intervention-based Training

In this section, we study the inner-workings of these models in order to explain the effectiveness of intervention-based training against deletion intervention. As mentioned in § 4.2, CLS embedding plays a crucial role in predicting an answer to a particular question. Hence, to begin with, we look at the cosine similarity (*cossim*) between CLS embeddings of OS and TS under the two training strategies (OT and IBT). Similarly, we also look at tge cossim between CLS embeddings of OS and TS-R under the two training strategies (OT and IBT). Figures 1 and 2 show the histogram of cossim on the development set for RoBERTa-large. In Figure 1, we see that the two histograms follow a similar pattern. The cossim is very high for almost all the cases. This is interesting since it shows that even if a significant chunk is removed from the story, it doesn't affect the CLS embedding in any meaningful way. However, in Figure 2, there is a drastic difference between the two histograms. Whereas the histogram for the OT strategy still follows a similar pattern as before, the histogram for IBT shows a significant drop in cossim. This shows that, for most of the cases under IBT, the CLS embedding is heavily affected once the rationale is removed from the story.

This effect is not only local to the CLS token but rather is observed for all the input tokens. To show this, we look at the cosine similarity of common tokens of OS and TS under the two training strategies, and similarly, the cosine similarity of common tokens of OS and TS-R under the two training strategies. Figures 3 and 4 show the corresponding histogram for RoBERTa-large. Here also, we can see that the cosine similarity of common tokens in OS and TS is very high for both training strategies. Once again, the model's representation of the common words doesn't seem to be affected by the removal of large parts of the textual context; this indicates either that the model finds the larger context irrelevant to the task or it might not be capable of encoding long distance contextual information for this task.

For common tokens in OS and TS-R, however, there is a stark contrast between the two training strategies. For OT, the cosine similarity of common words still remain high but for IBT, the cosine similarity drops by a large margin. This shows that, under IBT, the embeddings of the input tokens are more contextualized with respect to the rationale. Due to this, under IBT, the word embeddings get significantly altered once the rationale is removed from the story. Similar to RoBERTa-large, other models also exhibit similar pattern of cosine similarity for CLS and common tokens, as shown in Tables 6 and 7. From Table 6, we can see that, for all the models, cossim($h^{OS}_{CLS}, h^{TS-R}_{CLS}$) for IBT is much lower than the corresponding cosine similarity for OT; whereas cossim($h^{OS}_{CLS}, h^{TS}_{CLS}$) is similar for both the strategies. Similarly, Table 7 shows that, for all the models,

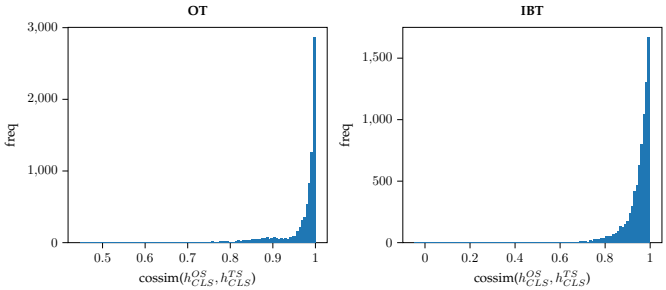

Figure 1: RoBERTa-large: Histogram plot of cosine similarity between CLS embedding for OS and TS under two training strategies (OT on left and IBT on right).

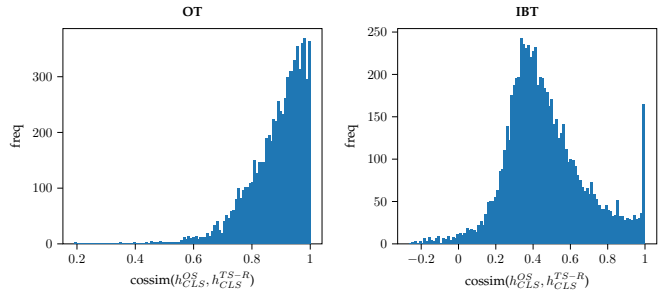

Figure 2: RoBERTa-large: Histogram plot of cosine similarity between CLS embedding for OS and TS-R for the two training strategies (OT on left and IBT on right).

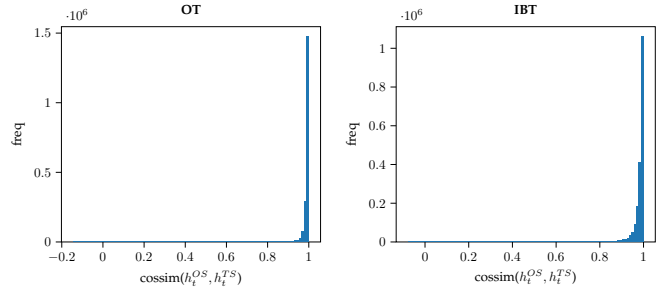

Figure 3: RoBERTa-large: Histogram plot of cosine similarity between common tokens of OS and TS for the two training strategies (OT on left and IBT on right).

cossim(h_t^{OS}, h_t^{TS-R}) for IBT is much lower than the corresponding cosine similarity for OT; whereas cossim(h_t^{OS}, h_t^{TS}) is similar for both the strategies.

From a more conceptual perspective, the sensitivity to the rationale in IBT suggests that IBT is providing the kind of instances needed to confirm the counterfactual, *were the rationale not present, the model would not answer as it does when the rationale is present*. Thus, at a macro level, attention based models can locate spans of text crucial to determining semantic content through particular forms of training.

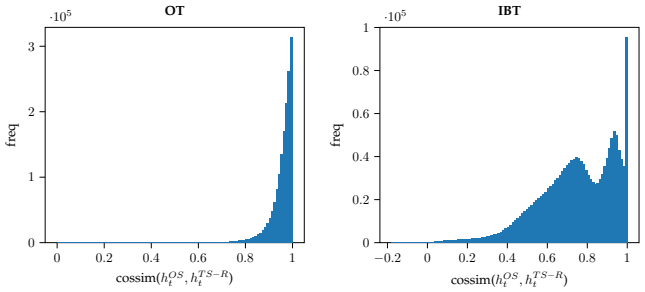

Figure 4: RoBERTa-large: Histogram plot of cosine similarity between common tokens of OS and TS-R under two training strategies (OT on left and IBT on right).

Model	OT		IBT	
	cossim($h_{CLS}^{OS}, h_{CLS}^{TS}$)	cossim($h_{CLS}^{OS}, h_{CLS}^{TS-R}$)	cossim($h_{CLS}^{OS}, h_{CLS}^{TS}$)	cossim($h_{CLS}^{OS}, h_{CLS}^{TS-R}$)
BERT-base	0.99 ± 0.02	0.97 ± 0.03	0.96 ± 0.07	0.33 ± 0.34
BERT-large	0.99 ± 0.02	0.98 ± 0.04	0.95 ± 0.12	0.42 ± 0.31
RoBERTa-base	0.96 ± 0.04	0.92 ± 0.08	0.94 ± 0.06	0.55 ± 0.22
RoBERTa-large	0.97 ± 0.06	0.88 ± 0.10	0.95 ± 0.07	0.47 ± 0.21
XLNet-base	0.99 ± 0.03	0.95 ± 0.09	0.97 ± 0.06	0.27 ± 0.33
XLNet-large	0.98 ± 0.04	0.89 ± 0.15	0.98 ± 0.05	0.53 ± 0.21

Table 6: Cosine similarity (mean$\pm$ std) of CLS embeddings for the two training strategies.

Model	OT		IBT	
	cossim(h_t^{OS}, h_t^{TS})	cossim(h_t^{OS}, h_t^{TS-R})	cossim(h_t^{OS}, h_t^{TS})	cossim(h_t^{OS}, h_t^{TS-R})
BERT-base	0.99 ± 0.04	0.94 ± 0.07	0.96 ± 0.06	0.66 ± 0.22
BERT-large	0.99 ± 0.04	0.94 ± 0.06	0.98 ± 0.04	0.71 ± 0.21
RoBERTa-base	0.99 ± 0.04	0.95 ± 0.06	0.97 ± 0.05	0.75 ± 0.20
RoBERTa-large	0.99 ± 0.03	0.95 ± 0.06	0.98 ± 0.04	0.74 ± 0.19
XLNet-base	0.96 ± 0.08	0.90 ± 0.13	0.96 ± 0.08	0.57 ± 0.40
XLNet-large	0.94 ± 0.14	0.86 ± 0.24	0.94 ± 0.15	0.52 ± 0.44

Table 7: Cosine similarity (mean$\pm$ std) of common tokens for the two training strategies.

7. Negation Intervention

In this section, we detail our experiments on negation intervention. Negation intervention investigates possible causal dependencies of a model's inferences based on logical structure, in particular the scope of negation operators. As we said in Section 2, the idea behind negation intervention is to alter a text with an intervention n such that $T, Q \models \psi$ iff$n(T), Q \models \neg\psi$.

For negation intervention, we randomly sampled 275 yes-no questions. We appropriately modified the rationale in the truncated story (i.e., TS) for these samples in order to switch the answer from yes to no and vice-versa. Table 8 shows the effect of negation intervention on the models. In the table, Org-Acc refers to accuracy of the model on the original sample, Mod-Acc refers to accuracy of the model post negation intervention (i.e., with respect to the modified ground truth answer), and Comb-Acc refers to the percentage of cases where the model answered correctly for both original and modified

Model	Org-Acc	Mod-Acc	Comb-Acc
BERT-base	78.2	58.9	41.5
BERT-large	84.7	65.1	52.0
RoBERTa-base	81.8	61.8	47.6
RoBERTa-large	94.2	72.7	67.3
XLNet-base	85.1	64.7	52.0
XLNet-large	90.2	68.7	59.6

Table 8: Effect of Negation intervention on different models.

sample. Table 8 shows a $\sim 20\%$ drop in accuracy for all the models when we compare Org-Acc and Mod-Acc. This significant drop highlights the inability of the models to handle negation intervention. The low Comb-Acc scores of the models further highlight this fact. Switching to the IBT regime provided no significant difference. This indicates that another type of training will be needed for these models to take into systematic account the semantic contributions of negation.

A natural option is to train over negated examples and non negated examples. Kassner and Schütze (2020) performs such an experiment and concludes that transformer style models do not learn the meaning of negation. And Hosseini et al. (2021) provides a particular training regime that seems to improve language models' performance on the data set of negated examples introduced by Kassner and Schütze (2020).

Nevertheless, while Kassner and Schütze (2020)'s conclusion is compatible with our findings, we are not sanguine that Hosseini et al. (2021)'s training regime will improve model performance on the operation of negation intervention. The interventions we needed to make to induce the appropriate shifts in answers often depended on quite important shifts in material. Simple insertions of negation often seemed to us to disrupt the coherence and flow of the text; these disruptions could provide superficial clues for shifting the model's behavior in a task. To give an example, here is a rationale from one of the stories in the CoQA dataset:

(2) A law enforcement official told The Dallas Morning News that a door was apparently kicked in

Given the question, *Was the house broken into?* (the original answer was *yes*), we changed the rationale to:

(3) A law enforcement official told The Dallas Morning News that a door was open, leaving the possibility that the killers had been invited in

to get a negative answer to the question.

In this intervention, we didn't insert a negation but rather changed the wording to get a text inconsistent with the original answer. More generally, in only 72 out of 275 cases of Negation Intervention (26 %), we added or removed "no/not". And within these 72 cases, only around 25 cases featured the simple addition/removal of "no/not"—e.g. the replacement of *six corn plants* to *but no corn plants*. In the rest of the cases, although we added/removed "no/not", we made more substantive changes to the story. Here is one example, where the question was, *did the wife console the boy?* The original rationale was as follows:

(4) "Robert Meyers said his wife tried to help Nowsch. "My wife spent countless hours at that park consoling this boy," he said.

We changed this to:

(5) Robert Meyers said his wife did not try to help Nowsch. "My wife spent countless hours at that park tormenting this boy," he said.

For the other 203 out of 275 cases (74%), there were lexical changes and substantial changes to the rationale to preserve stylistic consistency.

Thus the simple addition/removal of "no/not" cases numbered around 25 cases ($\sim 10\%$). In general, the models were able to switch their answers on such cases. Out of 73 cases, where Roberta-large answered the question for the original story correctly and didn't switch the answer post negation intervention, there are only 6 trivial cases.

The inferences involved in negation intervention are thus quite complex and go beyond the recognition of a simple negation. A mastery of such inferences would indicate a mastery not only of negation but of inconsistency, which would be a considerable achievement for a machine learned model. So simply alerting the model to the presence of negation will not suffice to guarantee reasoning ability with negation.

The alternative is to create a corpus with many more negation intervention examples. We remark that it was difficult to construct the requisite data so as to meet our view of negation intervention. We did this for almost 300 examples, but we would need a lot more examples for fine tuning.

8. Predicate Argument Structure

In this section, we study whether the models stay faithful to simple cases of predicate argument structure. As we already mentioned in the introduction, we propose two types of experiments. In the first simple experiment, we ask a question Q about the properties of objects in a text T. Given an answer ψ such that $T, Q \mid\models \psi$, we expect that for a semantically faithful model M_T that $M_T, Q \mid\models \psi$.

The second set of experiments is more involved. Formally, it involves the following set up. Given:

- two questions, Q, Q',
- $T \models Q \leftrightarrow Q'$

we should predict

$$T, Q \mid\models \psi \textit{ iff } T, Q' \mid\models \psi$$

To test for semantic faithfulness in these contexts, we devised synthetic, textual data for these experiments. We used five different schemas:

1. The *col1* car was standing in front of a *col2* house.
2. They played with a *col1* ball and *col2* bat.
3. The man was wearing a *col1* shirt and a *col2* jacket.
4. The house had a *col1* window and a *col2* door.
5. A *col1* glass was placed on a *col2* table.

Model	Org-Acc	Mod-Acc
BERT-base	50.0 (100.0)	69.4 (59.8)
BERT-large	95.2 (51.0)	77.3 (27.3)
RoBERTa-base	51.0 (99.0)	70.0 (78.5)
RoBERTa-large	99.4 (49.4)	95.0 (45.0)
XLNet-base	50.6 (6.0)	50.8 (0.7)
XLNet-large	75.2 (74.8)	79.8 (36.3)

Table 9: Effect of question paraphrasing on different models. Org-Acc, and Mod-Acc denote accuracy on original and modified paraphrased question respectively. The number in bracket denote percentage of cases where the model predicted "no" as the answer.

where *col1* and *col2* denote two distinct colors. Using these 5 schemas and different color combinations, we constructed a dataset of 130 stories. For each story, we have 4 questions. (2 "yes" and 2 "no" questions). As an example, for the story, "The blue car was standing in front of a red house.", the 2 "yes" questions are "Was the car blue?" and "Was the house red?"; and 2 "no" questions are "Was the car red?" and "Was the house blue?". Thus, we have a total of 520 questions. The Org-Acc in Table 9 shows the accuracy of the models on these questions and indicates a huge variance in accuracy across the models. We observed that BERT-base predicted *no* for all questions, and RoBERTa-base predicted *no* for most of the questions, while XLNet-base mostly predicted "yes". For the large models, RoBERTa-large and BERT-large achieved very high accuracies. We note, however, that this dataset is very simple.

These results indicate that the small models really didn't do much better than chance in answering our yes/ no questions; hence either they didn't capture of the predicate argument structure of the sentences, or they could not use that information to reason to an answer to our questions. They failed on the most basic level. The large models fared much better, but this in itself didn't suffice to determine a causal link between the predicate argument information and the inference to the answers.

Probing further, we then examined how the models fared under semantically equivalent questions. Q' is semantically the same as ($\equiv$) Q given context C iff they have the same answer sets in C (Bennett 1979; Karttunen 1977; Groenendijk 2003). In our situation, the context is given by the text T. Thus, we have $T \models Q \leftrightarrow Q'$ and $T, Q \mid\models \psi$ *iff* $T, Q' \mid\models \psi$. If M_T is semantically faithful and $T \models Q \leftrightarrow Q'$, then we should have $M_T, Q \mid\models \psi$ iff $M_T, Q' \mid\models \psi$. To construct semantically equivalent questions, we paraphrased the initial question in our data set, i.e., *"Was the car red?"* to *"Was there a red car?"*. This resulted in new set of 520 questions for the 130 stories. The Mod-Acc in Table 9 shows the accuracy of the models on the modified questions. Apart from XLNet-base which predicted "yes" for most of the modified questions and RoBERTa-large which retains it high accuracy, all the other models behave very differently from before. The accuracy of BERT-large drops drastically on these very simple questions, while BERT-base and RoBERTa-base perform significantly better on modified questions as they no longer mostly predict "no". For XLNet-large, while the two accuracies are similar, the model goes from predicting mostly "no" to mostly "yes". This contrast in behavior, as shown in Table 9, indicates that these models are unstable and lack semantic faithfulness on this task. There are two possible explanations; either the semantic structure of the two questions is not exploited in inferring an answer or the predicate argument structure of the text is not exploited in the context of one of the questions.

To further investigate our models' behavior with respect to questions, we tested their ability to handle semantically equivalent questions using CoQAR dataset (Brabant,

Model	EM/F1	Match%	IoU
BERT-base	61.8/72.0	54.9	0.16
BERT-large	67.2/77.1	61.5	0.18
RoBERTa-base	66.6/76.7	63.1	0.19
RoBERTa-large	74.2/84.4	72.9	0.26
XLNet-base	70.0/78.1	64.2	0.14
XLNet-large	75.0/83.4	70.3	0.19

Table 10: EM, and F1 score of the models on CoQAR dataset along with Match% and IoU.

Model	EM	F1
text-davinci-002	46.4	58.5
text-davinci-003	28.0	45.6

Table 11: EM and F1 score for the two InstructGPT models on TS-R dataset.

Lecorvé, and Rojas Barahona 2022). This dataset modified the original CoQA dataset by paraphrasing each question in CoQA to three semantically equivalent questions. While the questions in CoQAR are not *conversational* in nature, we found that our models performed better when the question context was provided as input. Table 10 shows the performance of the models on the CoQAR development set. In the table, $Match\%$ denotes the percentage of cases where the model gave the exact same answer for all the three variants of the question. For the unsuccessful cases (i.e. where the model gave different answers for the three question variants), we also report IoU which denotes the number of common tokens divided by the total number of unique tokens in the three predicted answers. The table shows that, in significant number of cases, the models do not give the exact same answer for the three question variants as indicated by the low $Match\%$. Moreover, for such unsuccessful cases, the three answers vary significantly as highlighted by low IoU scores.

9. Analyzing InstructGPT via prompting

In this section, we report on the behavior of InstructGPT (Ouyang et al. 2022) using OpenAI API on our two interventions, namely deletion and negation intervention, and our dataset for predicate argument structure. We looked at two InstructGPT models: *text-davinci-002* and *text-davinci-003*. For the two interventions, we proceeded similarly to our approach with the other models: we provided the story, the question context (i.e. two previous question with their ground truth answer), and the current question as input prompt. For predicate argument experiment, we only provided the story and current question in our input prompt, as shown below:

```
The man was wearing a blue shirt and a red jacket.

Q: Was the jacket red?
A:
```

For deletion intervention, we calculated the EM and F1 score for the two models on TS-R dataset. Table 11 shows these scores. As we can see from the table, *text-davinci-002* achieves very high scores; for nearly $\sim 50\%$ of cases the model continues to predict the ground truth answer post deletion intervention. This pathological behavior is similar to other models studied in this work. While *text-davinci-003* does significantly better,

Model	Org-Acc	Mod-Acc	Comb-Acc
text-davinci-002	88.0	61.1	53.5
text-davinci-003	94.2	61.5	56.7

Table 12: Effect of Negation intervention on InstructGPT.

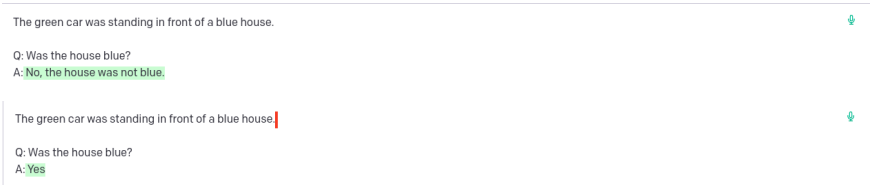

Figure 5: The *text-davinci-002* model predicts correctly when an extra space (shown in red) is added.

the scores are still on the higher side as the model is predicting ground truth answer for $\sim 30\%$ of the cases. Overall, these result showcase that InstructGPT models do not respond appropriately to deletion intervention.

For negation intervention and similarly to Section 7, we looked at Org-Acc, Mod-Acc, and Comb-Acc. Table 12 shows these results. From the table, we can see that both the models suffer a significant drop in accuracy for negated questions in comparison with original questions. Thus, similar to other models, InstructGPT fails to respond well to negation intervention.

For predicate argument experiment, on the overall dataset of 1040 questions, *text-davinci-002* achieved an accuracy of 96.7% (i.e. total of 34 failure cases) with Org-Acc of 99.6% and Mod-Acc of 93.8%. Interestingly, all the 34 failure cases in the original predicate argument data set were "yes" questions. For such cases, in many instances, we observed that adding an extra space to the prompt reverses the model's prediction. One such example is shown in Figure 5. As for *text-davinci-003*, the model achieved perfect accuracy. Unlike *text-davinci-002*, we found that *text-davinci-003* is stable in its prediction with regards to extra spaces in the prompt. However, there were 14 cases where *text-davinci-003* predicted "not necessarily" instead of "no". The question in all these cases was of the form "Was there a *col2* car?" Note that we had 26 cases with this question format and the model predicted "no" for the remaining 12 cases. This showcases instability in model's prediction for two very similar input prompts.

We also tested the model's sensitivity to a contrastive set of examples in which we inserted negations in our predicate argument data set sentences (e.g. "The blue car was standing in front of a house that was not red." for the question "Was the house red?"). In contrast to its performance on negation intervention, the InstructGPT models achieved perfect accuracy on such negated examples. This further demonstrates that negation intervention is different from the tasks given by Naik et al. (2018); Kassner and Schütze (2020); Hossain et al. (2020); Hosseini et al. (2021).

10. Discussion

In conclusion, concerning our findings about predicate argument structure and logical structure more generally, we address three points.

1) Larger Transformer-based models have shown to generally perform better than their smaller variants (Roberts, Raffel, and Shazeer 2020; Liu et al. 2019a). However, some exceptions to this trend have also been observed (Zhong et al. 2021). Our experiments show that with respect to the notion of semantic faithfulness, in general sensitivity to semantic structure and content, larger models fare better in predicate-argument experiments but not in our negation and deletion intervention experiments. For deletion intervention, they are mostly worse-off than smaller models. Section 9 shows that InstructGPT also fails to tackle the two interventions in an efficient manner.

2) Is prompting really superior to fine tuning? Our prompting experiment with InstructGPT allowed us to get results without fine-tuning. This is essentially zero shot learning since no input-output pairs are provided in the prompt. However, for deletion and negation intervention, we observed that InstructGPT models do not present an advancement over other Transformer-based models with respect to behavior post these interventions. Moreover, like Jiang et al. (2020); Liu et al. (2021); Shin et al. (2021), we have found *text-davinci-002* to be extremely sensitive to what should be and intuitively is irrelevant information in the prompt. With regard to semantic faithfulness on predicate argument structure, this shows an astonishing lack of robustness to totally irrelevant material, even if *text-davinci-002* scores very well on this data set. This brittleness is telling; a semantically faithful model that exploits semantic structure to answer questions about which objects have which properties should not be sensitive to formatting changes in the prompt. This indicates to us that even if predicate argument structure questions are answered correctly, *text-davinci-002* is not using that information as it should. *text-davinci-003* is stable to such insignificant changes in the prompt. However, the model still shows instability in its predictions for two very similar prompts as highlighted earlier. Once again, we have our doubts that the right information, i.e. semantic structure, is being leveraged for the answer; if it were, *text-davinci-003* would answer in the same way for all the questions with "no" answers.

3) Extending semantic faithfulness beyond the question answering tasks in NLP. The definition of semantic faithfulness in Section 2 is geared to testing the semantic knowledge of LMs in question answering tasks. Question answering can take many forms and is a natural way to investigate many forms of inference or exploitations of semantic and logical structure. It also underlies many real-world NLP applications, like chatbots, virtual assistants and web searches (Liang et al. 2022). Semantic faithfulness can be extended to probe for a model's inferences concerning artificial languages like first order logic or any other formal language or programming language for which there is a well defined notion of semantic consequence ($\models$). In such cases, the role of the "text" in semantic faithfulness would be played by a set of premises, a logical or mathematical theory, or code for an algorithm or procedure. Similar experiments of deletion or negation intervention could in principle be performed in these settings, which opens up a to us novel way of investigating LM models' performance on tasks like code generation (Chen et al. 2021; Sarsa et al. 2022). Alternatively, as suggested by Shin and Van Durme (2021), exploiting formal logical forms may help with semantic faithfulness.

11. Conclusion and Future Work

We have studied the semantic faithfulness of popular Transformer-based language models for two input intervention strategies, deletion intervention, and negation intervention, and with respect to their responses to simple, semantically equivalent questions. Despite high performance on the original CoQA data set, the models exhibited

very low sensitivity to deletion intervention and suffered a significant drop in accuracy for negation intervention. They also exhibited unreliable and unstable behavior with respect to semantically equivalent questions (Q≡). Our simple intervention-based training (IBT) strategy made the contextualized embeddings more sensitive to the rationale and corrected the models' erroneous reasoning in the case of deletion intervention.

Our paper has exposed flaws in popular language models. In general, we have shown that even large models are not guaranteed to respect semantic faithfulness. This likely indicates that the models rely on superficial cues for answering questions about a given input text. While IBT is successful at remedying models' lack of attention to logical structure in cases of deletion intervention, it doesn't generalize well to the other experimental setups we have discussed. We do not have easy fixes for negation interventions or for the inferences involving predicate argument structure. This is because it is difficult to generate enough data through negation intervention to retrain the model in the way we did for deletion intervention. Automating the process of negation intervention while preserving a text's discourse coherence and particular style remains a challenge. In addition, our investigations concerning predicate argument structure and responses to semantically equivalent questions have pointed to a serious failing but it remains unclear why the models are behaving in an erratic or almost random fashion. We plan to address this issue in future research.

Another limitation is that we have only shown three out of potentially myriad ways in which language models might fail to capture semantic content. A general solution to the problem of semantic unfaithfulness is something we have not provided in this paper. However, we believe that key to solving this problem is a full scale integration of semantic structure without loss of inferential power in the transformer based language models, something we plan to show in future work.

References

Asher, Nicholas. 1993. *Reference to Abstract Objects in Discourse*. Kluwer Academic Publishers.

Asher, Nicholas. 2011. *Lexical meaning in context: A web of words*. Cambridge University Press.

Asher, Nicholas, Soumya Paul, and Chris Russell. 2021. Fair and adequate explanations. In *International Cross-Domain Conference for Machine Learning and Knowledge Extraction*, pages 79–97, Springer.

Asher, Nicholas, Soumya Paul, and Antoine Venant. 2017. Message exchange games in strategic contexts. *Journal of Philosophical Logic*, 46(4):355–404.

Asher, Nicholas and Sylvain Pogodalla. 2010. Sdrt and continuation semantics. In *JSAI International Symposium on Artificial Intelligence*, pages 3–15, Springer.

Balasubramanian, Sriram, Naman Jain, Gaurav Jindal, Abhijeet Awasthi, and Sunita Sarawagi. 2020. What's in a name? are BERT named entity representations just as good for any other name? In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 205–214, Association for Computational Linguistics, Online.

Barocas, Solon, Moritz Hardt, and Arvind Narayanan. 2019. *Fairness and Machine Learning*. fairmlbook.org. http://www.fairmlbook.org.

Belinkov, Yonatan and Yonatan Bisk. 2018. Synthetic and natural noise both break neural machine translation. In *International Conference on Learning Representations*.

Bennett, M. 1979. *Questions in Montague Grammar*. Indiana University Linguistics Club.

Black, Sid, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, et al. 2022. Gpt-neox-20b: An open-source autoregressive language model. *arXiv preprint arXiv:2204.06745*.

Brabant, Quentin, Gwénolé Lecorvé, and Lina M. Rojas Barahona. 2022. CoQAR: Question rewriting on CoQA. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 119–126, European Language Resources Association, Marseille, France.

Castelvecchi, D. 2022. Are chatgpt and alphacode going to replace programmers? *Nature*.

Chen, Mark, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Chi, Ethan A, John Hewitt, and Christopher D Manning. 2020. Finding universal grammatical relations in multilingual bert. *arXiv preprint arXiv:2005.04511*.

Conia, Simone and Roberto Navigli. 2020. Bridging the gap in multilingual semantic role labeling: a language-agnostic approach. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1396–1410.

Conia, Simone and Roberto Navigli. 2022. Probing for predicate argument structures in pretrained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4622–4632.

De Groote, Philippe. 2006. Towards a montagovian account of dynamics. In *Semantics and linguistic theory*, volume 16, pages 1–16.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Dowty, David R, Robert Wall, and Stanley Peters. 1981. *Introduction to Montague semantics*. Dordrecht. Synthese Library vol. 11.

Elazar, Yanai, Shauli Ravfogel, Alon Jacovi, and Yoav Goldberg. 2021. Amnesic Probing: Behavioral Explanation with Amnesic Counterfactuals. *Transactions of the Association for Computational Linguistics*, 9:160–175.

Fernando, Tim. 2004. A finite-state approach to events in natural language semantics. *Journal of Logic and Computation*, 14(1):79–92.

Fernando, Tim. 2022. Strings from neurons to language. In *Proceedings of the 3rd Natural Logic Meets Machine Learning Workshop (NALOMA III)*, pages 1–10.

Gardner, Matt, Yoav Artzi, Victoria Basmova, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, et al. 2020. Evaluating models' local decision boundaries via contrast sets. *arXiv preprint arXiv:2004.02709*.

Geva, Mor, Uri Katz, Aviv Ben-Arie, and Jonathan Berant. 2021. What's in your head? Emergent behaviour in multi-task transformer models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8201–8215, Association for Computational Linguistics, Online and Punta Cana, Dominican Republic.

Groenendijk, Jeroen. 2003. Questions and answers: Semantics and logic. In *The 2nd CologNET-ElSNET Symposium. Questions and Answers: Theoretical and Applied Perspectives*, pages 16–23, Utrecht.

Hewitt, John and Percy Liang. 2019. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2733–2743, Association for Computational Linguistics, Hong Kong, China.

Hewitt, John and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4129–4138, Association for Computational Linguistics, Minneapolis, Minnesota.

Hossain, Md Mosharaf, Venelin Kovatchev, Pranoy Dutta, Tiffany Kao, Elizabeth Wei, and Eduardo Blanco. 2020. An analysis of natural language inference benchmarks through the lens of negation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9106–9118, Association for Computational Linguistics, Online.

Hosseini, Arian, Siva Reddy, Dzmitry Bahdanau, R Devon Hjelm, Alessandro Sordoni, and Aaron Courville. 2021. Understanding by understanding not: Modeling negation in language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1301–1312, Association for Computational Linguistics, Online.

Hu, Minghao, Yuxing Peng, Zhen Huang, and Dongsheng Li. 2019. A multi-type multi-span network for reading comprehension that requires discrete reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural*

Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 1596–1606, Association for Computational Linguistics, Hong Kong, China.

Jia, Robin and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031, Association for Computational Linguistics, Copenhagen, Denmark.

Jiang, Zhengbao, Frank F Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Ju, Ying, Fubang Zhao, Shijie Chen, Bowen Zheng, Xuefeng Yang, and Yunfeng Liu. 2019. Technical report on conversational question answering.

Kamp, H. and U. Reyle. 1993. *From Discourse to Logic: Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory*. Kluwer Academic Publishers.

Karttunen, L. 1977. Syntax and semantics of questions. *Linguistics and Philosophy*, 1(1):3–44.

Kassner, Nora and Hinrich Schütze. 2020. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7811–7818, Association for Computational Linguistics, Online.

Kaushik, Divyansh, Eduard Hovy, and Zachary C Lipton. 2019. Learning the difference that makes a difference with counterfactually-augmented data. *arXiv preprint arXiv:1909.12434*.

Kusner, Matt J, Joshua Loftus, Chris Russell, and Ricardo Silva. 2017. Counterfactual fairness. In *Advances in Neural Information Processing Systems*, volume 30, pages 4066–4076, Curran Associates, Inc.

Liang, Percy, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. 2022. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*.

Liu, Nelson F., Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019a. Linguistic knowledge and transferability of contextual representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1073–1094, Association for Computational Linguistics, Minneapolis, Minnesota.

Liu, Pengfei, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*.

Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. Roberta: A robustly optimized bert pretraining approach.

Naik, Aakanksha, Abhilasha Ravichander, Norman Sadeh, Carolyn Rose, and Graham Neubig. 2018. Stress test evaluation for natural language inference. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2340–2353, Association for Computational Linguistics, Santa Fe, New Mexico, USA.

Ouyang, Long, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

Pimentel, Tiago, Josef Valvoda, Rowan Hall Maudslay, Ran Zmigrod, Adina Williams, and Ryan Cotterell. 2020. Information-theoretic probing for linguistic structure. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4609–4622, Association for Computational Linguistics, Online.

Rajpurkar, Pranav, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Association for Computational Linguistics, Melbourne, Australia.

Reddy, Siva, Danqi Chen, and Christopher D. Manning. 2019. CoQA: A Conversational Question Answering Challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266.

Reynolds, John C. 1974. On the relation between direct and continuation semantics. In *International Colloquium on Automata, Languages and Programming*.

Roberts, Adam, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Association for Computational Linguistics, Online.

Rogers, Anna, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Sarsa, Sami, Paul Denny, Arto Hellas, and Juho Leinonen. 2022. Automatic generation of programming exercises and code explanations using large language models. In *Proceedings of the 2022 ACM Conference on International Computing Education Research-Volume 1*, pages 27–43.

Schuff, Hendrik, Heike Adel, and Ngoc Thang Vu. 2020. F1 is Not Enough! Models and Evaluation Towards User-Centered Explainable Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7076–7095, Association for Computational Linguistics, Online.

Schölkopf, Bernhard. 2019. Causality for machine learning. *arXiv preprint arXiv:1911.10500*.

Shin, Richard, Christopher H Lin, Sam Thomson, Charles Chen, Subhro Roy, Emmanouil Antonios Platanios, Adam Pauls, Dan Klein, Jason Eisner, and Benjamin Van Durme. 2021. Constrained language models yield few-shot semantic parsers. *arXiv preprint arXiv:2104.08768*.

Shin, Richard and Benjamin Van Durme. 2021. Few-shot semantic parsing with language models trained on code. *arXiv preprint arXiv:2112.08696*.

Song, Liwei, Xinwei Yu, Hsuan-Tung Peng, and Karthik Narasimhan. 2021. Universal adversarial attacks with natural triggers for text classification. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3724–3733, Association for Computational Linguistics, Online.

Sun, Lichao, Kazuma Hashimoto, Wenpeng Yin, Akari Asai, Jia Li, Philip S. Yu, and Caiming Xiong. 2020. Adv-bert: BERT is not robust on misspellings! generating nature adversarial samples on BERT. *CoRR*, abs/2003.04985.

Talmor, Alon, Yanai Elazar, Yoav Goldberg, and Jonathan Berant. 2020. oLMpics-on what language model pre-training captures. *Transactions of the Association for Computational Linguistics*, 8:743–758.

Tenney, Ian, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Association for Computational Linguistics, Florence, Italy.

Tenney, Ian, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Sam Bowman, Dipanjan Das, and Ellie Pavlick. 2019. What do you learn from context? probing for sentence structure in contextualized word representations. In *International Conference on Learning Representations*.

Wolf, Thomas, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Association for Computational Linguistics, Online.

Yang, Zhilin, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, volume 32, Curran Associates, Inc.

Zhang, Wei Emma, Quan Z Sheng, Ahoud Alhazmi, and Chenliang Li. 2020. Adversarial attacks on deep-learning models in natural language processing: A survey. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(3):1–41.

Zhong, Ruiqi, Dhruba Ghosh, Dan Klein, and Jacob Steinhardt. 2021. Are larger pretrained language models uniformly better? comparing performance at the instance level. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3813–3827, Association for

Computational Linguistics, Online.