Are Rotten Apples Edible? Challenging Commonsense Inference Ability with Exceptions

Nam Do
Brown University
Providence, RI, USA
nam_do@alumni.brown.edu

Ellie Pavlick
Brown University
Providence, RI, USA
ellie_pavlick@brown.edu

Abstract

Previous studies have argued that pre-trained language models encode commonsense relational knowledge (e.g. that *apples* are *edible*). However, simultaneous work has revealed that such models are often insensitive to context, even ignoring overt contextual cues such as negations. In this paper, we investigate whether masked language models (the BERT family) can move beyond naive associative biases (e.g., *apple* → *edible*) when the context warrants (e.g. ranking *inedible* higher when presented with the information that the *apple* is *rotten*). We introduce the WINOVENTI procedure, which adversarially exploits generic associations in masked language models to create model-specific Winograd-style entailment schemas. Using our constructed WINOVENTI challenges set of over $2,000$ schemas, we show that language models in the BERT family experience a steep drop in performance on prompts that require them to pick answers which require reasoning about context (e.g., from 89.8% to 18.4% for $\text{BERT}_{\text{LARGE}}$). We present evidence that language models exhibit different associative biases, suggesting a need for future work in developing and analyzing frameworks similar to WINOVENTI that are tuned to model-specific weaknesses.

1 Introduction

Humans exhibit commonsense knowledge through their ability to identify generics (e.g., that a dog has four legs) while still recognizing that exceptions to such rules are possible (e.g., that there are cases of three-legged dogs) (Greenberg, 2007), and that the probability of such exceptions can vary based on the context (e.g. "*the dog is running*" vs. "*the dog is hobbling*"). A prerequisite to comparing a machine's performance to human intelligence is, hence, the verification that machines can exhibit a sensitivity to context that would allow them to perform as well on cases that require reasoning about exceptions as on cases that require recalling generic associations.

Recent work (Petroni et al., 2019) has shown that large pretrained language models, in particular Masked Language Models (MLMs) such as BERT (Devlin et al., 2018) are competent at associating entities with their common characteristics. For example, $\text{BERT}_{\text{LARGE}}$ readily recalls *apple* → *edible* and *charcoal* → *hot*. However, as demonstrated by Ettinger (2020) and Kassner and Schütze (2019), BERT is insensitive to various overt contextual cues, notably negation. For example, given the context "*The shower is* ___," $\text{BERT}_{\text{LARGE}}$ predicts the words "*cold*", "*long*", and "*empty*", the same top 3 predictions it makes given the context "*The shower is not* ___". Such results suggest that while language models like BERT capture many commonsense patterns, such success might be limited to inferences involving common generalizations (appearing in an affirmative context, or using common lexical associations) and not those involving exceptions (appearing in a negative context, or requiring the models to choose less frequently associated lexical items).

In this paper, we investigate whether it is indeed the case that the "commonsense reasoning" exhibited by MLMs is limited to frequent generalizations as opposed to exception cases. We make three main contributions. First, we present the WINOVENTI procedure (§2) for identifying model-specific associative biases and adversarially building Winograd-style challenges (Levesque et al., 2012) to test models' commonsense inference ability. Second, we apply the WINOVENTI procedure to evaluate the commonsense inference performance of a suite of pre-trained MLMs (§3). We find that all the evaluated models experience dramatic performance drops on prompts that require the models to reason about exceptions to commonsense generaliza-

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 2061–2073
August 1–6, 2021. ©2021 Association for Computational Linguistics

WinoVenti Procedure Summary
Step 1: Identifying Generic Associations ***Sample In***: { honey, tarantula } ***Sample Out*** (**BERT$_{\textbf{LARGE}}$**): honey → good, tarantula → poisonous, tarantula → female *See Table 3*
Step 2: Collecting Exceptions of Associations ***Sample Out***: (honey, good) → terrible (tarantula, poisonous) → safe (tarantula, female) → male
Step 3: Adversarial Filtering ***Sample Out*** (**BERT$_{\textbf{LARGE}}$**): True, True, False ***Rationale****: In the last example,* BERT$_{\text{LARGE}}$ *associates the exception characteristic (male) with the entity (tarantula) more strongly than the generic association (female).*
Step 4: Collecting Premises ***Sample Prompt:*** (1) The honey is [good/terrible]. (2) The tarantula is [poisonous/safe]. ***Sample Out***: (1) After adding honey to my tea, it was (delicious/oversweet). (2) Kim was (terrified/thrilled) when he asked her to hold the tarantula.
Step 5: Challenge Set Validation ***Sample Out***: `False`, `False` ***Rationale****: In the first example, word association can be used to select the correct answer. In the second, the property-to-association mapping is not one-to-one, causing ambiguity.*

Table 1: Summary of the WinoVenti pipeline for constructing common sense inference challenge set.

tions. Humans, in contrast, perform consistently well (∼90%) across inferences in our data. Third, we release our human-curated evaluation dataset of 2, 176 sentence pairs probing inferences about commonsense generalizations and exceptions. All of our code and data are available at `http://http://commonsense-exception.github.io`.

2 WinoVenti Procedure

2.1 Overview

The WinoVenti procedure aims to produce Winograd-style sentence pairs to test models' ability to reason about common sense generics and exceptions. For example, a sentence pair might look like the following:

s_g: Zeke says that the apple is <u>delicious</u>. The apple is [MASK]. → edible > inedible

s_e: Zeke says that the apple is <u>rotten</u>. The apple is [MASK]. → inedible > edible

That is, we seek to generate pairs of sentences–which we call s_g and s_e, for generic and exception–which differ by a single word (w_g/w_e), such that that difference should lead to a change in the relative probability of other words (o_g/o_e) in the context. For example, the presence of $w_e =$ *"rotten"* causes $o_e =$ *"inedible"* to be more probable given s_e than given s_g. We seek to generate such pairs adversarially, meaning that, in the example above, we want to ensure that the model generally associates *"edible"* with *"apple"* and thus performing correctly on s_e requires using context to override this prior association.

Our five-step procedure is summarized in Table 1. First (§2.2), given a model and a target noun, we identify the model's *generic associations* (or just *generics*), i.e., the characteristics that the model tends to associate with the noun in a generic context. For example, given a target noun *"apple"*, we might identify *"edible"* as one such association. Second (§2.3), crowd workers are asked to provide contrasting characteristics (or *exceptions*) that could plausibly describe the noun. For example, workers might provide *"inedible"* or *"plastic"* as characteristics that contrast with *"edible"* in the context of *"apple"*. Third (§2.4), we perform an adversarial post-processing step in which we filter out worker-provided exceptions that the model associates with the target more strongly than the original generic. That is, if a worker provides the characteristic *"poison"* and it turns out that the model associates *"poison"* with *"apple"* more strongly than it associates *"edible"* with *"apple"*, we would filter *"poison"* out of our list of exceptions. Fourth (§2.5), we crowdsource premise pairs that would ideally effect the relative probability of the generic characteristic vs. the exception characteristic, such as those shown in s_g and s_e above. Finally (§2.6), the schemas are validated by human annotators, after which we filter out annotations that are trivial or ambiguous. Using this five-step procedure, we construct the WinoVenti$_{\text{BERT LARGE}}$ challenge set of 2,176 sentence pairs.[1]

[1] Available at: `http://commonsense-exception.github.io`.

Prompt pairs	Generic/Exception Outcomes
$s_g =$ Regina screamed when she picked up the pan. The pan is __.	**hot** / cold
$s_e =$ Regina shivered when she picked up the pan. The pan is __.	hot / **cold**
$s_g =$ Tonight Mike's pets would be happy. The pet food is __.	**available** / unavailable
$s_e =$ Tonight Mike's pets would be hungry. The pet food is __.	available / **unavailable**

Table 2: Examples of our schemas. Each *prompt* contains a *premise* with an underlined *special word* (first sentence) and an *outcome sentence*. s_g = generic prompt. s_e = exception prompt. All prompts and exception outcomes are crowdsourced, whereas *generic outcomes* are generic associations identified in some MLM (BERT$_{\text{LARGE}}$, in our paper). Correct answers are in bold.

We draw our target nouns from the THINGS dataset (Hebart et al., 2019) which consists of 1,854 concepts from 27 semantic categories. All of our crowdsourcing tasks are run on Surge (`surgehq.ai`), a high-quality crowdsourcing platform similar to Amazon Mechanical Turk. For the exception and premise generation stages (§2.3 and §2.5), we recruited 100 workers through a qualification task that required the workers to walk through the entire WINOVENTI pipeline. The validation annotation (§2.6) could be performed by anyone on the Surge platform. The pay per response for the crowdsourcing stages is as follows, chosen based on difficulty and time taken to complete the task: (1) exception generation (§2.3): \$0.1, (2) premise generation (§2.5): \$0.3, (3) challenge set validation (§2.6): \$0.2, and (4) human performance: \$0.03.

2.2 Step 1: Identifying Generic Associations

Given a set of target nouns, our first step is to find a set of generic associations which models associate with the noun regardless of context. To do this, we focus on the widely-used BERT$_{\text{LARGE}}$ model (Devlin et al., 2018), specifically the HuggingFace cased, whole word masking implementation (24-layer, 1024 hidden dimension, 16 attention heads, 336M parameters). We base our procedure off of the insight from Ettinger (2020), which demonstrated BERT's insensitivity to negation. For example, given the contexts "*A robin is (not) a [MASK],*" BERT's top two predictions would share {*bird, robin*} in common, or given the contexts "*A daisy is (not) a [MASK],*" BERT's top three predictions would share {*daisy, rose, flower*} in common. Our experiments show that this behavior holds consistent for other MLMs such as RoBERTa (Liu et al., 2019), ALBERT (Lan et al., 2019), Longformer (Beltagy et al., 2020), SqueezeBERT (Iandola et al., 2020), and MobileBERT (Sun et al., 2020). Thus, to identify context-invariant associations, we feed in both affirmative and negated templates[2] of the form "*The* `[ENTITY]` *is (not) [MASK]*". We then define the models *generic associations* to be the set of words that are in common between the top-k predictions in the affirmative context and the top-k predictions in the negative context. We experiment with $k \in \{1, 3, 5, 8\}$. Table 3 shows generic associations generated for BERT$_{\text{LARGE}}$ in this way for different values of k.

k	Generic Associations for BERT$_{\text{LARGE}}$
1	desk → empty, tarantula → poisonous, couch → comfortable, syrup → sweet, honey → sweet, compass → accurate
3	desk → there, tarantula → edible, couch → empty, syrup → bitter, honey → good, compass → true
5	desk → full, tarantula → {female, male}, couch → warm, syrup → edible honey → edible, compass → right
8	desk → {mine, clean}, tarantula → small couch → {clean, big}, syrup → orange, honey → {delicious, there, honey} compass → wrong

Table 3: Examples of BERT$_{\text{LARGE}}$ generic associations for different values of k. Generic associations are cumulative across different k values (e.g., in the end, we have *compass* → { accurate, true, right, wrong }).

How similar are associations across models? Since our dataset is constructed in particular to be adversarial against BERT$_{\text{LARGE}}$, it is relevant to ask whether such a dataset will be equally adversarial against all models in the BERT family. To quantify whether models in the BERT family (described in §3.2) differ in the generic associations

[2] We use a simple set S of two templates that differ in the ending punctuation (S = {"*The* `[ENTITY]` *is* `[MASK].''`, ``` ``The ``` `[ENTITY]` *is* `[MASK],''`}).

Model name	k = 1	k = 3	k = 5
BERT$_{\text{BASE}}$	0.59	0.26	0.17
BERT$_{\text{LARGE}}$	1	1	1
RoBERTa$_{\text{SMALL}}$	0.61	0.2	0.11
RoBERTa$_{\text{LARGE}}$	0.57	0.19	0.12
DistilRoBERTa	0.53	0.15	0.08
DistilBERT	0.57	0.18	0.12
SqueezeBERT	0.6	0.23	0.13
MobileBERT	0.65	0.28	0.16
ALBERT$_{\text{BASE}}$	0.42	0.07	0.04
ALBERT$_{\text{LARGE}}$	0.51	0.1	0.07
ALBERT$_{\text{XLARGE}}$	0.59	0.18	0.08
ALBERT$_{\text{XXLARGE}}$	0.48	0.13	0.07

Table 4: Jaccard similarity of Generic Associations made by models with respect to BERT$_{\text{LARGE}}$. The closer to 1, the more similar.

identified using this procedure, we compute the Jaccard similarity–i.e., $\frac{|A \cap B|}{|A \cup B|}$, with A and B being the set of associations made by two models. Table 4 shows the results for $k = 1, 3$, and 5. BERT$_{\text{BASE}}$ and BERT$_{\text{LARGE}}$ have the highest similarity (with a mean Jaccard score of 0.72). Most other models only share roughly half of the generic associations with another model. As k increases, generic associations start to differ across models (signified by a decrease in the Jaccard similarity score). This signifies that models differ significantly in the generic associations that they make.

Qualitatively, associations made by BERT-family, RoBERTa-family and other models differ from one another. While the evaluated RoBERTa-family models display a slight tendency to associate the words *"empty"*, *"broken"*, *"dead"* with many nouns, the BERT-family models tend to make other associations such as *"used"*, and *"edible"*. ALBERT models, on the other hand, make quite different associations from all other models, such as *"adjustable"*, *"gone"*, *"a"*, and *"covered"* (with a full 41% of the associations made by ALBERT$_{\text{XXLARGE}}$ being *"adjustable"*).

As we base this study on the associations made by BERT$_{\text{LARGE}}$, future work is to be done to extract the different associations made by other MLMs to pose a harder and richer set of challenges for those models.

2.3 Step 2: Crowdsourcing Exceptions to Generic Associations

Given the identified associative biases, we then use crowdsourcing to find alternative characteristics that can be true of the noun, but are perhaps less stereotypically associated (e.g., *"apples"* can be *"inedible"* if, for example, they are made of plastic, or are poisoned). Workers are given statements of the form "*The* `[NOUN]` *is* `[PROPERTY]`" where `[PROPERTY]` is one of the generic associations collected as described above (e.g., "*The apple is* *edible*"), and then is asked to provide up to three adjectives that would describe `[NOUN]` in the case where it is *not* `[PROPERTY]` (e.g., in the case that the "*apple*" is not "*edible*"). To increase the quality of tasks presented to workers in later stages, we also ask workers in this stage whether or not a presented statement makes sense, and filter out any sentences which workers flag as nonsensical. Of the noun-property associations generated in our first step, 10.45% are filtered out in this step.

The model-generated associative pairs can be noisy. Particularly, we note that as we increase k, the model becomes more likely to generate nonsense pairs (e.g., *glass* → *glass*, *dart* → *legal*, *triangle* → *circular*), and that the stereotypical strength of association decreases (as shown in Table 3). Thus, to increase the quality of the final challenge set and minimize workers' confusion, our pipeline uses two main criteria to select which association pairs to present to workers. First, associations are selected in an increasing order of k, meaning we would include all associations at $k = 1$ first, and then those at $k = 3$, and so on. Second, the templated statements presented to workers are ranked according to perplexity scores assigned to them by a unidirectional language model. For each k of interest, we identify the inflection point in the perplexity score, and only retain samples for which the perplexity is below this point. At the end when we have gathered more samples than we plan to post, we perform a final round of inflection point identification and retention. This leaves us with 1990 entity-association pairs.

To construct WinoVenti$_{\text{BERT LARGE}}$, we selected 1990 entity-bias pairs to present to workers (331 pairs were of 1-associative biases, 780 of 3-associative biases, 825 of 5-associative biases, and 54 of 8-associative biases). 28 workers participated in this stage, and for each task workers were paid \$0.1. At the end, we received $2,994$ valid exception annotations from crowd workers.

2.4 Step 3: Adversarial Filtering

The goal of adversarial filtering is to make sure that a model clearly favors the generic association with a target noun (identified by an MLM) over an exception association (crowdsourced) without any additionally introduced context. A triplet of (*Target Noun*, *Generic Association*, *Exception Association*) passes our adversarial filtering stage if the probability that the model associates the *Generic Association* with the *Target Noun* (through statements of template “*The* `[NOUN]` *is* ___”) is higher than that which the model associates with the *Exception Association*.

This filtering is adversarial in that, by making sure that a model innately favors one association (e.g., *edible*) with a target noun (e.g., *apple*) over another (e.g., *inedible*), the model has to demonstrate a strong enough sensitivity to context to select the other association over the one that it innately prefers (e.g., when presented with the information that the *apple* is *rotten*).

In the construction of WinoVenti$_{\text{BERT LARGE}}$, after adversarially filtering using BERT$_{\text{LARGE}}$, we retained 2,745 (*Target Noun*, *Generic Association*, *Exception Association*) triplets. Some examples of triplets that were filtered out (in the same format) are: (*stair*, *long*, *short*), or (*mug*, *full*, *empty*).

2.5 Step 4: Crowdsourcing Premises

As a result of Step 3, we have a set of (noun, generic associations, exception associations) triplets that have met our adversarial filtering criteria. We then ask workers to create a minimal pair of *premise* sentences (s_g/s_e) differing by exactly one word (w_g/w_e) which differ in whether the *generic outcome* (o_g) or the *exception outcome* (o_e) is the most probable continuation. For example, given the triplet (*“apple”*, *“edible”*, *“inedible”*), our sentence pair might be $s_g =$*“the apple is sweet* and $s_e =$*“the apple is rotten*. To minimize the cognitive load on crowdworkers while still communicating these requirements, we provided examples of two good and two bad premise pairs with a brief explanation each of why each was good/bad, covering all the requirements above.[3] We also encouraged workers to be creative and provide premises that have diverse sentence structures. In total, 2,745 premise pairs were collected. Table 5 shows several examples of contexts generated in this way.

Crowdsourced Premises
Given: The mail is [anonymous/identifiable]. **Annotation:** I received a letter in the mail from a (stranger/friend).
Given: The pill is [safe/unsafe]. **Annotation:** You will feel (better/worse) if you take the pill.
Given: The paper bag is [empty/full]. **Annotation:** I took a (folded/heavy) bag to the store.
Given: The timer is [accurate/inaccurate]. **Annotation:** Jim was (early/late) to work of his timer.
Given: The bed is [comfortable/uncomfortable]. **Annotation:** Lola slept [soundly/miserably].

Table 5: Premise annotations collected given a target noun and the identified generic/exception associations, combined into a natural language sentence of format “*The* ___ *is* ___” presented to workers.

2.6 Step 5: Challenge Set Validation

We evaluate each sentence pair using two criteria. First, is it the case that the contexts differentiate the probabilities of o_g and o_e such that o_g only makes sense given s_g and o_e only makes sense given s_e. E.g., given “*Matthew says that the apple is rotten. The apple is [MASK]*”, can *edible* be a sensible answer)? Second, we ensure that the special words are not synonymous with the outcomes, i.e., that w_g and o_g are not synonyms, nor are w_e and o_e. If the majority of workers (out of three) judge the sentence pairs to pass the above criteria, the pair is deemed valid. Criterion 1 is to ensure that the outcomes are unambiguous to humans, while criterion 2 is to ensure that synonymy can not be used to trivially find the correct answer. For example, criterion 1 filtered out prompts such as “*Mary made popsicles out of vitamins. The popsicle is [MASK].*” where both the choices (*edible / nutritious*) could apply. Criterion 2 filtered out prompts like “*The doctor used a loaded test tube. The test tube is [MASK].*” where the correct answer (*filled/hollow*) could easily be selected using word association. Of the 2,678 prompt pairs posted for evaluation, 502 were filtered out for failing at least one of the two aforementioned criteria, leaving us with the

[3] Our exact instructions and examples given to workers are provided in Appendix A.2

final challenge set of $2,176$ prompt pairs ($4,352$ prompts in total).

2.7 WinoVenti Challenge Set

To summarize, the final WINOVENTI$_{\text{BERT LARGE}}$ challenge set consists of $2,176$ prompt pairs ($4,352$ challenge prompts in total) about 978 distinct entities. Each entity has at most 5 generic associations, for a total of 186 distinct generic associations identified by BERT$_{\text{LARGE}}$ across all entities. The length of premises are on average 8 words. Using SpaCy's Part of Speech tagger, we identified that the special words (w_g and w_e) across different premises are predominantly adjectives (apprx. 34%), nouns (apprx. 30.3%), and verbs (apprx. 22.4%), with the presence of other parts of speech such as adverbs, adpositions, numbers, determiners, proper nouns, or pronouns. Approximately 54.5% of the premises have the special words in the last third, 28.4% in the middle third, and the rest 16.9% in the first third of the premise. The challenge set is available at `http://commonsense-exception.github.io/`.

3 Experiments

3.1 Task Definition

The task is defined as: given a pair of sentences (s_g, s_e) with the format in Table 2, and the pair of single-word candidate outcomes (o_g, o_e), does the model correctly rank o_g as more probable than o_e given s_g (general test), and, symmetrically, rank o_e as more probable than o_g given s_e (exception test). An MLM's performance is reported as the percentage of sentences for which the language model successfully gives the correct answer a higher probability of filling in the blank than the incorrect answer. Each model is evaluated on three different subsets of the full WINOVENTI$_{\text{BERT LARGE}}$ dataset, as follows: **All** refers simply to the entire WINOVENTI$_{\text{BERT LARGE}}$ challenge set (which is only adversarially filtered using BERT$_{\text{LARGE}}$); **Individual** refers to a model-specific subset of WINOVENTI$_{\text{BERT LARGE}}$, specifically those pairs which result after additionally model-specific adversarial filtering; **Intersection** refers to the set of 188 prompts (94 each for the general test and the exception test) that result when we take the intersection of each of the model-specific subsets generated described in Individual. Note that both Individual and Intersection reflect the performance of models in an adversarial setting, and we do not expect these subsets to show meaningfully different results. We include Intersection simply so that we can compare all models on a fixed test set in an apples-to-apples setting, since the Individual subsets will vary from one model to the next.

3.2 Models

We study the performance of the following pretrained models (HuggingFace implementation) on our WINOVENTI$_{\text{BERT LARGE}}$ challenge set:

BERT-family We used BERT$_{\text{BASE}}$ (cased) and BERT$_{\text{LARGE}}$ (cased, trained with a whole word masked language modeling objective) models (Devlin et al., 2018). We also evaluated DistilBERT (Sanh et al., 2019) (cased), which has 40% less parameters than BERT while still performing almost as good as the original model. MobileBERT (Sun et al., 2020), similarly seeking to compress and accelerate the BERT model, is also included in our experiments (uncased). Additionally, our experiments also evaluated SqueezeBERT (Iandola et al., 2020) (uncased), which has a similar bidirectional transformer architecture like BERT, but uses grouped convolutions instead of certain fully-connected layers.

RoBERTa-family Our experiments also evaluate RoBERTa$_{\text{BASE}}$ and RoBERTa$_{\text{LARGE}}$ (Liu et al., 2019) versions that were trained on a masked modeling objective. RoBERTa build on BERT, differing in hyperparameter choices and pre-training objective. We also used DistilRoBERTa (Sanh et al., 2019), which follows the same training and distillation process as DistilBERT, but based on RoBERTa instead.

ALBERT-family We additionally evaluated ALBERT$_{\{\text{BASE, LARGE, XLARGE, XXLARGE}\}}$ models (Lan et al., 2019). Built off of BERT, ALBERT is designed to reduce the number of parameters and perform better on downstream tasks with multi-sentence inputs.

3.3 Finetuning

To evaluate how the models perform after being fine-tuned on datasets that contain exceptions in similar sentence structures to our challenge set, we fine-tuned a subset of the models in question (BERT$_{\text{BASE}}$, RoBERTa$_{\text{BASE}}$, DistilRoBERTa, SqueezeBERT, MobileBERT, and ALBERT$_{\text{BASE}}$) on two subsets of our dataset, the *half* and the *full* exception train sets. The *half*

	All		Individual			Intersection	
Model name	**Generic**	**Exceptn**	**Generic**	**Exceptn**	**N**	**Generic**	**Exceptn**
BERT$_{\text{BASE}}$	83.0	23.1	89.3	16.7	1871	96.0	8.3
BERT$_{\text{LARGE}}$	89.8	18.4	89.8	18.4	2176	97.4	11.4
RoBERTa$_{\text{BASE}}$	79.5	27.2	87.3	18.8	1801	95.6	12.7
RoBERTa$_{\text{LARGE}}$	81.3	29.2	91.1	18.9	994	95.6	17.5
DistilRoBERTa	78.3	26.5	91.1	13.1	948	93.9	14.0
DistilBERT	85.8	19.4	89.8	15.2	1628	94.8	12.7
SqueezeBERT	85.5	21.1	92.1	13.6	1823	97.4	7.9
MobileBERT	82.7	23.0	89.8	15.6	1856	96.9	7.0
ALBERT$_{\text{BASE}}$	77.4	28.0	87.2	18.4	1525	94.3	11.0
ALBERT$_{\text{LARGE}}$	79.6	26.8	87.6	18.9	1324	93.9	12.2
ALBERT$_{\text{XLARGE}}$	80.7	31.9	89.0	23.5	777	96.1	13.1
ALBERT$_{\text{XXLARGE}}$	77.7	32.4	80.0	32.4	959	83.8	31.9
Human performance	91.1	90.2	-	-	-	-	-

Table 6: Models' performance on our *generic* and *exception* prompts. Generic tests evaluate whether, given a generic prompt, the model would rank the generic outcome o_g higher than the exception o_e). Exceptn tests similarly check if the exception outcome o_e is ranked higher than the generic o_g given an exception context.

exception set is created by selecting half of the WinoVenti$_{\text{BERT LARGE}}$ challenge set (2176 out of 4352 schemas). 50% of the training schemas (1088 out of 2176) are selected to be generic schemas, and the rest are exception schemas. With this configuration, models are fine-tuned on both generic and exception schemas, and are sequentially evaluated on unseen challenges (with a similar distribution of 50% generic schemas and 50% exception).

With the *full exception* train set, models are fine-tuned exclusively on exception schemas. From the set of exception challenges (2176 schemas), we performed a 80-20 train-test split to select the *full exception* train set. To evaluate models trained on the *full exception train set*, in addition to evaluating the model accuracy on the held out exceptions (20% - 435 of the 2176 exception schemas), we also evaluate the fine-tuned models' on (1) all the generic challenges (2176 schemas), and (2) on a test set similar to the above where half of the schemas are generic challenges and another half are unseen exception challenges (870 schemas). The different test scenarios help us understand how finetuning on exception challenges influences models' performance on not only unseen exception challenges, but also on generic challenges.

3.4 Results and Analysis

Table 6 shows each model's performance across each data subset (All, Individual, and Intersection) broken down by task (i.e., generic vs. exception test). Across the board, models perform significantly better on the generic tests than on the exception tests (where accuracies are well below the random baseline of 50%). This provides strong evidence that models do not truly encode "common sense", but rather simply recall generic associations in a context-agnostic manner.

Looking closely at the results on the All subset, we see that models' performance on the generic test is overall lower and the performance drop on the *exception test* is less dramatic, compared to the results on the Individual and Intersection subsets. This trend is expected, since, after adversarial filtering, each model is only evaluated on prompts where the model is inherently (before being introduced to any additional context) skewed towards choosing the generic association as a description of the target noun. Even so, the numbers on this All set are informative: they emphasize that models' apparent success at recalling "commonsense" associations is likely largely driven by artifact. That is, when assessed on a set of common sense inferences that don't necessarily involve words from the top of a model's distribution in a given context, performance is quite poor. On inspection, we see that, for models outside the BERT family, the poor performance is often attributable to low probabilities for both the generic and exception outcomes (o_g and o_e) in both contexts, meaning the difference between the probabilities is often small. In other words, these models don't encode the same generic associations that $\text{BERT}_{\text{LARGE}}$ encodes and, moreover, don't encode much difference at all between o_g and o_e.

Error Analysis Looking closely at specific prompts on which models succeed and fail reveals some interesting trends. For each model, we look at the top 20 sentences from the *exception test* on which the model exhibited the largest errors, where the size of the error is measured by the difference in the probability that the model assigns to the incorrect outcome (in this case o_g) compared to the correct outcome (o_e). For $\text{BERT}_{\text{LARGE}}$, we find that 55% of these top 20 failures involve strong generic associations (i.e., those generated with $k = 1$). A full 65% are cases when o_g is "*empty*", suggesting that $\text{BERT}_{\text{LARGE}}$ prefers to assign high probability to this word across all contexts (see §2.2). In contrast, looking at the top 20 sentences on which $\text{BERT}_{\text{LARGE}}$ performed best, we learn that only one involves $o_g =$"*empty*", and only four involve strong generic associations ($k = 1$).

Performing similar analysis for the other BERT-family models ($\text{BERT}_{\text{SMALL}}$, DistilBERT, MobileBERT, SqueezeBERT), we see that the majority of the successes involve non-associative noun-characteristic pairs (i.e., pairs where the characteristic is not identified as a generic association with the noun by our procedure described in Section 2.2). For $\text{BERT}_{\text{SMALL}}$ and DistilBERT, 40% of their 20 most successful cases involved nouns for which the models did not encode any generic associations. For SqueezeBERT and MobileBERT, it is 55% and 60%, respectively. This may signify stronger a relationship between a model **not** identifying a generic association with a target noun and that model being sensitive to a change in context about that target noun.

Fine-tuning Analysis The results of our fine-tuning experiments are shown in Figure 1 (with additional results in §A.1). We see that, in general, fine-tuning models on a dataset that contains exceptions (where the challenge format remains the same between the train and test sets) can increase the performance on unseen exceptions, but does so at the expense of performance on generic prompts. Specifically, when we train on a mix of generic and exception schemas (our *half exception set*), the model improves only slightly in performance on exceptions, and converges to the same trend as the un-finetuned model: i.e., performance on generics far exceeds that on exceptions. In contrast, when we train on only exception schemas (our *full exception set*), the performance on unseen exception challenges increases faster and more significantly, but this increase is at the expense of the rapid decrease of performance on generic challenges.

This poor performance on exceptions (at the expense of their performance on generics), suggests that the conceptual associations encoded by MLM models is fairly shallow: even with finetuning, the models are not able to differentiate these types of context-specific associations in a way that allows them to perform well on both types of inferences simultaneously. Future work is needed to develop models with different architectures or loss functions that might be capable of encoding more nuanced conditional associations.

4 Related Work

Ettinger et al. (2017) bring up the fundamental problem of NLP models ignoring rare language phenomena, as they typically rely on independently and identically distributed probably-approximately-correct model of learning, and as they often use overly simplistic loss functions. Complementary to our project, Ettinger et al. (2017) encourage robust error analysis of NLP systems through developing challenges that are based on linguistic phenomena, and that have a low barrier to entry.

Common sense and probing. NLP has been interested in encoding commonsense relations for a long time (Liu and Singh, 2004). Recent work has shown how pre-trained LMs exhibit common sense knowledge even before fine-tuning (Petroni et al., 2019), and that they can be built and used to mine more commonsense information (Bosselut et al., 2016; Davison et al., 2019). While this signifies how LMs encode some common sense and prototypical properties of nouns (Weir et al., 2020), many researchers are pointing out these models' insensitivity to context (Ettinger, 2020; Ravichander et al., 2020), which is antithetical to common sense.

Challenge Sets Many existing challenge sets have provided concrete frameworks to evaluate models inference ability, through coreference resolution (notably Winograd Schema Challenge - WSC (Levesque et al., 2012)) or pronoun resolution (notably in PDP (Morgenstern et al., 2016)). In this work, similar to the Winograd Schemas (Levesque

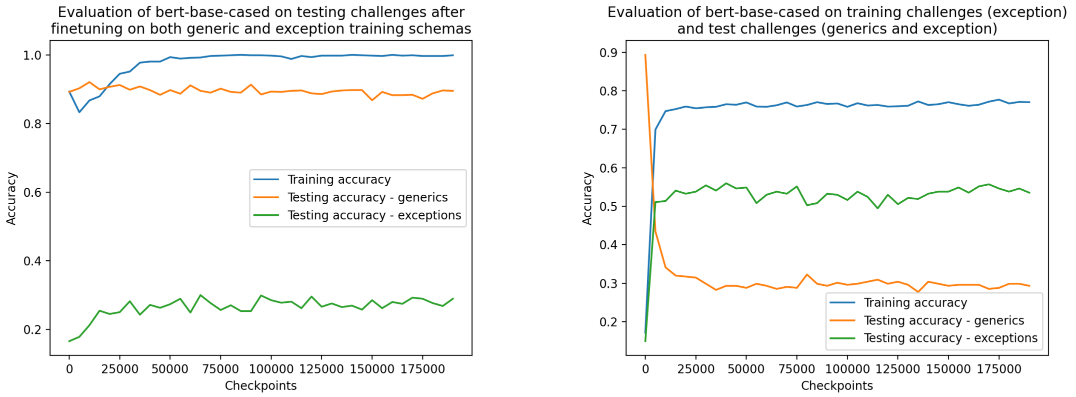


Figure 1: **Left**: Fine-tuning BERT$_{\text{BASE}}$ on a dataset that contains both generics and exceptions results in a minimal increase in performance on exceptions. **Right**: Fine-tuning BERT$_{\text{BASE}}$ on a dataset containing only exceptions results in an increase in performance on exceptions, at the expense of the accuracy on generics.

et al., 2012), we also generate pairs with a simialr structure (pairs of premises that differ in one word that would determine the answer). However, while Winograd schemas are model-agnostic, our approach factors in models' behavior in the design of schemas in order to guarantee models' bias towards one specific answer (for each prompt).

Sakaguchi et al. (2020) build the WINOGRANDE adversarial challenge set through using language models to detect and filter out schemas with language-based biases that would trivialize the task of picking the correct answer. WINOGRANDE aims to minimize the chance of models getting the right answers for the wrong reasons (through leveraging simple lexical associations that are annotation artifacts by human annotators). Our key motivation, meanwhile, is to adversarially leverage the biases that models associate with entities to "trick" them into choosing incorrect answers. Our work uses adversarially constructed test sets to expose heuristics that models use. This technique has been used widely in probing/analysis work, e.g., (Glockner et al., 2018; Naik et al., 2018; Jia and Liang, 2017; Nie et al., 2019; McCoy et al., 2019). The idea of improving models performance on "exceptions" to "generalizations" also shares much in common with work on bias and fairness in NLP (Rudinger et al., 2018; Zhao et al., 2018, 2019).

Gardner et al. (2020) propose the development of *contrast sets*, which can be developed by manually perturbing existing datasets in small but meaningful ways that would change the gold label. Our work, in contrast, factor in models' insensitive associations into the construction of challenges in addition to a slight change in context that is leveraged by contrast sets. Kaushik et al. (2019) similarly propose using *counterfactually-augmented data* to make models more robust against spurious associations. Our work adds to this work by demonstrating that fine-tuning on exception challenges can increase the performance of models on tail cases at the expense of the performance on generic prompts.

5 Conclusion

We present the WINOVENTI procedure, which adversarially exploits generic associations in masked language models to create model-specific Winograd-style schemas. Using our constructed WINOVENTI$_{\text{BERT LARGE}}$ challenge set, we test whether MLMs can move beyond their naive associations to select the more likely outcomes depending on the input context. We find a steep drop in models' performance on our challenges that require a sensitivity to context. We present evidence that generic associations differ from one model to another, highlighting the need for other model-specific challenge sets that are tuned to associative biases of models other than BERT$_{\text{LARGE}}$, and to (2) develop and analyze frameworks like WINOVENTI.

Acknowledgments

We thank Charlie Lovering, and the members and friends of the Brown University LUNAR Lab for all the invaluable conversations, proofreading, and feedback. We thank Edwin Chen (Surge) for their support with our crowdsourcing tasks.

References

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Antoine Bosselut, Jianfu Chen, David Warren, Hannaneh Hajishirzi, and Yejin Choi. 2016. Learning prototypical event structure from photo albums. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1769–1779, Berlin, Germany. Association for Computational Linguistics.

Joe Davison, Joshua Feldman, and Alexander Rush. 2019. Commonsense knowledge mining from pretrained models. 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 1173–1178, Hong Kong, China. Association for Computational Linguistics.

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

Allyson Ettinger. 2020. What bert is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, 8:34–48.

Allyson Ettinger, Sudha Rao, Hal Daumé III, and Emily M Bender. 2017. Towards linguistically generalizable nlp systems: A workshop and shared task. *arXiv preprint arXiv:1711.01505*.

Matt Gardner, Yoav Artzi, Victoria Basmov, 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. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 1307–1323.

Max Glockner, Vered Shwartz, and Yoav Goldberg. 2018. Breaking NLI Systems with Sentences that Require Simple Lexical Inferences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 650–655. Association for Computational Linguistics.

Yael Greenberg. 2007. Exceptions to generics: Where vagueness, context dependence and modality interact. *Journal of Semantics*, 24(2):131–167.

Martin N Hebart, Adam H Dickter, Alexis Kidder, Wan Y Kwok, Anna Corriveau, Caitlin Van Wicklin, and Chris I Baker. 2019. Things: A database of 1,854 object concepts and more than 26,000 naturalistic object images. *PloS one*, 14(10):e0223792.

Forrest N Iandola, Albert E Shaw, Ravi Krishna, and Kurt W Keutzer. 2020. Squeezebert: What can computer vision teach nlp about efficient neural networks? *arXiv preprint arXiv:2006.11316*.

Robin Jia 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.

Nora Kassner and Hinrich Schütze. 2019. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. *arXiv preprint arXiv:1911.03343*.

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

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Hector Levesque, Ernest Davis, and Leora Morgenstern. 2012. The winograd schema challenge. In *Thirteenth International Conference on the Principles of Knowledge Representation and Reasoning*. Citeseer.

Hugo Liu and Push Singh. 2004. Conceptnet—a practical commonsense reasoning tool-kit. *BT technology journal*, 22(4):211–226.

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. *arXiv preprint arXiv:1907.11692*.

R Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. *arXiv preprint arXiv:1902.01007*.

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

Aakanksha Naik, 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.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2019. Adversarial nli: A new benchmark for natural language understanding. *arXiv preprint arXiv:1910.14599*.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. 2019. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*.

Abhilasha Ravichander, Eduard Hovy, Kaheer Suleman, Adam Trischler, and Jackie Chi Kit Cheung. 2020. On the systematicity of probing contextualized word representations: The case of hypernymy in BERT. In *Proceedings of the Ninth Joint Conference on Lexical and Computational Semantics*, pages 88–102, Barcelona, Spain (Online). Association for Computational Linguistics.

Rachel Rudinger, Jason Naradowsky, Brian Leonard, and Benjamin Van Durme. 2018. Gender bias in coreference resolution. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 8–14, New Orleans, Louisiana. Association for Computational Linguistics.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Winogrande: An adversarial winograd schema challenge at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8732–8740.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Zhiqing Sun, Hongkun Yu, Xiaodan Song, Renjie Liu, Yiming Yang, and Denny Zhou. 2020. Mobilebert: a compact task-agnostic bert for resource-limited devices. *arXiv preprint arXiv:2004.02984*.

Nathaniel Weir, Adam Poliak, and Benjamin Van Durme. 2020. Probing neural language models for human tacit assumptions. CogSci.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Ryan Cotterell, Vicente Ordonez, and Kai-Wei Chang. 2019. Gender bias in contextualized word embeddings. *arXiv preprint arXiv:1904.03310*.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2018. Gender bias in coreference resolution: Evaluation and debiasing methods. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 15–20, New Orleans, Louisiana. Association for Computational Linguistics.

A Appendices

A.1 Fine-tuning Experiment

A.2 Crowdsourcing Instructions

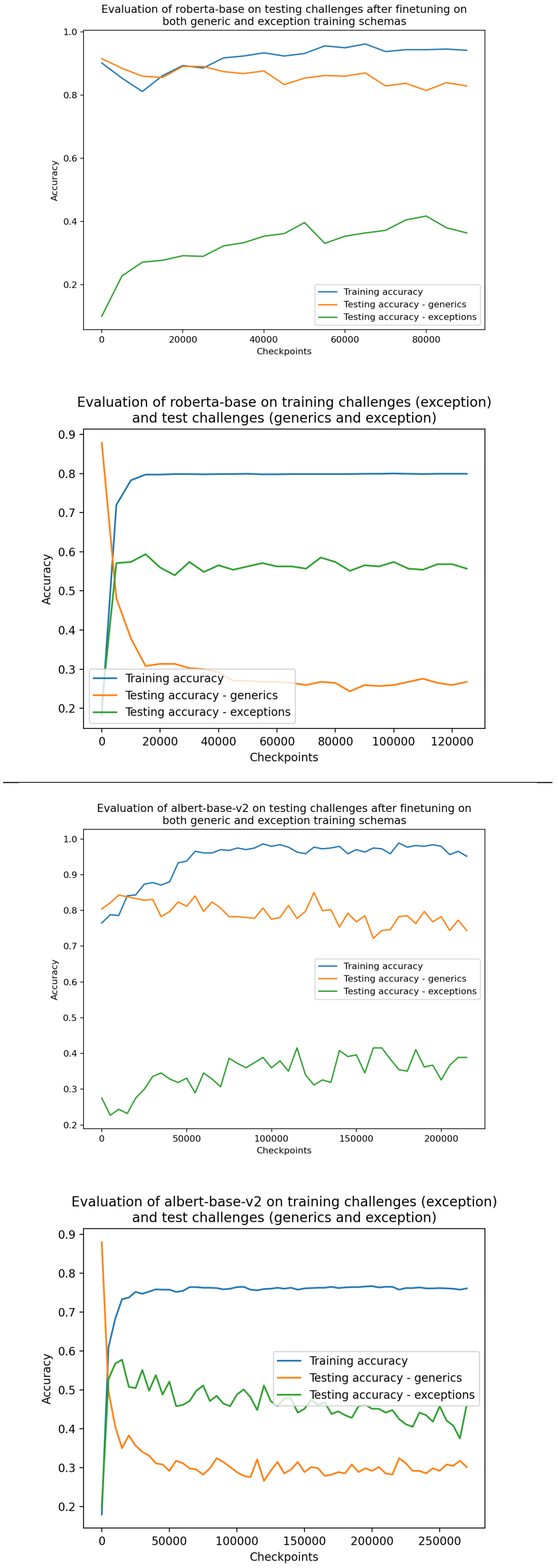
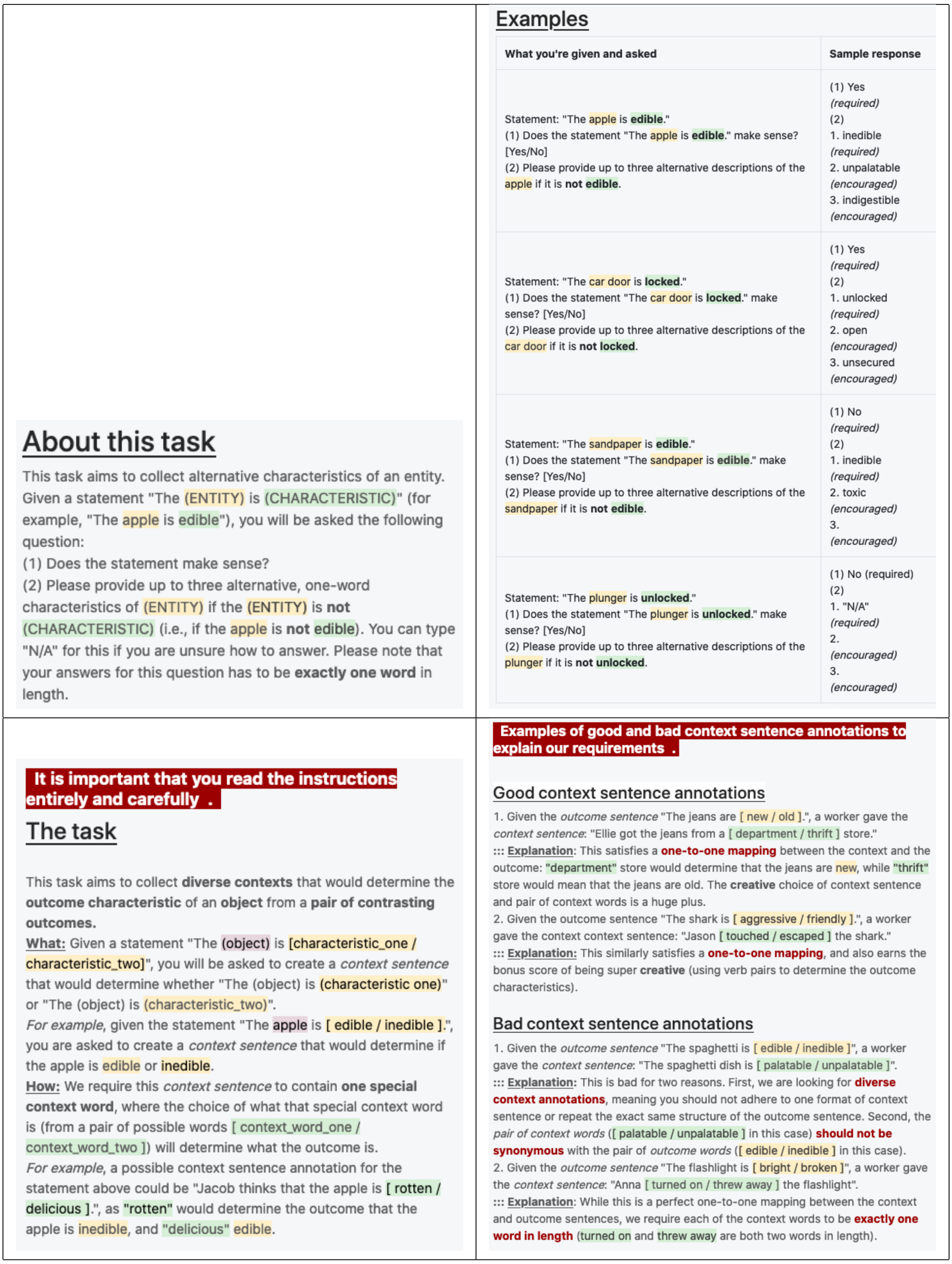


Figure 2: Fine-tuned ALBERT$_{\text{BASE}}$ and RoBERTa$_{\text{BASE}}$ show the same negative relationship between generic and exception performances. Fine-tuned RoBERTa$_{\text{BASE}}$ shows a higher increase in performance on unseen exceptions in comparison to BERT$_{\text{BASE}}$.



About this task

This task aims to collect alternative characteristics of an entity. Given a statement "The (ENTITY) is (CHARACTERISTIC)" (for example, "The apple is edible"), you will be asked the following question:
(1) Does the statement make sense?
(2) Please provide up to three alternative, one-word characteristics of (ENTITY) if the (ENTITY) is **not** (CHARACTERISTIC) (i.e., if the apple is **not** edible). You can type "N/A" for this if you are unsure how to answer. Please note that your answers for this question has to be **exactly one word** in length.

Examples

What you're given and asked	Sample response
Statement: "The apple is **edible**." (1) Does the statement "The apple is **edible**." make sense? [Yes/No] (2) Please provide up to three alternative descriptions of the apple if it is **not edible**.	(1) Yes *(required)* (2) 1. inedible *(required)* 2. unpalatable *(encouraged)* 3. indigestible *(encouraged)*
Statement: "The car door is **locked**." (1) Does the statement "The car door is **locked**." make sense? [Yes/No] (2) Please provide up to three alternative descriptions of the car door if it is **not locked**.	(1) Yes *(required)* (2) 1. unlocked *(required)* 2. open *(encouraged)* 3. unsecured *(encouraged)*
Statement: "The sandpaper is **edible**." (1) Does the statement "The sandpaper is **edible**." make sense? [Yes/No] (2) Please provide up to three alternative descriptions of the sandpaper if it is **not edible**.	(1) No *(required)* (2) 1. inedible *(required)* 2. toxic *(encouraged)* 3. *(encouraged)*
Statement: "The plunger is **unlocked**." (1) Does the statement "The plunger is **unlocked**." make sense? [Yes/No] (2) Please provide up to three alternative descriptions of the plunger if it is **not unlocked**.	(1) No (required) (2) 1. "N/A" *(required)* 2. *(encouraged)* 3. *(encouraged)*

It is important that you read the instructions entirely and carefully .

The task

This task aims to collect **diverse contexts** that would determine the **outcome characteristic** of an **object** from a **pair of contrasting outcomes.**
What: Given a statement "The (object) is [characteristic_one / characteristic_two]", you will be asked to create a *context sentence* that would determine whether "The (object) is (characteristic one)" or "The (object) is (characteristic_two)".
For example, given the statement "The apple is [edible / inedible].", you are asked to create a *context sentence* that would determine if the apple is edible or inedible.
How: We require this *context sentence* to contain **one special context word**, where the choice of what that special context word is (from a pair of possible words [context_word_one / context_word_two]) will determine what the outcome is.
For example, a possible context sentence annotation for the statement above could be "Jacob thinks that the apple is [rotten / delicious].", as "rotten" would determine the outcome that the apple is inedible, and "delicious" edible.

Examples of good and bad context sentence annotations to explain our requirements .

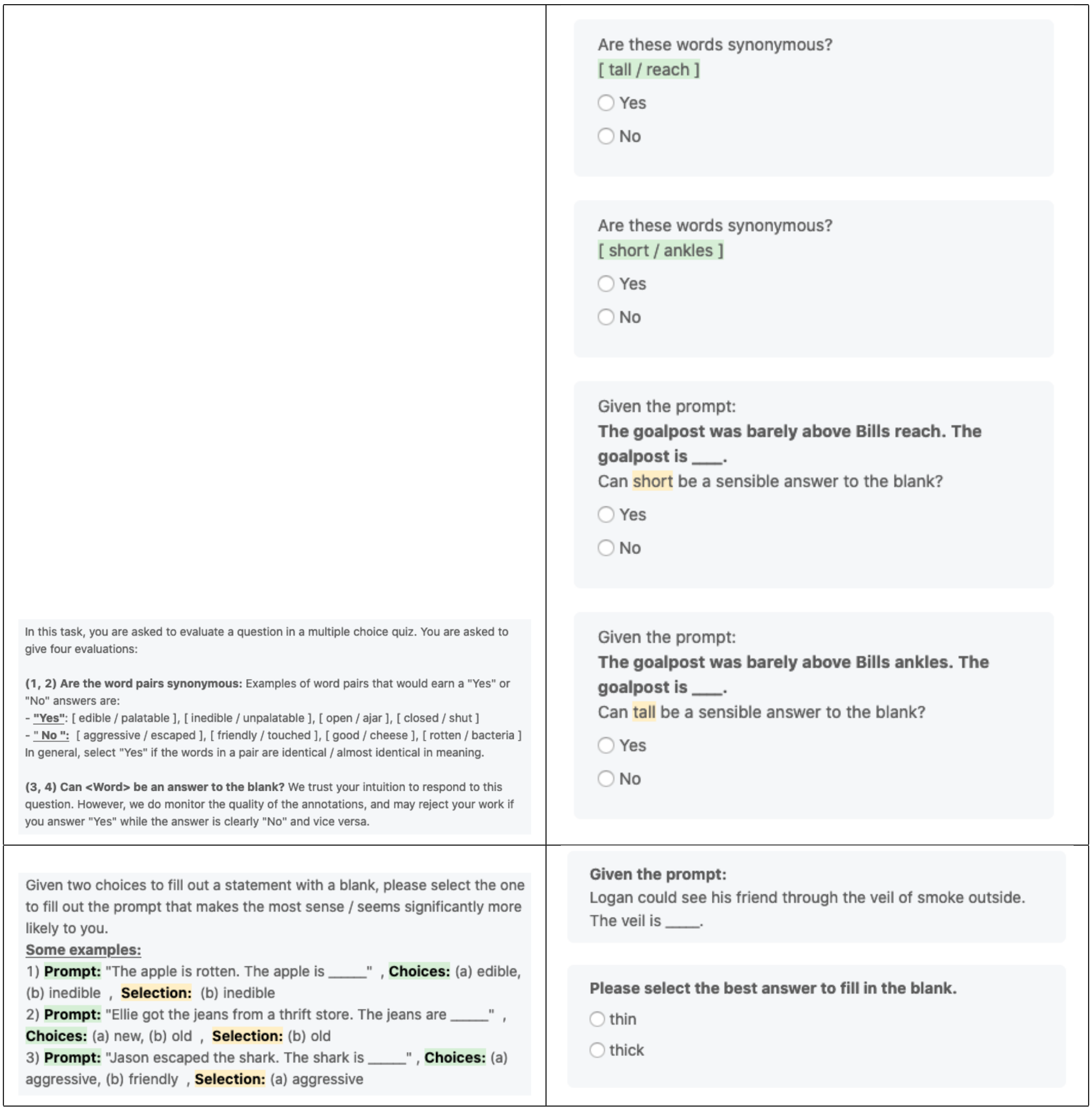
Good context sentence annotations

1. Given the *outcome sentence* "The jeans are [new / old].", a worker gave the *context sentence*: "Ellie got the jeans from a [department / thrift] store."
::: **Explanation**: This satisfies a **one-to-one mapping** between the context and the outcome: "department" store would determine that the jeans are new, while "thrift" store would mean that the jeans are old. The **creative** choice of context sentence and pair of context words is a huge plus.
2. Given the outcome sentence "The shark is [aggressive / friendly].", a worker gave the context context sentence: "Jason [touched / escaped] the shark."
::: **Explanation:** This similarly satisfies a **one-to-one mapping**, and also earns the bonus score of being super **creative** (using verb pairs to determine the outcome characteristics).

Bad context sentence annotations

1. Given the *outcome sentence* "The spaghetti is [edible / inedible]", a worker gave the *context sentence*: "The spaghetti dish is [palatable / unpalatable]".
::: **Explanation:** This is bad for two reasons. First, we are looking for **diverse context annotations**, meaning you should not adhere to one format of context sentence or repeat the exact same structure of the outcome sentence. Second, the *pair of context words* ([palatable / unpalatable] in this case) **should not be synonymous** with the pair of *outcome words* ([edible / inedible] in this case).
2. Given the *outcome sentence* "The flashlight is [bright / broken]", a worker gave the *context sentence*: "Anna [turned on / threw away] the flashlight".
::: **Explanation**: While this is a perfect one-to-one mapping between the context and outcome sentences, we require each of the context words to be **exactly one word in length** (turned on and threw away are both two words in length).

Figure 3: **Top Left**: Exception Association Collection - Instructions. **Top Right**: Exception Association Collection - Examples. **Bottom Left**: Premise Collection - Instructions. **Bottom Right**: Premise Collection - Examples.



In this task, you are asked to evaluate a question in a multiple choice quiz. You are asked to give four evaluations:

(1, 2) Are the word pairs synonymous: Examples of word pairs that would earn a "Yes" or "No" answers are:
- **"Yes"**: [edible / palatable], [inedible / unpalatable], [open / ajar], [closed / shut]
- **" No ":** [aggressive / escaped], [friendly / touched], [good / cheese], [rotten / bacteria]

In general, select "Yes" if the words in a pair are identical / almost identical in meaning.

(3, 4) Can <Word> be an answer to the blank? We trust your intuition to respond to this question. However, we do monitor the quality of the annotations, and may reject your work if you answer "Yes" while the answer is clearly "No" and vice versa.

Are these words synonymous?
[tall / reach]
○ Yes
○ No

Are these words synonymous?
[short / ankles]
○ Yes
○ No

Given the prompt:
The goalpost was barely above Bills reach. The goalpost is ___.
Can short be a sensible answer to the blank?
○ Yes
○ No

Given the prompt:
The goalpost was barely above Bills ankles. The goalpost is ___.
Can tall be a sensible answer to the blank?
○ Yes
○ No

Given two choices to fill out a statement with a blank, please select the one to fill out the prompt that makes the most sense / seems significantly more likely to you.
Some examples:
1) **Prompt:** "The apple is rotten. The apple is _____" , **Choices:** (a) edible, (b) inedible , **Selection:** (b) inedible
2) **Prompt:** "Ellie got the jeans from a thrift store. The jeans are _____" , **Choices:** (a) new, (b) old , **Selection:** (b) old
3) **Prompt:** "Jason escaped the shark. The shark is _____" , **Choices:** (a) aggressive, (b) friendly , **Selection:** (a) aggressive

Given the prompt:
Logan could see his friend through the veil of smoke outside. The veil is ____.

Please select the best answer to fill in the blank.
○ thin
○ thick

Figure 4: **Top Left**: Challenge Set Validation - Instructions. **Top Right**: Challenge Set Validation - Task Sample. **Bottom Left**: Human Performance Collection - Instruction and Examples. **Bottom Right**: Human Performance Collection - Task Sample.