Unnatural Language Inference

Koustuv Sinha [1,2,3], Prasanna Parthasarathi [1,2], Joelle Pineau [1,2,3] and Adina Williams [3]
[1] School of Computer Science, McGill University, Canada
[2] Montreal Institute of Learning Algorithms (Mila), Canada
[3] Facebook AI Research (FAIR)
{koustuv.sinha, prasanna.parthasarathi, jpineau, adinawilliams}
@{mail.mcgill.ca, mail.mcgill.ca, cs.mcgill.ca, fb.com}

Abstract

Natural Language Understanding has witnessed a watershed moment with the introduction of large pre-trained Transformer networks. These models achieve state-of-the-art on various tasks, notably including Natural Language Inference (NLI). Many studies have shown that the large representation space imbibed by the models encodes some syntactic and semantic information. However, to really "know syntax", a model must recognize when its input violates syntactic rules and calculate inferences accordingly. In this work, we find that state-of-the-art NLI models, such as RoBERTa and BART are invariant to, and sometimes even perform better on, examples with randomly reordered words. With iterative search, we are able to construct randomized versions of NLI test sets, which contain permuted hypothesis-premise pairs with the same words as the original, yet are classified with perfect accuracy by large pre-trained models, as well as pre-Transformer state-of-the-art encoders. We find the issue to be language and model invariant, and hence investigate the root cause and thereby propose a simple training methodology to partially alleviate this effect. This finding calls into question the idea that our natural language understanding models, and the tasks used for measuring their progress, genuinely require a human-like understanding of syntax.

1 Introduction

Of late, large scale pre-trained Transformer-based (Vaswani et al., 2017) models—such as RoBERTa (Liu et al., 2019), BART (Lewis et al., 2020), and GPT-2 and -3 (Radford et al., 2019; Brown et al., 2020)—have exceeded recurrent neural networks' performance on many NLU tasks (Wang et al., 2018, 2019). In particular, several papers have even suggested that Transformers pretrained on a language modeling (LM) objective capture syntactic information (Goldberg, 2019; Hewitt and Manning, 2019), at least to some reasonable extent (Warstadt et al., 2019), and have shown that their self-attention layers are capable of surprisingly effective learning mechanisms (Rogers et al., 2020). In this work, we raise questions about claims that current models are able to "understand syntax".

Gold Label	Premise	Hypothesis
E	Boats in daily use lie within feet of the fashionable bars and restaurants.	There are boats close to bars and restaurants.
E	restaurants and use feet of fashionable lie the in Boats within bars daily .	bars restaurants are There and to close boats .
C	He and his associates weren't operating at the level of metaphor.	He and his associates were operating at the level of the metaphor.
C	his at and metaphor the of were He operating associates n't level .	his the and metaphor level the were He at associates operating of .

Table 1: Examples from MNLI Matched development set. Both the original and the example with permuted sentences elicits the same classification label (Entailment and Contradiction) from RoBERTa (large). We provide a simple demo of this behaviour in the associated Google Colab notebook.

Since there are many ways to investigate "syntax", we must be clear on what we mean by the term. Clearly, "language is not merely a bag of words" (Harris, 1954, p.156). A natural and common perspective from many formal theories of linguistics (e.g., Chomsky 1995) is that knowing a natural language requires that you know the syntax of that language. Knowing the syntax of a sentence means being sensitive to (at least) the *order of the words* in that sentence (potentially among other things). For example, it is well known that humans exhibit a *"sentence superiority effect"* (Cattell, 1886; Scheerer, 1981)—it is easier for

us to identify or recall words presented in canonical orders than in disordered, ungrammatical sentences (Toyota 2001; Baddeley et al. 2009; Snell and Grainger 2017; Wen et al. 2019, i.a.). In addition, understanding the syntax of a sentence is often taken to be a prerequisite for knowing what that sentence means (Heim and Kratzer, 1998). If performing an NLU task actually requires a humanlike understanding of sentence meaning, and of syntax as we've defined it, then we can ask: are NLU models sensitive to corrupted word order?

To investigate this, we focus on textual entailment, one of the hallmark tasks used to measure the linguistic reasoning capacity of Natural Language Understanding (NLU) models (Condoravdi et al., 2003; Dagan et al., 2005). This task, often also called Natural Language Inference (NLI; Bowman et al. 2015, i.a.) typically consists of two sentences, a premise and a hypothesis, and the objective of the (machine) learner is to predict whether the hypothesis: entails the premise, contradicts it, or is neutral with respect to it. We perform a battery of tests on models trained on NLI (transformers pre-trained on LM, a CNN and an RNN) where we permute the original *word order* of examples such that no word is present in its original position, and the relative word ordering is minimized (see Table 1).

We find, albeit surprisingly, that for nearly all premise-hypothesis pairs **there exists at least one permutation** that fools the models into providing the correct prediction. We verify our findings with a range of English NLI datasets, including SNLI (Bowman et al., 2015), MultiNLI (Williams et al., 2018) and ANLI (Nie et al., 2020). We observe similar results in Original Chinese NLI corpus (OCNLI; Hu et al. 2020a), which makes it unlikely that our findings are purely a consequence of English language. Our main contributions are:

- We propose a metric suite (Permutation Acceptance) for evaluating NLU models' insensitivity to unnatural word orders. (§3)
- We construct permuted sets for multiple test dataset splits to investigate a statistical view of Permutation Acceptance, and measure NLI model performance along with nature of predictions on permuted sentences via several large scale tests (§5).
- We perform an initial attempt to mitigate such issues in NLI models by devising a simple maximum entropy based method (§7).
- We show that NLI models focus on words rather than word order. Although they seem to be able to partially reconstruct some syntax from permuted examples (§6), as we explore with word-overlap and Parts-of-Speech overlap metrics.
- Finally, we provide initial evidence that humans struggle to perform unnatural language inference (§8).

2 Related Work

Researchers in NLP have studied syntactic structure in neural networks going back to Tabor (1994). Anecdotally, anyone who has interacted with large generative language models like GPT-2 or -3 will have marveled at their human-like ability to generate fluent and grammatical text (Goldberg, 2019; Wolf, 2019). When researchers have attempted to peek inside transformer LM's pretrained representations, often familiar syntactic representations (Hewitt and Manning, 2019), or a familiar order of linguistic operations (Tenney et al., 2019) appears.

There is also evidence, notably from agreement attraction phenomena (Linzen et al., 2016), that transformer-based models pretrained on an LM objective do acquire an understanding of natural language syntax (Gulordava et al., 2018; Chrupała and Alishahi, 2019; Jawahar et al., 2019; Lin et al., 2019; Manning et al., 2020; Hawkins et al., 2020; Linzen and Baroni, 2021). The claim that LMs acquire some syntactic knowledge has been made not only transformers, but also for convolutional neural nets (Bernardy and Lappin, 2017) and RNNs (Gulordava et al., 2018; van Schijndel and Linzen, 2018; Wilcox et al., 2018; Zhang and Bowman, 2018; Prasad et al., 2019; Ravfogel et al., 2019)—although there are many caveats (Ravfogel et al., 2018; White et al., 2018; Davis and van Schijndel, 2020; Chaves, 2020; Da Costa and Chaves, 2020; Kodner and Gupta, 2020).

Several works have debated the extent to which NLI models in particular know syntax (although each work adopts a slightly different idea of what "knowing syntax" entails). For example, McCoy et al. (2019) argued that the knowledge acquired by models trained on NLI (for at least some popular datasets) is actually not as syntactically sophisticated as it might have initially seemed; some transformer models rely mainly on simpler, non-humanlike heuristics. In general, transformer LM performance has been found to be patchy and

variable across linguistic phenomena (Dasgupta et al., 2018; Naik et al., 2018; An et al., 2019; Ravichander et al., 2019; Jeretic et al., 2020). This is especially true for syntactic phenomena (**?**Marvin and Linzen, 2018; Hu et al., 2020b; Gauthier et al., 2020; McCoy et al., 2020), where transformers are, for some phenomena and settings, worse than RNNs (van Schijndel et al., 2019). From another angle, many have explored architectural approaches for increasing a network's sensitivity to syntactic structure (Chen et al., 2017; Li et al., 2020). **?** showed that models that learn jointly to perform NLI well and to parse do not generate parse trees that match popular syntactic formalisms. Furthermore, models trained explicitly to differentiate acceptable from unacceptable sentences (i.e., one of the most common syntactic tests used by linguists) have, to date, come nowhere near human performance (**?**).

Additionally, NLI models often over-attend to particular words to predict the correct answer (Gururangan et al., 2018; Clark et al., 2019). Wallace et al. (2019) show that some short sequences of non-human-readable text can fool many NLU models, including NLI models trained on SNLI, into predicting a specific label. In fact, Ettinger (2020) observed that for one of three test sets, BERT loses some accuracy in word-perturbed sentences, but that there exists a subset of examples for which BERT's accuracy remains intact. This led Ettinger to speculate that "some of BERT's success on these items may be attributable to simpler lexical or n-gram information".

Thus, it is reasonable to wonder whether models can perform equally well on permuted sentences since the model is able to view the same collection of words. If this is the case, it suggests that these state-of-the-art models actually perform as bag-of-words models (Blei et al., 2003; Mikolov et al., 2013), i.e. they little focus syntax. This being said, there are empirical reasons (in addition to the theoretical ones describe above) to expect that a knowledge of syntax is crucial for performing NLI. An early human annotation effort on top of the PASCAL RTE dataset (Dagan et al., 2006) discovered that "syntactic information alone is sufficient to make a judgment" on one third of the textual entailment examples in RTE, whereas almost a half could be solved if additionally provided a thesaurus (Vanderwende and Dolan, 2005).

3 Syntactic Permutation Acceptance

As we mentioned, many formal linguistic theories take syntactic structure to be necessary for humans to determine the meaning of sentences. NLP as a field has found NLI to be very promising, in part because it is rooted in the tradition of logical entailment, and adheres to the spirit of many propositional logics based on the truth-conditional perspective (Frege, 1948; Montague, 1970; Chierchia and McConnell-Ginet, 1990; Heim and Kratzer, 1998). The randomized sentences we have are so mixed up as to be ungrammatical, and, if we adhere to such a truth conditional perspective (Montague, 1970), uninterpretable. Put another way, the meanings of highly permuted sentences (if they exist) are not propositions, and then can't have their truth conditions checked. If sentences have no truth conditions, then we cannot tell if they entail each other or not, since this requires knowing what the actual conditions in the world would be under which the sentences would be (judged) true. In short, we shouldn't expect the task of textual entailment to be defined at all in our case.

For our purposes, we hypothesize that a syntax-aware model might perform NLI in one of a few ways. First, if for every example in the dataset all of its permuted counterparts were hopelessly mixed up, the model might assign near zero probability mass on the gold label (effectively recognizing the ungrammaticality). In this somewhat extreme case, no examples would have any of their permuted counterparts *accepted* (i.e., assigned the original example's gold label) by the model. Second, permuted examples might instead baffle the model into assigning equal probability mass on all three labels, and effectively choosing a label at random (this is inconsistent with our findings). Third, the model might just be unable to interpret permuted sentences at all. In that case, it might either just always assign a particular label (e.g., neutral) or assign a non-entailment label (along the lines of 2-class textual entailment datasets like RTE (Dagan et al., 2005)); neither option is consistent with what we find. However, if the model doesn't care about word order and operates as a bag-of-words, it would accept permuted examples at a high rate.

We show that all investigated models do not care about word order according to a suite of **Permutation Acceptance** metrics we define to quantify how many the permuted sentences are accepted, i.e. are assigned the gold label, by a model. In light of

the "sentence superiority" findings, if the syntax is corrupted in a way such that the resulting sentences are ungrammatical (and largely meaningless), we would predict that humans (and human-like NLU models) would score very low Permutation Acceptance scores (owing to low or no permuted examples being assigned the gold label). We show that humans indeed struggle with unnatural language inference data in §8.

4 Methods

Constructing the permuted dataset. Based on our notion of Permutation Acceptance, we devise a series of experiments using trained models on various NLI datasets. Concretely, for a given dataset D having splits D_{train} and D_{test}, we train an NLI model M first on D_{train} that achieves performance comparable to that was reported reported in the original papers. We then construct a randomized version of D_{test}, which we term as $\hat{D}_{\text{test}}$ such that: for each example $(p_i, h_i, y_i) \in D_{\text{test}}$ (where p_i and h_i are the premise and hypothesis sentences of the example respectively and y_i is the gold label), we use a permutation operator $\mathcal{F}$ that returns a list $(\hat{P}_i, \hat{H}_i)$ of n permuted sentences ($\hat{p}_i$ and $\hat{h}_i$), where n is a hyperparameter. $\mathcal{F}$ essentially permutes all positions of the words in a given sentence (i.e., either in premise or hypothesis) with the restriction that *no words are in their original position*. In our initial setting, we do not explicitly control the placement of the words relative to their original neighbors, but we analyze such clumping of words effect in §5.

Thus, $\hat{D}_{\text{test}}$ now consists of $|D_{\text{test}}| \times n$ examples, with n different permutations of hypothesis and premise for each original test example pair. If a sentence S contains w words, then the total number of available permutations are $(w-1)!$, thus making the output of $\mathcal{F}$ a list of $\binom{(w-1)!}{n}$ permutations.

Defining Permutation Acceptance. The choice of n naturally allows us to analyze a statistical view of the predictability of a model on the permuted sentences. To that end, we define the following notational conventions. Let $\mathcal{A}$ be the original accuracy of a given model M on a dataset D, and c be the number of examples in a dataset which are marked as correct as per the above formulation of standard dataset accuracy. Typically $\mathcal{A}$ is given by $\frac{c}{|D_{test}|}$ or $\frac{c}{|D_{dev}|}$, where c is the number of examples which are predicted correctly compared to ground truth.

Let $M(\hat{P}_i, \hat{H}_i)_{\text{cor}}$ be the percentage of n permutations of an example (p_i, h_i) deemed correct (i.e. assigned the ground truth label) by the model M:

$$M(\hat{P}_i, \hat{H}_i)_{\text{cor}} = \frac{1}{n} \sum_{(\hat{p}_j \in \hat{P}_i, \hat{h}_j \in \hat{H}_i)} ((M(\hat{p}_j, \hat{h}_j) = y_i) \rightarrow 1) \quad (1)$$

Let Ω_x be the percentage of examples $(p_i, h_i) \in D_{\text{test}}$ for which $M(\hat{P}_i, \hat{H}_i)_{\text{cor}}$ exceeds a threshold $0 < x < 1$. Concretely, a given p_i and h_i will count as being predicted correctly according to Ω_x if more than x percent of its permutations ($\hat{P}_i$ and $\hat{H}_i$) are assigned the gold label y_i by the model M. Mathematically,

$$\Omega_x = \frac{1}{|\ D_{test}\ |} \sum_{(p_i, h_i) \in D_{test}} ((M(\hat{P}_i, \hat{H}_i)_{\text{cor}} > x) \rightarrow 1). \quad (2)$$

There are two specific cases of the Ω_{x} that we are most interested in. First, we define Ω_{max} or the *Maximum Accuracy*, where $x = 1/|D_{\text{test}}|$. In short, Ω_{max} gives the percentage of examples $(p_i, h_i) \in D_{\text{test}}$ for which *at least one* of $(\hat{p_j}, \hat{h_j})$ model M assigns the gold label y_i. Second, we define Ω_{rand}, or *random baseline accuracy*, where $x = 1/3$ or chance probability (for balanced 3-way classification). This metric is less stringent than Ω_{max}, and provides a lower-bound relaxation.

We also define D^f to be the list of examples originally marked incorrect according to $\mathcal{A}$, but are now deemed correct according Ω_{max}. D^c is the list of examples originally marked correct according to $\mathcal{A}$. Thus, $D^f < D^c$ necessarily. Additionally, we define $\mathcal{P}^c$ and $\mathcal{P}^f$, ranging from 0 to 1, as the dataset average percentage of permuted examples deemed correct, when the examples were originally correct (D^c) and when the examples were originally incorrect (D^f) as per M (hence, flipped).

$$\mathcal{P}^c = \frac{1}{|D^c|} \sum_{i=0}^{|D^c|} M(\hat{P}_i, \hat{H}_i)_{\text{cor}}, \qquad \mathcal{P}^f = \frac{1}{|D^f|} \sum_{i=0}^{|D^f|} M(\hat{P}_i, \hat{H}_i)_{\text{cor}} \quad (3)$$

Since the permutation function $\mathcal{F}$ results in ungrammatical, nonsensical sentences (Table 1), we have two hypotheses given our discussion so far:

Model	Eval Dataset	$\mathcal{A}$	Ω_{max}	$\mathcal{P}^c$	$\mathcal{P}^f$	Ω_{rand}
RoBERTa (large)	MNLI_m_dev	0.906	0.987	0.707	0.383	0.794
	MNLI_mm_dev	0.901	0.987	0.707	0.387	0.790
	SNLI_dev	0.879	0.988	0.768	0.393	0.826
	SNLI_test	0.883	0.988	0.760	0.407	0.828
	A1_dev	0.456	0.897	0.392	0.286	0.364
	A2_dev	0.271	0.889	0.465	0.292	0.359
	A3_dev	0.268	0.902	0.480	0.308	0.397
	Mean	0.652	0.948	0.611	0.351	0.623
	Harmonic Mean	0.497	0.946	0.572	0.344	0.539
BART (large)	MNLI_m_dev	0.902	0.989	0.689	0.393	0.784
	MNLI_mm_dev	0.900	0.986	0.695	0.399	0.788
	SNLI_dev	0.886	0.991	0.762	0.363	0.834
	SNLI_test	0.888	0.990	0.762	0.370	0.836
	A1_dev	0.455	0.894	0.379	0.295	0.374
	A2_dev	0.316	0.887	0.428	0.303	0.397
	A3_dev	0.327	0.931	0.428	0.333	0.424
	Mean	**0.668**	**0.953**	0.592	**0.351**	**0.634**
	Harmonic Mean	**0.543**	**0.951**	0.546	**0.347**	**0.561**
DistilBERT	MNLI_m_dev	0.800	0.968	0.775	0.343	0.779
	MNLI_mm_dev	0.811	0.968	0.775	0.346	0.786
	SNLI_dev	0.732	0.956	0.767	0.307	0.731
	SNLI_test	0.738	0.950	0.770	0.312	0.725
	A1_dev	0.251	0.750	0.511	0.267	0.300
	A2_dev	0.300	0.760	0.619	0.265	0.343
	A3_dev	0.312	0.830	0.559	0.259	0.363
	Mean	0.564	0.883	**0.682**	0.3	0.575
	Harmonic Mean	0.445	0.873	**0.664**	0.296	0.49

Table 2: Statistics for Transformer based models. All models are trained on MNLI corpus (Williams et al., 2018). Ω_{max} or Max Accuracy is computed if *any* of the $n = 100$ permutations per data point yield correct results. $\mathcal{P}^f$ stands for the mean number of permutations which were correct when the original prediction is correct. $\mathcal{P}^f$ stats for the mean number of permutations which are correct when the original prediction is incorrect (flip). Ω_{rand} is computed as the percentage of data points the ground truth label is chosen over a random uniform baseline (1/3). Bold marks the highest value per metric (**red** shows the model is insensitive to permutation).

Model	Eval Dataset	$\mathcal{A}$	Ω_{max}	$\mathcal{P}^c$	$\mathcal{P}^f$	Ω_{rand}
InferSent	MNLI_m_dev	0.658	0.904	0.842	0.359	0.712
	MNLI_mm_dev	0.669	0.905	0.844	0.368	0.723
	SNLI_dev	0.556	0.820	0.821	0.323	0.587
	SNLI_test	0.560	0.826	0.824	0.321	0.600
	A1_dev	0.316	0.669	0.425	0.395	0.313
	A2_dev	0.310	0.662	0.689	0.249	0.330
	A3_dev	0.300	0.677	0.675	0.236	0.332
	Mean	**0.481**	0.78	0.731	**0.322**	0.514
	Harmonic Mean	0.429	0.767	0.694	**0.311**	0.455
ConvNet	MNLI_m_dev	0.631	0.926	0.773	0.340	0.684
	MNLI_mm_dev	0.640	0.926	0.782	0.343	0.694
	SNLI_dev	0.506	0.819	0.813	0.339	0.597
	SNLI_test	0.501	0.821	0.809	0.341	0.596
	A1_dev	0.271	0.708	0.648	0.218	0.316
	A2_dev	0.307	0.725	0.703	0.224	0.356
	A3_dev	0.306	0.798	0.688	0.234	0.388
	Mean	0.452	**0.817**	**0.745**	0.291	0.519
	Harmonic Mean	0.404	**0.81**	**0.74**	0.279	**0.473**
BiLSTM	MNLI_m_dev	0.662	0.925	0.800	0.351	0.711
	MNLI_mm_dev	0.681	0.924	0.809	0.344	0.724
	SNLI_dev	0.547	0.860	0.762	0.351	0.598
	SNLI_test	0.552	0.862	0.771	0.363	0.607
	A1_dev	0.262	0.671	0.648	0.271	0.340
	A2_dev	0.297	0.728	0.672	0.209	0.328
	A3_dev	0.304	0.731	0.656	0.219	0.331
	Mean	0.472	0.814	0.731	0.301	**0.52**
	Harmonic Mean	0.41	0.803	0.725	0.287	0.463

Table 3: Statistics for Non-Transformer Models. All models are trained on MNLI corpus (Williams et al., 2018). Ω_{max} or Max accuracy is computed if *any* of the $n = 100$ permutations per data point yield correct results. $\mathcal{P}^c$ stands for the mean number of permutations which were correct when the original prediction is correct. $\mathcal{P}^f$ stats for the mean number of permutations which are correct when the original prediction is incorrect (flip). Ω_{rand} is computed as the percentage of data points the ground truth label is chosen over a random uniform baseline (1/3). Bold marks the highest value per metric (**red** shows the model is insensitive to permutation).

(a) permutations resulting from permutation operator $\mathcal{F}$ will elicit random outcomes, with $\mathcal{P}^c$ and $\mathcal{P}^f$ being in random uniform probability of $1/n$, **(b)** permutations from $\mathcal{F}$ will receive incorrect predictions, resulting in $\mathcal{P}^c = 0$ and $\mathcal{P}^f = 0$. However, we neither observe **(a)** nor **(b)** with the state-of-the-art models.

5 Results

We present results for two types of models: **(a)** Transformer-based models and **(b)** Non-Transformer Models. In Transformer-based models, we investigate the state-of-the-art pre-trained models such as RoBertA (large) (Liu et al., 2019), BART (large) (Lewis et al., 2020) as well as a relatively small DistilBERT model (Sanh et al., 2020). For **(b)** we consider several pre-Transformer era recurrent and convolution based neural networks, such as InferSent (Conneau et al., 2017), Bidirectional LSTM (Collobert and Weston, 2008) and ConvNet (Zhao et al., 2015). We train all models on MNLI (Williams et al., 2018), and evaluate on in-distribution (SNLI (Bowman et al., 2015) and MNLI) and out-of-distribution datasets (ANLI (Nie et al., 2020)). We independently verified Transformer-based results on our trained model using HuggingFace Transformers (Wolf et al., 2020), as well as pre-trained checkpoints from FairSeq (Ott et al., 2019) using PyTorch Model Hub. For Non-Transformer models, we use the codebase from Conneau et al. (2017). We use $n = 100$, and use 100 seeds for the randomizations for each example in D_{test} to ensure full reproducibility. We drop examples from test sets where we are unable

Model	$\mathcal{A}$	Ω_{max}	$\mathcal{P}^c$	$\mathcal{P}^f$	Ω_{rand}
RoBERTa (large)	**0.784**	**0.988**	0.726	**0.339**	**0.773**
InferSent	0.573	0.931	0.771	0.265	0.615
ConvNet	0.407	0.752	**0.808**	0.199	0.426
BiLSTM	0.566	0.963	0.701	0.271	0.611

Table 4: Results on evaluation on OCLI Dev set. All models are trained on OCNLI corpus (Hu et al., 2020a). Max accuracy (Ω_{max}) is computed based on whether *any* of the $n = 100$ permutations per data point yield correct results. $\mathcal{P}^c$ stands for the mean number of permutations which were correct when the original prediction is correct. $\mathcal{P}^f$ stats for the mean number of permutations which are correct when the original prediction is incorrect (flip). Bold marks the highest value per metric (**red** shows the model is insensitive to permutation).

to compute *all unique* randomizations, typically these are examples with sentences of length of less than 6 tokens. Code, randomized data, and model checkpoints will be released publicly.

Models accept many permuted examples. We find Ω_{max} on models trained on MNLI and evaluation on MNLI (in-domain generalization) is significantly high: **98.7%** on MNLI dev and test sets. This shows there exists at least one permutation for almost all examples in D_{test} such that the model M predicts the correct answer. We also observe significantly high Ω_{rand} at 79.4%, suggesting the models outdo even a random baseline in accepting permuted, ungrammatical sentences.

Furthermore, we observe similar effects of Ω_{max} in out-of-domain generalization on evaluating with ANLI dataset splits, where Ω_{max} is signficantly higher than $\mathcal{A}$. As a consequence, we encounter many *flips*, where the example was originally predicted incorrectly by the model but at least one permutation for that example elicits the correct response. However, recall this analysis expects us to know the correct label upfront, so this test can be thought of as running a syntax-based stress test on the model until we reach the correct answer (or give up by exhausting our set of permutations, n).

In the case of out-of-domain generalization, Ω_{rand} reduces considerably. Probability of permuted sentences to be predicted correctly also is significantly higher for examples which were predicted correctly ($\mathcal{P}^c >> \mathcal{P}^f$ for all test splits). These two results suggest that the investigated NLU models are acting as bag-of-words models, where it is harder to find the correct permutation for already misclassified, non-generalizable sentences.

Models are very confident. The phenomenon we observe would be of less concern if the correct label prediction was just an outcome of chance, which could occur when the entropy of the log probabilities of the model output is high (suggesting uniform probabilities on entailment, neutral and contradiction labels). We first investigate the model probabilities for the Transformer-based models on the permutations that lead to the correct answer in Figure 1. We find overwhelming evidence that model confidences on in-distribution datasets (MNLI, SNLI) are highly skewed, resulting in low entropy, and it varies among different model types. BART proves to be the most skewed among the three types of model we consider.

To investigate whether the skewedness is a function of model capacity, we investigate the log probabilities of a lower capacity model DistilBERT on random perturbations, and find it is very similar to the RoBERTa (large) model, although, DistilBERT does exhibit lower $\mathcal{A}$, Ω_{max}, and Ω_{rand}. This suggesting its inability to understand certain premise-hypothesis pairs completely due to lack of capacity.

For non-Transformers whose accuracy $\mathcal{A}$ is lower, we observe the relative performance in the terms of Ω_{max} (Table 3) and Average entropy (Figure 1) for these classes of models. As expected, since the non-Transformer models are significantly worse, the Ω_{max} achieved by these models are also lower. However, while comparing the averaged entropy of the model predictions, it is clear that there is some benefit to being a worse model—the models are not as overconfident on randomized sentences as they are Transformers.

Similar artifacts in Chinese NLU. To conclusively verify the observation in NLI, we extended the experiments to Original Chinese NLI dataset (Hu et al., 2020a, OCNLI). We re-use pretrained RoBERTa (large) model and InferSent (non-Transformer) models on OCNLI. We find similar observations (Table 4), thereby conclusively proving that the phenomenon is not just an artifact of English language, but natural language understanding as a whole.

Other Results In addition to the results presented here, we investigate the effect of length (which correlates with number of possible permutations), and the effect of hypothesis only randomization. Results are presented in the Appendix Appendix A and Appendix C.

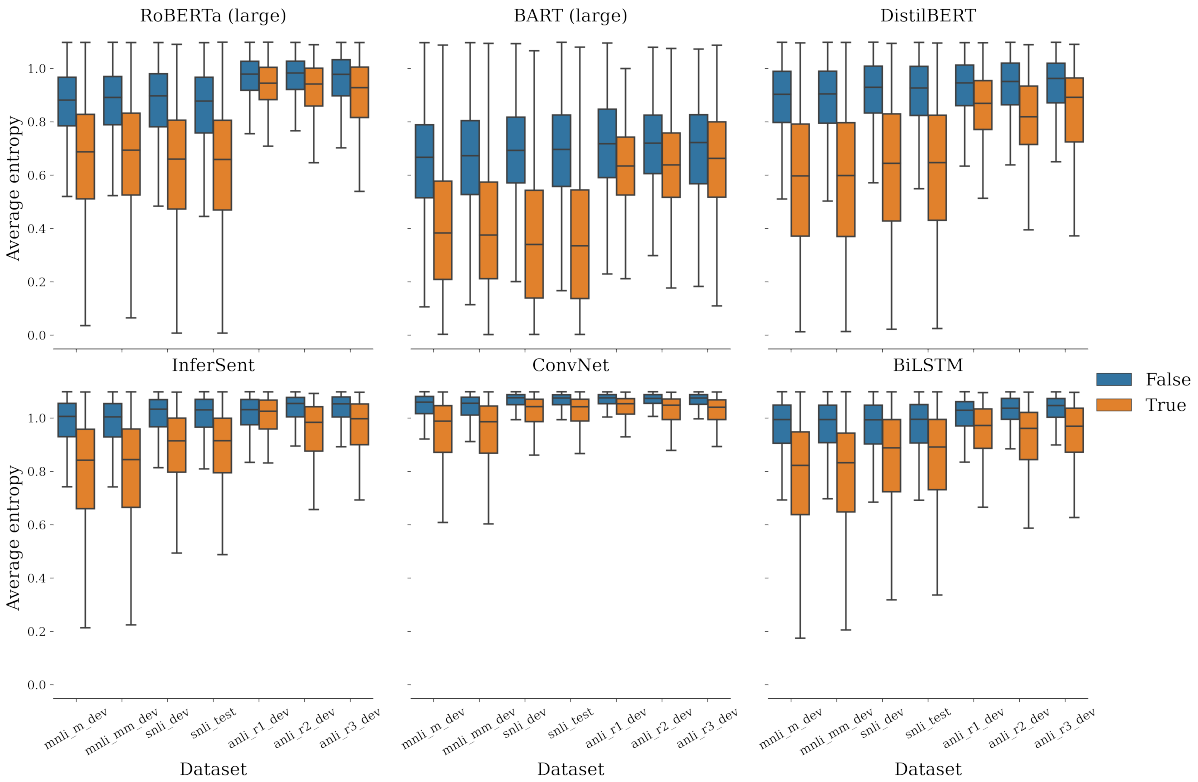


Figure 1: Average entropy of model confidences on permutations that yielded the correct results presented in a box plot, computed for Transformer-based models (top row) and Non-Transformer-based models (bottom row). Results are shown for D^c (True) and D^f (False) separately. The boxes signifies the quartiles of the entropy distributions.

6 Analyzing Syntactic Structure Associated with Tokens

Since all investigated models have relatively high permutation acceptance, we are lead to ask: what properties of particular permutations lead them to be accepted. We perform two initial analyses to shed light on this question. First, we ask to what extent preserving local word order despite permutation is correlated with higher Permutation Acceptancescores. We find that there is some correlation but it doesn't fully explain the high Permutation Acceptancescores. Second, we ask whether Ω is related to more abstract measure of local word relations, i.e., part-of-speech neighborhood. We find that there is little effect of POS-neighbors for non-Transformer models, but that RoBERTa, BART, and DistilBERT show a distict effect. Taken together these analyses suggest that some local word order information affects model's Permutation Acceptancescores, and perhaps incorporating methods of decreasing model reliance on this information could be fruitful.

Preserving Local Word Order Leads to Higher Permutation Acceptance. In our initial experiments, we randomized the word order of the sentences with the constraint that no word appear in its original position. This kind of randomization can preserve the relative positions of n-grams. To analyze the effect of this, we compute BLEU scores on 2-, 3- and 4 n-grams and compare the acceptability of the permuted sentences across different models. If preserved n-grams are driving the Permutation Acceptanceeffects, we should see a correlation between BLEU score and Ω. As a result of our permutation process, the maximum BLEU-3 and BLEU-4 scores are neglibly low (< 0.2 BLEU-3 and < 0.1 BLEU-4), already calling into question the hypothesis that n-grams are the sole explanation for our finding. Because of this, we only compare BLEU-2 scores. (Detailed experiments on specially constructed permutations that cover the entire range of BLEU-3 and BLEU-4 is provided in Appendix D) We find that the probability of a permuted sentence to be predicted correctly by the model correlates BLEU-2 score (Figure 2). However, the base prediction of Transformer-based models is still far from random (66% for BLEU-2 range of 0-0.15), and hence requires further investigation on the language processing mechanisms

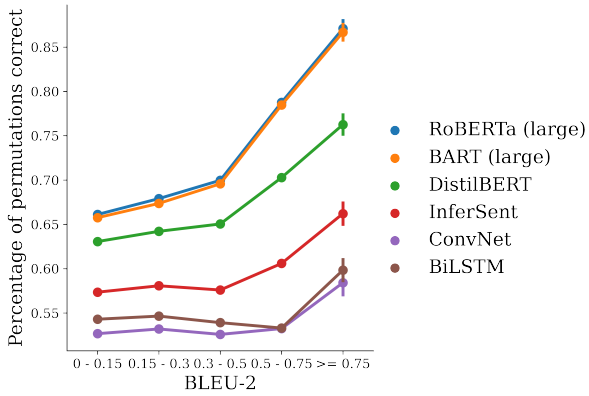


Figure 2: Relation of BLEU-2 score against the acceptability of permuted sentences across all test datasets on four models. We observe that performance of RoBERTa and BART are surprisingly similar and can be set apart considerably from the non-Transformer based models, such as InferSent and ConvNet.

employed by these models.

Part-of-speech neighborhood tracks Permutation Acceptance. Many syntactic formalisms, like Lexical Functional Grammar (Kaplan and Bresnan, 1995; Bresnan et al., 2015, LFG), Head-drive Phrase Structure Grammar (Pollard and Sag, 1994, HPSG) and Lexicalized Tree Adjoining Grammar (Schabes et al., 1988; Abeille, 1990, LTAG), are "lexicalized", i.e., individual words or morphemes to bear syntactic features telling you which other things they can combine with. Taking LTAG as an example, two lexicalized trees would be associated with the verb "buy", one which projects a phrase structure minitree containing two noun phrases (for the subject and direct object, as in *Kim bought a book*), and one which projects a minitree containing three (one for the subject, another for the direct object, and a third for the indirect object, as in *Kim bought Logan a book*). In this way, an average tree family for any particular word would provide information about what sort of syntactic contexts the word generally appears in, or in other words, what syntactic neighbors the word has. Taking inspiration from lexicalized grammatical formalisms, we speculate that our NLI models might be performing well on permuted examples because they are reconstructing perhaps noisily the word order of a sentence from its words.

To test this, we came up with a quick operationalization of the idea of a lexicalized minitree. First, we POS tagged every example in the corpus D_{train} using the 17 Universal Part-of-Speech tags (using spaCy Honnibal et al. 2020). For each $w_i \in D_{\text{train}}$, we compute the occurrence probability of POS tags on tokens in the *neighborhood* of w_i for each S_i in D_{train} containing w_i. The neighborhood is specified by the radius r (a symmetrical window r tokens from $w_i \in S_i \in D_{\text{train}}$ to the left and right). We denote this sentence level probability of POS tags for a word w_i as $\psi^r_{\{w_i,S_i\}} \in \mathcal{R}^{17}$ (see Figure 6). These sentence-level word POS neighbor scores can be averaged to get a corpus-level POS tag minitree probability $\psi^r_{\{w_i,D_{\text{train}}\}} \in \mathcal{R}^{17}, \forall w_i \in D_{\text{train}}$ (i.e., a type level score). Then, for a sentence $S_i \in D_{\text{test}}$, for each word $w_i \in S_i$, we compute a **POS minitree overlap score** $\beta^k_{\{w_i,S_i\}}$ as follows:

$$\beta^k_{\{w_i,S_i\}} = \frac{1}{k} \mid \text{argmax}_k \psi^r_{\{w_i,D_{\text{train}}\}} \cap \text{argmax}_k \psi^r_{\{w_i,S_i\}} \mid \quad (4)$$

Concretely, $\beta^k_{\{w_i,S_i\}}$ computes the overlap of top-k POS tags in the neighborhood of a word w_i with that of the train statistic. If a word exhibits the same mini-tree surrounding it in the sentence as that of the training set, then the overlap would be 1. For a given sentence S_i, the aggregate $\beta^k_{\{S_i\}}$ is defined by the average of the overlap scores of the consituent words: $\beta^k_{\{S_i\}} = \frac{1}{|S_i|} \sum_{w_i \in S_i} \beta^k_{\{w_i,S_i\}}$, and we call it a POS minitree *signature*.

Now, we can compute the same score for a permuted sentence $\hat{S}_i$ to have $\beta^k_{\{\hat{S}_i\}}$. The idea is if the permuted sentence POS signature comes close to the true sentence, then the ratio of $\beta^k_{\{\hat{S}_i\}}/\beta^k_{\{S_i\}}$ will be close to 1. If the ratio is > 1, that suggests the permuted sentence has more overlap than the original sentence based on the train statistic. Put in other words, if high overlap correlates with percentage of permutations deemed correct (even in randomized sentences), then our models treat words as if they bear syntactic minitrees. Therefore, for $\hat{D}_{\text{test}}$ where many of the n permutations have high average POS minitree overlap score, we should expect a higher prediction accuracy.

We investigate the relationship with percentage of permuted sentences accepted with $\beta^k_{\{\hat{S}_i\}}/\beta^k_{\{S_i\}}$ in Figure 3. We observe that the POS Tag Minitree hypothesis holds for Transformer-based models, RoBERTa, BART and DistilBERT, where the percentage of accepted pairs increase as the sentences have higher overlap with the un-permuted sentence in terms of POS signature. For non-Transformer

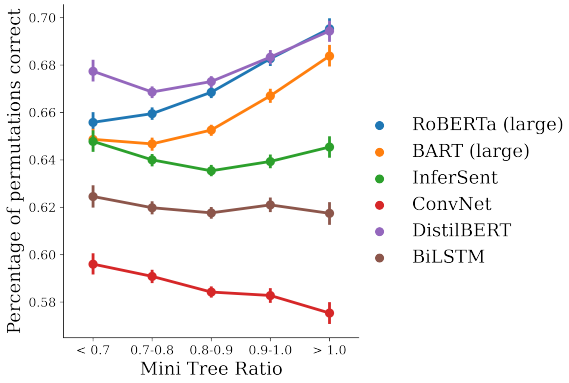


Figure 3: Comparison of POS Tag Mini Tree overlap score with the percentage of permutations deemed correct by the models.

models such as InferSent, ConvNet and BiLSTM models, the POS signature ratio to percentage of correct permutation remains the same or decreases, suggesting that the reasoning process employed by these models does not preserve local abtract syntax structure (i.e., POS neighbor relations).

7 Maximum Entropy Training

Here, we propose an initial attempt to mitigate the effect of correct prediction on permuted examples. We observed that the log probabilities of the output of a model on permuted examples are significantly higher than random. This kind of phenomenon has been observed prior in Computer Vision (Gandhi and Lake, 2019), and suggests model struggle to to learn *mutually exclusively*. Neural networks tend to output higher confidence than random for even unknown inputs, which might be an underlying cause of the high Permutation Acceptance phenomenon in our NLI models.

Eval Dataset	$\mathcal{A}$ (V)	$\mathcal{A}$ (ME)	$\Omega_{\max}$ (V)	$\Omega_{\max}$ (ME)
mnli_m_dev	0.905	0.908	0.984	0.328
mnli_mm_dev	0.901	0.903	0.985	0.329
snli_test	0.882	0.888	0.983	0.329
snli_dev	0.879	0.887	0.984	0.333
anli_r1_dev	0.456	0.470	0.890	0.333
anli_r2_dev	0.271	0.258	0.880	0.333
anli_r3_dev	0.268	0.243	0.892	0.334

Table 5: NLI Accuracy ($\mathcal{A}$) and Permutation Acceptancemetrics ($\Omega_{\max}$) of RoBERTa when trained on MNLI dataset using vanilla (V) and Maximum Random Entropy (ME) method

Since our ideal model would be ambivalent about the randomized ungrammatical sentences, we devise a simple objective to train NLU models by baking in the Mutual Exclusivity principle through Maximizing Entropy. Concretely, we train RoBERTa model to do well on the MNLI dataset while maximizing the entropy (**H**) on a subset of n randomized examples ($(\hat{p}_i, \hat{r}_i)$ for each example (p, h). We use a $n = 1$ randomizations for each example, and modify the loss function as follows:

$$\mathcal{L} = \underset{\theta}{\operatorname{argmin}} \sum_{((p,h),y)} y \log(p(y|(p,h);\theta)) + \sum_{i=1}^{n} \mathbf{H}\left(y|(\hat{p}_i, \hat{h}_i);\theta\right) \quad (5)$$

Using this simple maximum entropy method, we find that the model improves considerably with respect to its robustness to randomized sentences (Figure 4), all without taking a hit in the model accuracy (Table Table 5). We observe none of the models reach $\Omega_{\max}$ score close to 0, suggesting further room to explore other methods for decreasing models' Permutation Acceptance.

Evaluator	Accuracy	Macro F1	Acc on D^c	Acc on D^f
X	0.581 ±0.068	0.454	0.649 ±0.102	0.515 ±0.089
Y	0.378 ±0.064	0.378	0.411 ±0.098	0.349 ±0.087

Table 6: Human evaluation on 200 permuted sentence pairs from MNLI Matched development set (Williams et al., 2018) using two NLI experts. Half of the permuted pairs contained shorter sentences and the other, longer ones. Both experts were provided only the permuted sentences (not the original example or the label) and were disallowed from consulting with one another. All permuted sentences were predicted correctly by RoBERTa (large).

8 Human Evaluation

Since our models often accept permuted sentences, we ask how humans perform unnatural language inference on permuted sentences. We expect humans to struggle with the task, given our intuitions and the sentence superiority findings, but to test this, we presented two experts in Natural Language Inference (one a linguist) with a random sample of 200 permuted sentence pairs, and asked them to predict the entailment relation. The experts were provided with no information about the examples from which the permutations were drawn (above and beyond the common knowledge that NLI is usually defined as a roughly balanced 3-way classification task). Unbeknownst to the experts, all permuted sentences in the sample were actually deemed to

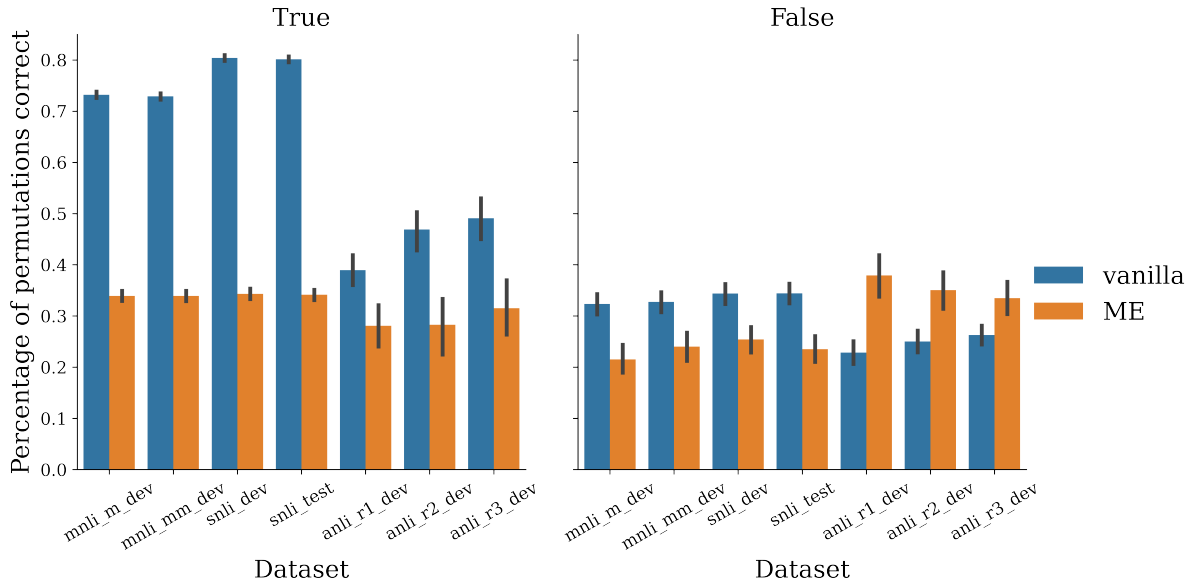


Figure 4: Effect of maximizing entropy training on RoBERTa (large)

be accepted by RoBERTa model (trained on MNLI dataset). We observe that the experts performed much worse than RoBERTa (Table 6), although their accuracy was a bit higher than random. In a second sample, and again, unbeknownst to the experts, we provided permuted sentence pairs from the MNLI Matched Dev. set some of which were originally predicted correctly and some incorrectly by RoBERTa (large) model. We find that for both experts, accuracy on permutations from originally correct examples was higher than on the incorrect examples, which verifies the Dasgupta et al. (2018); Gururangan et al. (2018); Naik et al. (2019) that word overlap is important.

9 Future Work & Conclusion

While we have shown that classification labels can be flipped based solely on a sentence reordering, future work could also explore the relationship between permutation and deletion. Although our results tentatively support the hypothesis that current models do not know "know syntax" in a human-like way (according to our definition) and are mostly just sensitive to words (or perhaps n-grams), they are preliminary, and future work is required to fully understand human-like classification of permuted NLI examples.

In this work we show that state-of-the-art models does not rely on sentence structure the way we think they should. On the task of Natural Language Inference, we show that models (both Transformer-based models, RNNs, and ConvNets) are largely insensitive to permutations of word order that corrupt the original syntax. This raises questions about the extent to which such systems understand "syntax", and highlights the unnatural language understanding processes they employ.

A few years ago, Manning (2015) encouraged NLP to consider "the details of human language, how it is learned, processed, and how it changes, rather than just chasing state-of-the-art numbers on a benchmark task." We expand upon this view, and suggest one particular future direction: we should train models not only to do well, but also to not to overgeneralize to corrupted input.

Acknowledgments

Thanks to Shagun Sodhani, Roy Schwartz, Emily Dinan, Hagen Blix, Ryan Cotterell, Nikita Nangia, and Grusha Prasad for many invaluable comments and feedback on the draft.

References

Anne Abeille. 1990. Lexical and syntactic rules in a Tree Adjoining Grammar. In *28th Annual Meeting of the Association for Computational Linguistics*, pages 292–298, Pittsburgh, Pennsylvania, USA. Association for Computational Linguistics.

Aixiu An, Peng Qian, Ethan Wilcox, and Roger Levy. 2019. Representation of constituents in neural language models: Coordination phrase as a case study. 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 2888–2899, Hong Kong, China. Association for Computational Linguistics.

Alan D Baddeley, GRAHAM J Hitch, and RICHARD J Allen. 2009. Working memory and binding in sentence recall. *Journal of Memory and Language*.

Jean-Phillipe Bernardy and Shalom Lappin. 2017. Using deep neural networks to learn syntactic agreement. In *Linguistic Issues in Language Technology, Volume 15, 2017*. CSLI Publications.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *JMLR*.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Joan Bresnan, Ash Asudeh, Ida Toivonen, and Stephen Wechsler. 2015. *Lexical-functional syntax*. John Wiley & Sons.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv*.

James McKeen Cattell. 1886. The time it takes to see and name objects. *Mind*.

Rui Chaves. 2020. What don't RNN language models learn about filler-gap dependencies? In *Proceedings of the Society for Computation in Linguistics 2020*, pages 1–11, New York, New York. Association for Computational Linguistics.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017. Enhanced LSTM for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1657–1668, Vancouver, Canada. Association for Computational Linguistics.

Gennaro Chierchia and Sally McConnell-Ginet. 1990. *Meaning and grammar: An Introduction to Semantics*. Cambridge, Ma: MIT Press.

Noam Chomsky. 1995. *The minimalist program*. Cambridge, Massachusetts: The MIT Press.

Grzegorz Chrupała and Afra Alishahi. 2019. Correlating neural and symbolic representations of language. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2952–2962, Florence, Italy. Association for Computational Linguistics.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy. Association for Computational Linguistics.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *ICML*.

Cleo Condoravdi, Dick Crouch, Valeria de Paiva, Reinhard Stolle, and Daniel G. Bobrow. 2003. Entailment, intensionality and text understanding. In *Proceedings of the HLT-NAACL 2003 Workshop on Text Meaning*, pages 38–45.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loic Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. *arXiv*.

Jillian Da Costa and Rui Chaves. 2020. Assessing the ability of transformer-based neural models to represent structurally unbounded dependencies. In *Proceedings of the Society for Computation in Linguistics 2020*, pages 12–21, New York, New York. Association for Computational Linguistics.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *Machine Learning Challenges Workshop*. Springer.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2006. The PASCAL recognising textual entailment challenge. In *Machine learning challenges. evaluating predictive uncertainty, visual object classification, and recognising tectual entailment*. Springer.

Ishita Dasgupta, Demi Guo, Andreas Stuhlmüller, Samuel J Gershman, and Noah D Goodman. 2018. Evaluating compositionality in sentence embeddings. In *Proceedings of Annual Meeting of the Cognitive Science Society*.

Forrest Davis and Marten van Schijndel. 2020. Recurrent neural network language models always learn English-like relative clause attachment. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1979–1990, Online. Association for Computational Linguistics.

Allyson Ettinger. 2020. What bert is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *TACL*.

Gottlob Frege. 1948. Sense and reference. *The philosophical review*.

Kanishk Gandhi and Brenden M Lake. 2019. Mutual exclusivity as a challenge for neural networks. *arXiv*.

Jon Gauthier, Jennifer Hu, Ethan Wilcox, Peng Qian, and Roger Levy. 2020. SyntaxGym: An online platform for targeted evaluation of language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 70–76, Online. Association for Computational Linguistics.

Yoav Goldberg. 2019. Assessing bert's syntactic abilities. *arXiv preprint arXiv:1901.05287*.

Kristina Gulordava, Piotr Bojanowski, Edouard Grave, Tal Linzen, and Marco Baroni. 2018. Colorless green recurrent networks dream hierarchically. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1195–1205, New Orleans, Louisiana. Association for Computational Linguistics.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. 2018. Annotation artifacts in natural language inference data. 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 107–112, New Orleans, Louisiana. Association for Computational Linguistics.

Zellig S Harris. 1954. Distributional structure. *Word*.

Robert Hawkins, Takateru Yamakoshi, Thomas Griffiths, and Adele Goldberg. 2020. Investigating representations of verb bias in neural language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4653–4663, Online. Association for Computational Linguistics.

Irene Heim and Angelika Kratzer. 1998. *Semantics in generative grammar*. Blackwell Oxford.

John Hewitt 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, Minneapolis, Minnesota. Association for Computational Linguistics.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Hai Hu, Kyle Richardson, Liang Xu, Lu Li, Sandra Kübler, and Lawrence Moss. 2020a. OCNLI: Original Chinese Natural Language Inference. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3512–3526, Online. Association for Computational Linguistics.

Jennifer Hu, Jon Gauthier, Peng Qian, Ethan Wilcox, and Roger Levy. 2020b. A systematic assessment of syntactic generalization in neural language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1725–1744, Online. Association for Computational Linguistics.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

Paloma Jeretic, Alex Warstadt, Suvrat Bhooshan, and Adina Williams. 2020. Are natural language inference models IMPPRESsive? Learning IMPlicature and PRESupposition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8690–8705, Online. Association for Computational Linguistics.

Ronald M Kaplan and Joan Bresnan. 1995. Formal system for grammatical representation. *Formal Issues in Lexical-Functional Grammar*.

Jordan Kodner and Nitish Gupta. 2020. Overestimation of syntactic representation in neural language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1757–1762, Online. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Peiguang Li, Hongfeng Yu, Wenkai Zhang, Guangluan Xu, and Xian Sun. 2020. SA-NLI: A supervised attention based framework for natural language inference. *Neurocomputing*.

Yongjie Lin, Yi Chern Tan, and Robert Frank. 2019. Open sesame: Getting inside BERT's linguistic knowledge. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 241–253, Florence, Italy. Association for Computational Linguistics.

Tal Linzen and Marco Baroni. 2021. Syntactic structure from deep learning. *Annual Review of Linguistics*.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. 2016. Assessing the ability of LSTMs to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics*, 4:521–535.

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*.

Christopher D Manning. 2015. Computational linguistics and deep learning. *Computational Linguistics*.

Christopher D Manning, Kevin Clark, John Hewitt, Urvashi Khandelwal, and Omer Levy. 2020. Emergent linguistic structure in artificial neural networks trained by self-supervision. *Proceedings of the National Academy of Sciences*.

Rebecca Marvin and Tal Linzen. 2018. Targeted syntactic evaluation of language models. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1202, Brussels, Belgium. Association for Computational Linguistics.

R. Thomas McCoy, Junghyun Min, and Tal Linzen. 2020. BERTs of a feather do not generalize together: Large variability in generalization across models with similar test set performance. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 217–227, Online. Association for Computational Linguistics.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv*.

Richard Montague. 1970. Universal grammar. *Theoria*.

Aakanksha Naik, Abhilasha Ravichander, Carolyn Rose, and Eduard Hovy. 2019. Exploring numeracy in word embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3374–3380, Florence, Italy. Association for Computational Linguistics.

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, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online. Association for Computational Linguistics.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. *arXiv*.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. Hypothesis only baselines in natural language inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 180–191, New Orleans, Louisiana. Association for Computational Linguistics.

Carl Pollard and Ivan A Sag. 1994. *Head-driven phrase structure grammar*. University of Chicago Press.

Grusha Prasad, Marten van Schijndel, and Tal Linzen. 2019. Using priming to uncover the organization of syntactic representations in neural language models. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 66–76, Hong Kong, China. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*.

Shauli Ravfogel, Yoav Goldberg, and Tal Linzen. 2019. Studying the inductive biases of RNNs with synthetic variations of natural languages. 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 3532–3542, Minneapolis, Minnesota. Association for Computational Linguistics.

Shauli Ravfogel, Yoav Goldberg, and Francis Tyers. 2018. Can LSTM learn to capture agreement? the case of Basque. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 98–107, Brussels, Belgium. Association for Computational Linguistics.

Abhilasha Ravichander, Aakanksha Naik, Carolyn Rose, and Eduard Hovy. 2019. EQUATE: A benchmark evaluation framework for quantitative reasoning in natural language inference. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 349–361, Hong Kong, China. Association for Computational Linguistics.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A Primer in BERTology: What we know about how BERT works. *arXiv*.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter.

Yves Schabes, Anne Abeille, and Aravind K. Joshi. 1988. Parsing strategies with 'lexicalized' grammars: Application to Tree Adjoining Grammars. In *Coling Budapest 1988 Volume 2: International Conference on Computational Linguistics*.

Eckart Scheerer. 1981. Early german approaches to experimental reading research: The contributions of wilhelm wundt and ernst meumann. *Psychological Research*.

Marten van Schijndel and Tal Linzen. 2018. A neural model of adaptation in reading. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4704–4710, Brussels, Belgium. Association for Computational Linguistics.

Marten van Schijndel, Aaron Mueller, and Tal Linzen. 2019. Quantity doesn't buy quality syntax with neural language 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 5831–5837, Hong Kong, China. Association for Computational Linguistics.

Joshua Snell and Jonathan Grainger. 2017. The sentence superiority effect revisited. *Cognition*.

Whitney Tabor. 1994. *Syntactic innovation: A connectionist model*. Ph.D. thesis.

Ian Tenney, 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, Florence, Italy. Association for Computational Linguistics.

Hiroshi Toyota. 2001. Changes in the constraints of semantic and syntactic congruity on memory across three age groups. *Perceptual and Motor Skills*.

Masatoshi Tsuchiya. 2018. Performance impact caused by hidden bias of training data for recognizing textual entailment. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Lucy Vanderwende and William B Dolan. 2005. What syntax can contribute in the entailment task. In *Machine Learning Challenges Workshop*. Springer.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*.

Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner, and Sameer Singh. 2019. Universal adversarial triggers for attacking and analyzing NLP. 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 2153–2162, Hong Kong, China. Association for Computational Linguistics.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. In *NeurIPS*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Alex Warstadt, Yu Cao, Ioana Grosu, Wei Peng, Hagen Blix, Yining Nie, Anna Alsop, Shikha Bordia, Haokun Liu, Alicia Parrish, Sheng-Fu Wang, Jason Phang, Anhad Mohananey, Phu Mon Htut, Paloma Jeretic, and Samuel R. Bowman. 2019. Investigating BERT's knowledge of language: Five analysis methods with NPIs. 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 2877–2887, Hong Kong, China. Association for Computational Linguistics.

Yun Wen, Joshua Snell, and Jonathan Grainger. 2019. Parallel, cascaded, interactive processing of words during sentence reading. *Cognition*.

Aaron Steven White, Rachel Rudinger, Kyle Rawlins, and Benjamin Van Durme. 2018. Lexicosyntactic inference in neural models. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4717–4724, Brussels, Belgium. Association for Computational Linguistics.

Ethan Wilcox, Roger Levy, Takashi Morita, and Richard Futrell. 2018. What do RNN language models learn about filler–gap dependencies? In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 211–221, Brussels, Belgium. Association for Computational Linguistics.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Thomas Wolf. 2019. Some additional experiments extending the tech report" assessing berts syntactic

abilities" by yoav goldberg. Technical report, HuggingFace.

Thomas Wolf, Julien Chaumond, Lysandre Debut, Victor Sanh, Clement Delangue, Anthony Moi, Pierric Cistac, Morgan Funtowicz, Joe Davison, Sam Shleifer, et al. 2020. Transformers: State-of-the-art natural language processing. In *EMNLP: System Demonstrations*.

Kelly Zhang and Samuel Bowman. 2018. Language modeling teaches you more than translation does: Lessons learned through auxiliary syntactic task analysis. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 359–361, Brussels, Belgium. Association for Computational Linguistics.

Han Zhao, Zhengdong Lu, and Pascal Poupart. 2015. Self-adaptive hierarchical sentence model. *arXiv*.

A Effect of Length on Permutation Acceptance

We investigate the effect of length on Permutation Acceptancein Figure 5. We observe that shorter sentences in general have higher probability of acceptability for examples which was originally predicted correctly - since shorter sentences have less number of unique permutations. However, for the examples which were originally incorrect, the trend is not present.

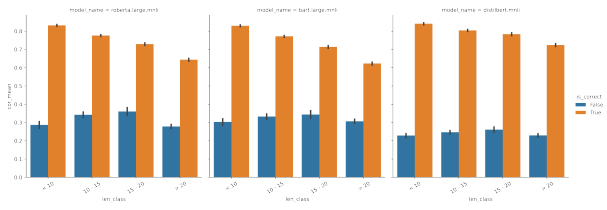


Figure 5: Length in Transformer-based models

B Example of POS Minitree

As we defined in §6, we develop a POS signature for each word in a sentence in a test set, comparing it with the distribution of the same word in the training set. Figure 6 provides a snapshot a word "river" from the test set and how the POS signature distribution of the word in a particular sentence match with that of aggregated training statistic. In practice, we the top k tags for the word in test signature as well as the train, and calculate the overlap of POS tags. When comparing the model performance with permuted sentences, we compute a ratio between this overlap score and the overlap score of the permuted sentence. In the Figure 6, 'river' would have a POS tag minitree score of 0.75.

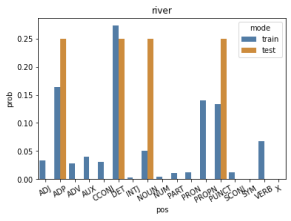


Figure 6: Example POS signature for the word 'river', calculated with a radius of 2. Probability of each neighbor POS tag is provided. Orange examples come from the permuted test set, and blue come from original train.

C Effect of Hypothesis only randomization

In recent years, the impact of the hypothesis sentence (Gururangan et al., 2018; Tsuchiya, 2018; Poliak et al., 2018) on NLI classification has been a topic of much interest. As we define in §3, logical entailment can only be defined for pairs of propositions. We investigated one effect where we randomize only the hypothesis sentences while keeping the premise intact. We find (Figure 8(a)) that the Ω_{max} value is almost similar for the two schemes, suggesting that even with randomizing the hypothesis the model can exhibit similar phenomenon.

D Effect of clumped words in random permutations

Since our original permuted dataset consists of extremely randomized words, we observe very low BLEU-3 (< 0.2) and BLEU-4 scores (< 0.1). To study the effect of overlap across a wider range of permutations, we devised an experiment where we clump certain words together before performing random permutations. Concretely, we clump 25%, 50% and 75% of the words in a sentence and then permute the remaining words and the clumped word as a whole. This type of clumped-permutation allows us to study the full range of BLEU-2/3/4 scores, which we present in Figure 9. As expected, the acceptability of permuted sentences increase linearly with BLEU score overlap.

E Effect of the threshold of Ω_x in various test splits

We defined two variations of Ω_x, Ω_{max} and Ω_{rand}, but theoretically it is possible to define any arbitrary threshold percentage x to evaluate the unnatural language inference mechanisms of different models. In Figure 7 we show the effect of different thresholds, includingΩ_{max} where $x = 1/|D_{\text{test}}|$ and Ω_{rand} where $x = 0.34$. We observe for in-distribution datasets (top row, MNLI and SNLI splits), in the extreme setting when $x = 1.0$, there are more than 10% of examples available, and more than 25% in case of InferSent and DistilBERT. For out-of-distribution datasets (bottom row, ANLI splits) we observe a much lower trend, suggesting generalization itself is the bottleneck in permuted sentence understanding.

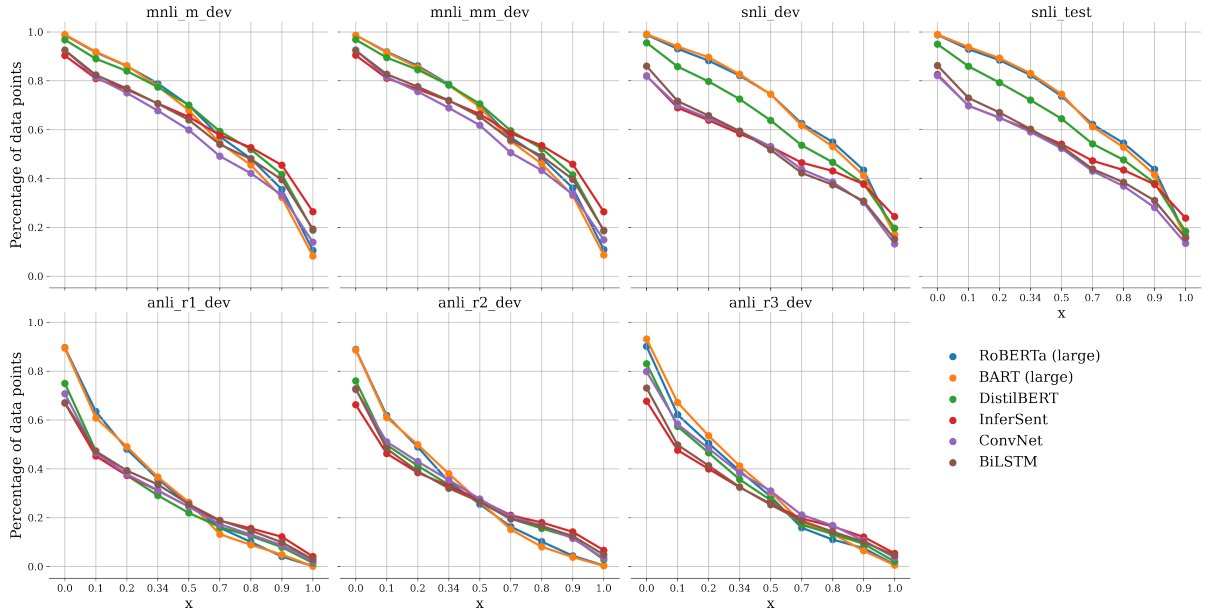
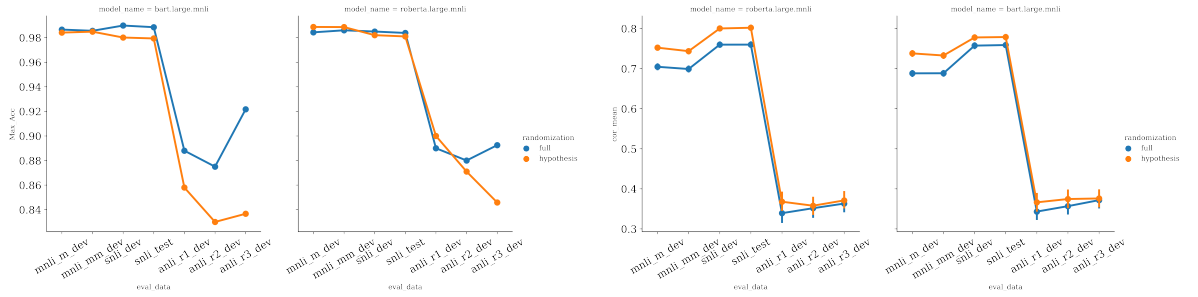


Figure 7: Ω_x threshold for all datasets with varying x and computing the percentage of examples that fall within the threshold. The top row consists of in-distribution datasets (MNLI, SNLI) and the bottom row contains out-of-distribution datasets (ANLI)



(a) (b)

Figure 8: Comparing the effect between randomizing both premise and hypothesis and only hypothesis on two Transformer-based models, RoBERTa and BART (For more comparisons please refer to Appendix). In 8(a), we observe the difference of Ω_{max} is marginal in in-distribution datasets (SNLI, MNLI), while hypothesis-only randomization is worse for out-of-distribution datasets (ANLI). In 8(b), we compare the mean number of permutations which elicited correct response, and naturally the hypothesis-only randomization causes more percentage of randomizations to be correct.

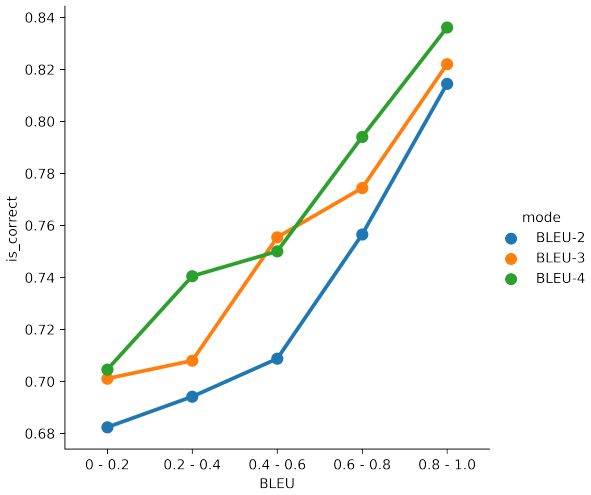


Figure 9: Relation of BLEU-2/3/4 scores against the acceptability of clumped-permuted sentences accross all test datasets on all models.