Systematic language understanding: a study on the capabilities and limits of language understanding by modern neural networks

Koustuv Sinha
Ph.D. Proposal Document

September 2021

1 Introduction

Language allows us to express and comprehend a vast variety of novel thoughts and ideas. Through language, humans exhibit higher-order reasoning and comprehension. Thus, to develop models which mimic human-like reasoning, a principled focus in computer science research is to develop models which understand and reason on natural language. To foster research in developing such state-of-the-art natural language understanding (NLU) models, several datasets and tasks on reading comprehension have been proposed in recent literature. These include tasks such as question answering (QA), natural language inference (NLI), commonsense reasoning to name a few. Over the last decade, several advancements have been made to develop such models, the most successful ones till date involve deep neural models, especially Transformers [94], a class of multi-head self-attention models. Since its introduction in 2017, Transformer-based models have achieved impressive results on numerous benchmarks and datasets, with BERT [20] being one of the most popular instantiation of the same. Using a technique known as "pre-training", Transformer-based models are first trained to replicate massive corpus of text. Through this kind of unsupervised training, the models learn and tune their millions and billions of parameters, and using which they solve NLU datasets with surprising, near-human efficiency [20, 52, 50].

While Transformer-based models excel in these datasets, it is less clear why do they work so well. Due to the sheer amount of overparameterization, direct inspection of the inner workings of these models are limited. Thus, various research have been conducted by using auxilliary tasks and probing functions to understand the reasoning processes employed by these models [75]. It has been claimed in the literature that BERT embeddings contain syntactic information about a given sentence, to the extent that the model may internally perform several natural language processing pipeline steps, involving parts-of-speech tagging, entity recognition etc . BERT has also been credited to acquire some level syntactic [37, 43], semantic [90, 22], and world knowledge [67, 75]. All of these results indicate to the fact that purely pre-training with massive overparameterized models and large corpora might just be the perfect roadmap to achieve "human-like" reasoning capabilities.

On the other hand, there have been growing concerns regarding the ability of these NLU models to understand language in a "systematic" and robust way. The phenomenon of *systematicity*, widely studied in the cognitive sciences, refers to the fact that lexical units such as words make consistent contributions to the meaning of the sentences in which they appear [24]. As an illustration, they provide an example that all English speakers who understand the sentence "John loves the girl" should also understand the phrase "the girl loves John". In case of NLU tasks, this accounts to model being consistent in understanding novel compositions of existing, learned words or phrases. However, there

is growing evidence in literature which highlight the brittleness of NLU systems to such adversarial examples [45, 47]. More so, there is strong evidence that state-of-the-art NLU models tend to exploit statistical artifacts in datasets, rather than exhibiting true reasoning and generalization capabilities [33, 70, 92, 61, 56].

In view of the positive and negative evidences towards Transformers acquiring "human-like" natural language understanding capacity, it is very important that we take a step back and carefully examine the reasoning processes of the NLU models in the view of systematicity and robustness. Since these NLU models are now being deployed in production and decision making systems, it is even more prudent to test the models towards systematic understanding in order to avoid catastrophic scenarios. In this proposal, I thus discuss my work till now in my doctoral studies to understand the limits of systematic and robust natural language understanding of NLU models. Concretely, first I discuss our proposed systematicity tests on artificial and natural languages by using first-order logic (FOL), and what we learned from the model using such tests. This involves the following paper:

- *CLUTRR: A Diagnostic Benchmark for Inductive Reasoning from Text*, published at Empirical Methods of Natural Language Processing (EMNLP) 2019 (Oral presentation) [82]

This document briefly discusses about my other related works in this topic, such as in Natural Language Inference, *Probing Linguistic Systematicity* [1], published at Association for Computational Linguistics (ACL) 2020 [29]; or in proof generation, *Measuring Systematic Generalization in Neural Proof Generation* [28], published at Neural Information Processing Systems (NeurIPS) 2020, but not in detail.

Secondly, I discuss our work on understanding the limits of systematicity of NLU models by subjecting these models to scrambled word order sentences, involving the following two papers:

- *UnNatural Language Inference*, published at Association for Computational Linguistics (ACL) 2021 (Oral presentation, Outstanding paper award) [80]
- *Masked Language Modeling and the Distributional Hypothesis: Order Word Matters Pre-training for Little*, published at Empirical Methods of Natural Language Processing (EMNLP) 2021 [79]

This document does not discuss concurrent works on analyzing faithfulness and robustness in translations [64] (published at EMNLP 2021), proposed dataset on systematic reasoning on graph neural networks [83], or an unreferenced automatic dialog evaluation framework [81] (published at ACL 2020) conducted during my doctoral studies.

2 Background: Natural language understanding through Neural approaches

Natural language is highly ambiguous (syntactic ambiguity, word sense ambiguity, semantic ambiguity) as well as contextual (multiple interpretations based on usage). To understand natural language, neural network based approaches operate on the notion of *distributional semantics* [5], which captures the meaning of words through vector representation to compose a meaning representation of the sentence or relations. This makes neural network approaches more robust to noise and ambiguity of natural language [3].

Models. Natural Language Understanding (NLU) models have thus built on top of neural models having distributional vector representation of the lexical items. Further improvements were made to handle the polysemous and context-dependent nature of words by using contextual embeddings through leveraging the sequences of sentences, typically using Recurrent (LSTM, GRU) or Convolutional

[1] Work done as second author.

models (CNN). Models based on Long Short Term Memory (LSTM) became the de-facto standard in NLU, as they can effectively capture the contextual information with locality bias present in natural language. However, its performance is often affected by the long-term dependency problem in several NLU tasks.

Recently, the field of NLP has witnessed a paradigm shift in research with the advent of Transformers [94]. Being a fully-connected multi-head self-attention model, Transformers can directly model the dependency between any two words in a sequence, thus proving to be more powerful and suitable to model long range dependencies of natural language. However, Transformers come with massively increased model parameters, thus requiring significantly large corpus to train and resulting in overfitting on small and medium sized datasets [72].

On the rise of pre-training. With increasing number of model parameters, the requirement to train on large datasets became inevitable - as otherwise it is not possible to fully train the model parameters. However, it is challenging to collect/design NLU tasks of such scale due to extremely expensive annotation costs. In contrast, it is relatively easy to obtain large-scale unlabeled corpora, typically through scraping the internet or collecting all literary works. Thus came the idea of *pre-training*, where this huge, unlabeled data is first leveraged to learn a good representation, and then the resulting model used for NLU tasks. The large, unlabeled data can be leveraged to learn useful word embeddings either using pairwise ranking [12], or through the use of shallow architectures [57], or by computing global word-word co-occurence statistics [65]. However, these pre-trained word embeddings are still context-independent, and requires learning context-dependent parameters from scratch on an NLU task. Thus, efforts have been made to pre-train the entire model parameters instead of generating word embeddings by training in a Language Modeling objective (predicting the probability of a word to appear after a sequence of words), using bidirectional LSTM based architectures [66]. These models are then used to fine-tune on downstream NLU tasks such as text classification [**?**].

More recently, Transformer-based architectures became widely popular with the introduction of BERT [20]. BERT is fundamentally a stack of Transformer layers, having multiple self-attention heads. To learn its massive amount of parameters, BERT is pre-trained on unlabeled corpora using two self-supervised objectives: masked language modeling (MLM, predicting the probability of randomly masked input tokens) and next sentence prediction (predicting if two sentences are adjacent to each other). BERT significantly improved the state-of-the art in many NLU tasks - ranging from text classication, question answering, natural language inference, machine translation, summarization etc [71].

Analysis and interpretability of NLU models. The sheer performance of pre-trained Transformer-based models following BERT created a watershed moment in NLU, as these models outperform previous LSTM-based models by a large extent. Due to this success, interest peaked in the community to investigate the inner workings of this largely black box model [75]. The investigation is primarily done using the tool of *probing*, which is learning a function on top of pre-trained representations to perform targeted assesment of linguistic information [42]. How well this probe learns a given signal can be seen as a proxy for linguistic knowledge encoded in the representations. Using these probes, it has been shown BERT representations contain adequate syntactic [37, 43], semantic [90, 22], and world knowledge [67, 75].

Brittleness issues in NLU models. Despite the large performance gain and representations containing the necessary linguistic information to process natural language, state-of-the-art NLU models are often subject to scrutiny due to their unsystematic behaviors on specifically crafted test suites. NLU models are repeatedly shown to be brittle when subject to adversarial attacks [45, 47] to the input sentence forms [48]. NLU models also tend to exploit the statistical irregularities and annotation artefacts [33, 70, 92] of a given datasets, resulting in failure cases on carefully crafted examples [61, 56]. NLU models are also subjected to tests of systematic generalization in context of semantic parsing in novel compositions [49], shuffled argument structure in natural language inference [18], etc. On these tests, typically the NLU models fail to behave in the expected ways, unless trained with the same objective

as the test sets.

My work during my doctoral thesis has been to uncover why these models reason un-systematically using targeted semantic and syntactic tests.

3 Contribution 1: Investigating systematicity of NLU models using first order logic

3.1 Motivation

An important challenge in NLU is to develop benchmarks which can precisely test a model's capability for robust and systematic generalization. Ideally, we want language understanding systems that can not only answer questions and draw inferences from text, but that can also do so in a systematic, logical and robust way. While such reasoning capabilities are certainly required for many existing NLU tasks, most datasets combine several challenges of language understanding into one, such as co-reference/entity resolution, incorporating world knowledge, and semantic parsing - making it difficult to isolate and diagnose a model's capabilities for systematic generalization and robustness.

Thus, inspired by the classic AI challenge of inductive logic programming, we propose a semi-synthetic benchmark designed to explicitly test an NLU model's ability for systematic and robust logical generalization. Our benchmark suite - termed CLUTRR (**C**ompositional **L**anguage **U**nderstanding with **T**ext-based **R**elational **R**easoning) - contains a large set of semi-synthetic stories involving hypothetical families. Given a story, the objective is to infer the relationship between two given characters in the story, whose relationship is not explicitly mentioned. To solve this task, a learning agent must extract the relationships mentioned in the text, induce the logical rules governing the kinship relationships (e.g, the transitivity of the sibling relation), and use rules to infer the relationship between a given pair of entities.

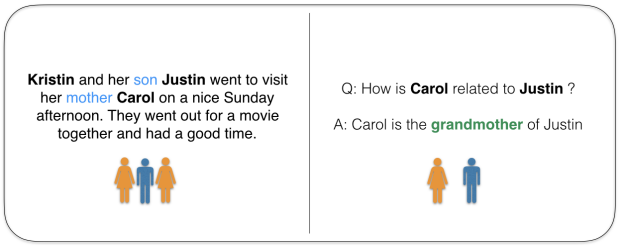

Figure 1: CLUTRR inductive reasoning task

3.2 Dataset

In order to design the CLUTRR benchmark, we build upon classic ILP task of inferring kinship relations [38, 59]. For example, given the facts that *"Alice is Bob's mother"* and *"Jim is Alice's father"*, one can infer with reasonable certainty that *"Jim is Bob's grandfather"*. While this example may appear trivial, it is challenging task to design models that can learn from data to *induce* the logical rules necessary to make such inferences, and it is even more challenging to design models that can systematically generalize by composing these induced rules. Thus, the core idea behind CLUTRR benchmark suite is the following: given a natural language story describing a set of kinship relations, the goal is to infer the relationship between

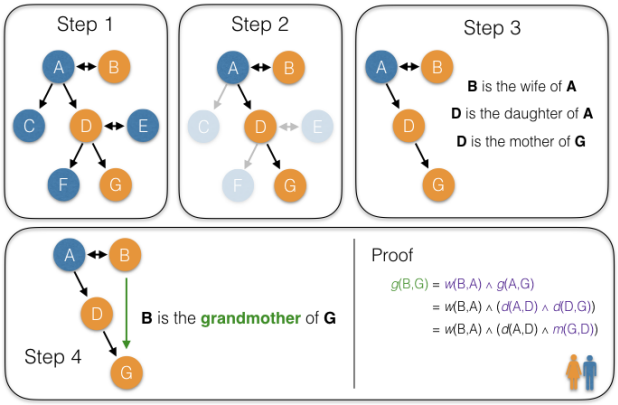

Figure 2: CLUTRR dataset design steps

two entities, whose relationship is *not* explicitly stated in the story.

Essentially, we use first order logic (FOL) to generate k number of provable facts and then apply natural language layer on top of it to create a semi-synthetic benchmark. The number k denotes the difficulty of the example. We use Amazon Mechanical Turk (AMT) crowd workers to annotate logical facts into narratives.

3.3 Experiments

In this section, we use CLUTRR to construct specific instances of the dataset to test various aspects of systematicity in natural language understanding. We report training and testing results on stories with different clause lengths k. (For brevity, we use the phrase "clause length" throughout this section to refer to the number of steps of reasoning that are required to predict the target query.) We also ensure the AMT templates are also split into train and test, to reduce the probability of overfitting to certain artifacts of the templates.

Human Performance. To get a sense of the data quality and difficulty involved in CLUTRR, we asked human annotators to solve the task for random examples of length $k = 2, 3, ..., 6$. We found that time-constrained AMT annotators performed well (i.e., $> 70\%$) accuracy for $k \leq 3$ but struggled with examples involving longer stories, achieving 40-50% accuracy for $k > 3$. However, trained annotators with unlimited time were able to solve 100% of the examples (Appendix 1.7), highlighting the fact that this task requires attention and involved reasoning, even for humans.

Are NLU models able to generalize systematically?

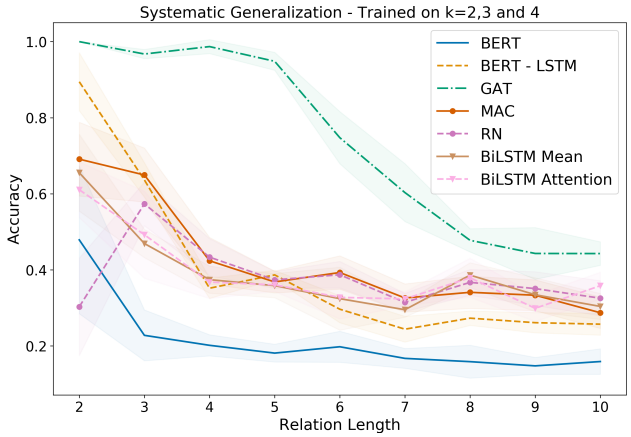

Figure 3: Systematic generalization results on CLUTRR, when trained on stories of length $k = 2, 3, 4$

In this setup, we consider the setting where the models are trained on stories generated from clauses of length $\leq k$ and evaluated on stories generated from larger clauses of length $> k$. Thus, we explicitly test the ability for models to generalize on examples that require more steps of reasoning that any example they encountered during training. In other words, during training, the model sees all logical rules but does not see all *combinations* of these logical rules.

Figure 3 illustrates the performance of different NLU models on this generalization task. For NLU models, we consider bidirectional LSTMs [39, 9] (with and without attention), as well as recently proposed models that aim to incorporate inductive biases towards relational reasoning: Relation Networks (RN) [77] and Compositional Memory Attention Network (MAC) [41]. We also use the large pretrained language model, BERT [20], as well as a modified version of BERT having a trainable LSTM encoder on top of the pretrained BERT embeddings. Since the underlying relations in the stories generated by CLUTRR inherently form a graph, we also experiment with a Graph Attention Network (GAT) [95]. Rather than taking the textual stories as input, the GAT baseline receives a structured graph representation of the facts that underlie the story.

We observe that the GAT model is able to perform near-perfectly on the held-out logical clauses of length $k = 3$, with the BERT-LSTM being the top-performer among the text-based models but still significantly below the GAT. Not surprisingly, the performance of all models degrades monotonically as we increase the length of the test clauses, which highlights the challenge of "zero-shot" systematic

Table 1: Testing the robustness of the various models when training and testing on stories containing various types of noise facts.

Models		Unstructured models (no graph)						Structured model (with graph)
Training	Testing	BiLSTM - Attention	BiLSTM - Mean	RN	MAC	BERT	BERT-LSTM	GAT
Clean	Clean	0.58 ±0.05	0.53 ±0.05	0.49 ±0.06	0.63 ±0.08	0.37 ±0.06	0.67 ±0.03	**1.0** ±0.0
	Supporting	**0.76** ±0.02	0.64 ±0.22	0.58 ±0.06	0.71 ±0.07	0.28 ±0.1	0.66 ±0.06	0.24 ±0.2
	Irrelevant	0.7 ±0.15	**0.76** ±0.02	0.59 ±0.06	0.69 ±0.05	0.24 ±0.08	0.55 ±0.03	0.51 ±0.15
	Disconnected	0.49 ±0.05	0.45 ±0.05	0.5 ±0.06	0.59 ±0.05	0.24 ±0.08	0.5 ±0.06	**0.8** ±0.17
Supporting	Supporting	0.67 ±0.06	0.66 ±0.07	0.68 ±0.05	0.65 ±0.04	0.32 ±0.09	0.57 ±0.04	**0.98** ±0.01
Irrelevant	Irrelevant	0.51 ±0.06	0.52 ±0.06	0.5 ±0.04	0.56 ±0.04	0.25 ±0.06	0.53 ±0.06	**0.93** ±0.01
Disconnected	Disconnected	0.57 ±0.07	0.57 ±0.06	0.45 ±0.11	0.4 ±0.1	0.17 ±0.05	0.47 ±0.06	**0.96** ±0.01
Average		**0.61** ±0.08	0.59 ±0.08	0.54 ±0.07	**0.61** ±0.06	0.30 ±0.07	0.56 ±0.05	**0.77** ±0.09

generalization [49, 87]. GAT, having access to structured input, is able to generalize significantly better compared to NLU models.

How does NLU systems cope with noise - how robustly do they reason?

Finally, we use CLUTRR to systematically evaluate how NLU models cope with noise. Any set of supporting facts generated by CLUTRR can be interpreted as a path in the corresponding kinship graph G (Figure 4). Based on this interpretation, we view adding noise facts to the *clean path* from the perspective of sampling three different types of noise paths, from the kinship graph G. First, we test for *irrelevant facts*, which consists of facts which has one entity common with the path, but they themselves are irrelevant to solve the task. Second, we test for *supporting facts*, which contain a set of facts which can be used to construct an alternate path between the source and sink (i.e, target entities). Finally, we test for *disconnected facts*, which contain facts which are irrelevant as well as do not contain any entities which are in the path.

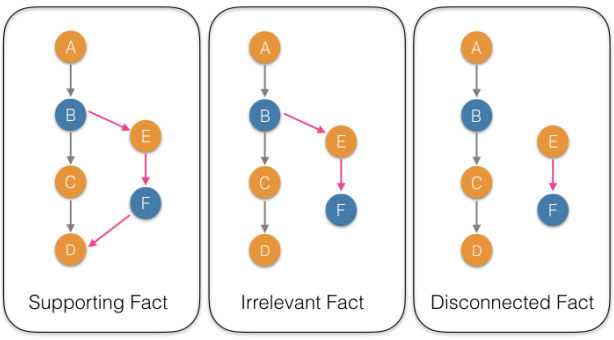

Figure 4: Noise generation methods in CLUTRR

Overall, we find that the GAT baseline outperforms the unstructured text-based models across most testing scenarios (Table 1), which showcases the benefit of a structured feature space for robust reasoning. When training on clean data and testing on noisy data, we observe two interesting trends that highlight the benefits and shortcomings of the various model classes. Firstly, all the text-based models excluding BERT actually perform better when testing on examples that have *supporting* or *irrelevant* facts added. This suggests that these models actually benefit from having more content related to the entities in the story. Even though this content is not strictly useful or needed for the reasoning task, it may provide some linguistic cues (e.g., about entity genders) that the models exploit. In contrast, the BERT-based models do not benefit from the inclusion of this extra content, which is perhaps due to the fact that they are already built upon a strong language model (e.g., that already adequately captures entity genders.) Secondly, the GAT model performs poorly when *supporting* facts are added but has no performance drop when *disconnected* facts are added. This suggests that the GAT model is sensitive to changes that introduce cycles in the underlying graph structure but is robust to the addition of noise that is disconnected from the target entities. Moreover, when we trained on noisy examples, we found that only the GAT model was able to consistently improve its performance (Table 1). Again, this highlights the performance gap between the unstructured text-based models and the GAT.

3.4 Discussion

In this paper we introduced the CLUTRR benchmark suite to test the systematic generalization and inductive reasoning capababilities of NLU systems. We demonstrated the diagnostic capabilities of CLUTRR and found that existing NLU systems exhibit relatively poor robustness and systematic generalization capabilities—especially when compared to a graph neural network that works directly with symbolic input. These results highlight the gap that remains between machine reasoning models that work with unstructured text and models that are given access to more structured input. We hope that by using this benchmark suite, progress can be made in building more compositional, modular, and robust NLU systems.

3.5 Related Works

We also conduct a couple of related studies in testing systematicity of Natural Language Understanding (NLU) and Natural Language Generation (NLG) models following the intuition gained from CLUTRR. ***Probing Linguistic Systematicity***. [29] In this work, we introduce several novel probes for testing systematic generalization in Natural Language Inference (NLI). Systematicity is the property whereby words have consistent contributions to composed meaning of the sentences. In this work, we employed an artificial, controlled language where we use *Jabberwocky*-type [2] sentences to inspect the generalizability of word representations learned by neural networks. We gradually and systematically expose the NLU model to new, *open-class* words in context of NLI tasks, and test whether this exposure alters the systematic understanding of existing, known *closed-class* words. For example, we might train an NLI models with the premise-hypothesis contradiction pair *All pigs sleep; some pigs don't sleep*, and test whether the network can identify the contradiction pair *All Jabberwocks flug; some Jabberwocks don't flug*. A systematic learner would reliably identify the contradiction, whereas a non-systematic learner may allow the closed-class words (*all, some, don't*) to take contextually conditioned meanings that depend on novel context words.

Position	1	2	3	4	5	6
Category	`quantifier`	`nominal premodifier`	`noun`	`nominal postmodifier`	`negation`	`verb`
Status	Obligatory	Optional	Obligatory	Optional	Optional	Obligatory
Class	Closed	Closed	Open	Closed	Closed	Open
Example	All	brown	dogs	that bark	don't	run

Table 2: A template for sentences in the artificial language. Each sentence fills the obligatory positions 1, 3, and 6 with a word: a quantifier, noun, and verb. Optional positions (2, 4 and 5) are filled by either a word (adjective, postmodifier or negation) or by the empty string. Closed-class categories (Quantifiers, adjectives, post modifiers and negation) do not include novel words, while open-class categories (nouns and verbs) includes novel words that are only exposed in the test set.

Concretely, we construct an artificial language with six-position template which includes a quantifier (position 1), noun (position 3), and a verb (position 6) with options pre- and post-modifiers (position 2 and 4) and optional negation (position 5). To mimic real world topicality, we contruct *block* structures consisting of nouns and verbs having taxonomic relationships (such as *lizards/animals. run/move*). Nouns and verbs from different blocks have no relationships (such as *lizards* and *screwdrivers* or *run* and *read*). The same set of closed-class words appear in all blocks with consistent meanings. We analyze several state-of-the-art NLI models such as Bidirectional LSTM, InferSent, self-attentive sentence encoder (SATT) and Hierarchical Convolutional Networks (CONV) [3].

[2] Jabberwocky is the term coined by Lewis Caroll in his poem, which combibes nonsense words with familiar words in a way that allows speakers to recognize the expression as well formed.

[3] Since the first version of the paper was done prior to the popularity of BERT, we were unable to test the systematicty of BERT-based models in this work. However, our database and code are online, and it would be trivial to use pre-trained

We observed all models to perform substantially worse on probing tasks, with standard deviation being significantly high among various blocks - indicating unsystematic behavior. Closed-class words do not maintain a consistent intepretation when paired with different open-class words. Variance across blocks shows that under all models the behaviour of closed-class words is highly sensitive to the novel words they appear with. Thus, our experiments highlight that fact their despite high overall performance, state-of-the-art NLU models generalize in ways that allow the meanings of individual words to vary in different contexts, even in an artificial language where a totally systematic solution is available.

Measuring Systematic Generalization in Neural Proof Generation with Transformers. [28] In this work, we extend our systematicity analysis to language generation using Transformer Language Models (TLMs). To analyze systematicity, we re-use our CLUTRR benchmark to conduct proof generation using forward and backward chaining concepts in first-order logic (FOL). For example, a set of facts in CLUTRR could be of the form: *"Nat is the granddaughter of Betty", "Greg is the brother of Nat", "Flo is the sister of Greg"*, where the relationship among *Flo* and *Betty* can be inferred using logical deduction (*"Flo is the granddaughter of Betty"*). In this example, we further task the TLM to generate a plausible proof along with the answer : *"Since Flo is the sister of Greg, and Nat is the granddaughter of Betty, and Greg is the brother of Nat, then Flo is the granddaugher to Betty"*.

In our work, we evaluate two popular proof resolution strategies used in Inductive Logic Programming [23] , *forward* and *backward* chaining resolution paths, expressed in natural language. We evaluate the validity of the proof and the answer accuracy on various settings: whether the TLM is tasked to generated forward or backward proofs, whether the TLM is provided with a gold proof, or when the TLM is neither provided nor tasked to generate a proof. We train a Transformer [93] model on scratch on the training set, and we observe that TLMs are only able to generalize to unseen proof steps in case of *interpolation*, that is when stories of lesser difficulties than training are provided during inference. In case of *extrapolation*, we observe similar generalization issues as in CLUTRR, where models fail to generalize beyond the difficulty trained. In terms of proof understanding, we observed backward chaining proofs are better understood by the model than forward chaining for the TLMs, mostly due to the fact that backward chaining proofs always begings with the target answer first, allowing the model to exploit the positional cues. Surprisingly, we found the no proof situation to have better answer accuracy than in the case of proof generation - alluding to the fact that proof generation might be actually deterioriating the model performance as it requires more involved reasoning.

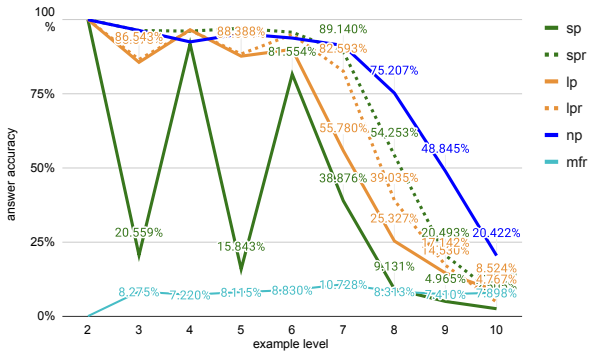

Figure 5: Systematic generalization issues in proof generation

Finally, in proof generation, we found forward-chaining generation is easier for TLM than backward chaining generation. This is contrary to our previous observation, and we believe this is due to the fact that the model has a higher chance of generating the first proof step correctly than the final proof step. Overall, TLMs are unable to generate valid proofs of unseen lengths, both in *interpolation* and *extrapolation* setting. However, when provided with the correct proof, TLMs are better able to exploit the information in it to be better at systematic generalization. Our results highlight multiple insights - first, TLMs suffer length generalization issues in proof generation, and TLMs get better at reasoning when provided with correct proofs. Our framework can thus be used to easily analyze systematicity issues in generation as it is grounded with first-order logic.

BERT models to run the same experiments.

4 Contribution 2: Probing systematicity of pre-trained models using word order

As we uncover the systematicity issues in state-of-the-art NLU models, one question emerges - does the unsystematic behaviour of NLU models stem from their inability to understand syntax? Since we use artificial and semi-synthetic datasets in all our prior works, having a rudimentary variation of syntax, our findings naturally raise questions on the actual syntax understanding capabilities of NLU models. Of late, we have witnessed a steady increase in performance of large scale pre-trained Transformer-based [94] models—such as RoBERTa [52], BART [50], and GPT-2 and -3 [73, 7]–on many NLU tasks [97, 96]. The success of these models has prompted serious investigation, leading claims that a language modeling (LM) objective can capture syntactic information [37, 43, 99, 104], with their self-attention layers being capable of surprisingly effective learning [75].

In the following section, we discuss two related works [80, 79] on testing systematicity of syntax understanding of NLU models using the proxy of *word order*.

4.1 UnNatural Language Inference

4.1.1 Motivation

In this contribution, [80] we revisit the notion of "knowing syntax" under the lens of systematicity. A natural and common perspective from many formal theories of linguistics (e.g., [11]) 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 the *order of the words* in that sentence (among other things). Humans are sensitive to word order, so clearly, "language is not merely a bag of words" [35, p.156]. Moreover, it is easier for us to identify or recall words presented in canonical orders than in disordered, ungrammatical sentences; this phenomenon is called the *"sentence superiority effect"* ([8, 78, 91, 1, 84, 85, 101], i.a.). In our estimation then, if one wants to claim that a model "knows syntax", then they should minimally show that the model is sensitive to word order (at least for e.g. English or Mandarin Chinese).

Generally, knowing the syntax of a sentence is taken to be a prerequisite for understanding what that sentence means [36]. Models should have to know the syntax first then, if performing any particular NLU task that genuinely requires a humanlike understanding of meaning (cf. [2]). Thus, if our models are as good at NLU as our current evaluation methods suggest, we should expect them to be sensitive to word order (see tab:example). We find, based on a suite of permutation metrics, that they are not.

4.1.2 Experiments and Results

We focus here on textual entailment, one of the hallmark tasks used to measure how well models understand language [13, 16]. This task, often also called Natural Language Inference (NLI; [6], i.a.), typically consists of two sentences: a premise and a hypothesis. The objective is to predict whether the premise entails the hypothesis, contradicts it, or is neutral with respect to it.

We find rampant word order insensitivity in purportedly high performing NLI models. For nearly all premise-hypothesis pairs, **there are many permuted examples that fool the models** into providing the correct prediction. In case of MNLI, for example, the current state-of-the-art of 90.5% can be increased to **98.7**% merely by

Premise	**Hypothesis**	**Predicted Label**
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 .	E
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 .	C

Table 3: Examples from the MNLI Matched development set. Both the original example and the permuted one elicit the same classification label (entailment and contradiction respectively) from

permuting the word order of test set examples. We even find drastically increased cross-dataset generalization when we reorder words. This is not just a matter of chance—we show that the model output probabilities are significantly different from uniform. We verify our findings with three popular English NLI datasets—SNLI [6], MultiNLI [102] and ANLI [62])— and one Chinese one, OCNLI [40]. It is thus less likely that our findings result from some quirk of English or a particular tokenization strategy. We also observe the effect for various transformer architectures pre-trained on language modeling (BERT, RoBERTa, DistilBERT), and non-transformers, including a ConvNet, an InferSent model, and a BiLSTM.

How many permuted examples does the model accept?

We find for models trained and evaluated on MNLI (in-domain generalization), there exists at least one permutation for each example using which we can reach a *maximum accuracy* of **98.7%** on MNLI dev. and test sets (in RoBERTa, compared to *original accuracy* of 90.6% in the natural data. If we choose examples such that at least 33% of its permutations are assigned the gold label (random baseline), we can still cover 79.4% of the test set (*random accuracy*, see our paper [79] for more details on the construction of the metrics.) We observe the effect even in out-of-domain generalization with ANLI dataset, where the data splits resulted in a maximum accuracy value that is notably higher than the original accuracy (89.7% vs 45.6% for RoBERTa). As a consequence, we encounter many *flips*, i.e., examples where the model is unable to predict the gold label, but at least one permutation of that example is able to. However, recall this analysis expects us to know the gold label upfront, so this test can be thought of as running a word-order probe test on the model until the model predicts the gold label (or give up by exhausting our set of 100 permutations). For out-of-domain generalization, random accuracy decreases considerably (36.4% on A1). Overall, we find the *probability of acceptance* of permutations to be significantly high, suggesting a bag-of-words (BoW) likeness of the models. We find this BOW-likeness to be higher for certain non-Transformer models, (InferSent) (84.2% for InferSent compared to 70.7% for RoBERTa on MNLI). We extended the experiments to the Original Chinese NLI dataset [40, OCNLI], and re-used the pre-trained RoBERTa-Large and InferSent (non-Transformer) models on OCNLI. Our findings are similar to the English results, thereby suggesting that the phenomenon is not just an artifact of English text or tokenization.

How confident are models in accepting permuted examples?

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 fig:all˙entropy. 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 Transformer-based model. This skewness is not a property of model capacity, as we observe DistilBERT log probabilities to have similar skewness as RoBERTa (large) model, while exhibiting lower model accuracy, maximum accuracy, and random accuracy.

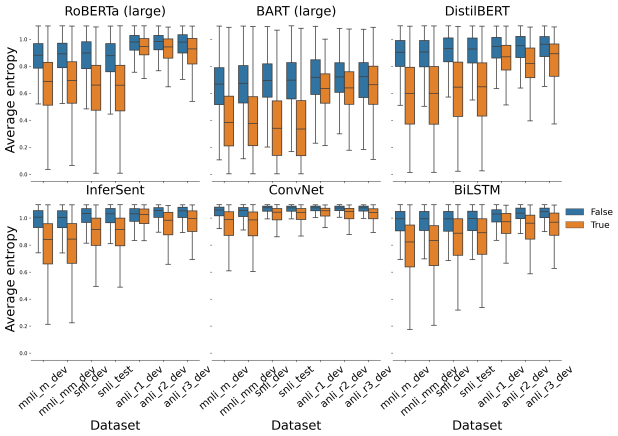

Figure 6: Model confidences on unnatural input

How does humans perform on permuted examples?

We expect humans to struggle with unnatural data, given our intuitions and the sentence superiority findings [58]. To test this, we presented two experts in NLI (one a linguist) with permuted sentence pairs to label.[4] Concretely, we draw equal number of examples from MNLI Matched dev set (100 examples where RoBERTa predicts the gold label, D^c and 100 examples where it fails to do so, D^f), and then permute these examples using $\mathscr{F}$. The experts were given no additional information (recall that it is common knowledge that NLI is a roughly balanced 3-way classification task). Unbeknownst to the experts, all permuted sentences in the sample were actually accepted by the RoBERTa (large) model (trained on MNLI dataset). We observe that the experts performed much worse than RoBERTa (58.1% and 37.8%), although their accuracy was a bit higher than random. We also find that for both experts, accuracy on permutations from D^c was higher than on D^f, which verifies findings that showed high word overlap can give hints about the ground truth label [18, 70, 32, 60].

4.1.3 Discussion

We show that state-of-the-art models do not rely on sentence structure the way we think they should: NLI models (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. We also show that reordering words can cause models to flip classification labels. We do find that models seem to have learned some syntactic information (we observed a correlation between preservation of abstract POS neighborhood information and rate of acceptance by models, please refer to the paper for more details) but these results do not discount the high rates of permutation acceptance, and require further verification. Coupled with the finding that humans cannot perform UNLI at all well, the high rate of permutation acceptance that we observe leads us to conclude that current models do not yet "know syntax" in the fully systematic and humanlike way we would like them to.

4.2 Masked Language Modeling and the Distributional Hypothesis: *Order word matters pre-training for little*

4.2.1 Motivation

In our previous work, we uncovered the unsystematic language understanding mechanisms employed by large, pre-trained NLU models, as they accept permuted sentences in surprising quantity. This questions the claims recently proposed in the literature that these models "rediscovers the classical NLP pipeline", suggesting that BERT-based models have the necessary inductive bias to learn syntax representations from self-supervised pre-training. Thus, we investigate further on the fact that how much of language models (specifically, Masked Language Models (MLM)) success is attributed to its knowledge of semantic and syntactic abstractions of natural language [79]. To do so, we measure the effect of removing word order during pre-training, with the assumption that any sophisticated (English) NLP pipeline presumably dpeends on the important syntactic information conveyed by the order of words. Suprisingly, we find the most of MLM's high performance can in fact be explained by the "distributional prior" - its ability to model word co-occurrence statistics - rather than its ability to replicate the classical NLP pipeline.

4.2.2 Experiments & Results

In our experiments, we pre-train MLMs (RoBERTa, [53]) on various corpora with permuted word order while preserving some degree of distributional information, and examine their downstream

[4]Concurrent work [31] found that untrained crowdworkers accept NLI examples that have been subjected to different kinds of perturbations at roughly most frequent class levels—i.e., only 35% of the time.

performance. In our main experiments, we pre-train models on various permuted corpora , by randomly shuffling n-grams within the sentence (where $n \in \{1,2,3,4\}$). We also experiment with training MLMs without positional embeddings, making them entirely order agnostic, and with training on a corpus sampled from the source corpus's unigram distribution, removing both distributional and word order information . We then evaluate these "permuted" models in a wide range of settings and compare with regularly-pre-trained models.

Concretely, we use the original 16GB BookWiki corpus (the Toronto Books Corpus, [106], plus English Wikipedia) from [53]. We denote the model trained on the original, un-modified BookWiki corpus as $\mathscr{M}_{\mathtt{N}}$ (for "natural"). We use two types of word order randomization methods: permuting words at the sentence level, and resampling words at the corpus level. We also account for n-grams such as bigram, trigram and four-gram - i.e. we construct permuted sentences by randomly keeping n-grams unchanged. We train RoBERTa models on four permutation variants of BookWiki corpus, $\mathscr{M}_{1}$, $\mathscr{M}_{2}$, $\mathscr{M}_{3}$, $\mathscr{M}_{4}$ for each n-gram value $\in \{1,2,3,4\}$. More details on the process, along with the pseudo code and sample quality, are provided in our paper [79]. To further measure the effect of the distributional information, we construct baselines devoid of any distributional signal. Specifically, we train RoBERTa on a variants of BookWiki Corpus, $\mathscr{M}_{\mathtt{UG}}$ where all unigrams are sampled from the corpus according to their frequencies, removing the co-occurrence information (i.e destroying all sentence and paragraph information). We also inspect the usefulness of word order by training a RoBERTa model without positional embedding ($\mathscr{M}_{\mathtt{NP}}$) and inspect the strength of inductive bias only by using a randomly initialized RoBERTa model ($\mathscr{M}_{\mathtt{RI}}$).

How does word order shuffled pre-trained models behave in downstream tasks?

In order to measure the effect of word order shuffled pre-training, we compare the models in the GLUE and PAWS downstream tasks. The GLUE [98] benchmark is a collection of 9 datasets for evaluating natural language understanding systems, of which we use Corpus of Linguistic Acceptability [CoLA, 100], Stanford Sentiment Treebank [SST, 86], Microsoft Research Paragraph Corpus [MRPC, 21], Quora Question Pairs (QQP)[5], Multi-Genre NLI [MNLI, 102], Question NLI [QNLI, 74, 19], Recognizing Textual Entailment [RTE, 17, 34, 26, 4]. The PAWS task [105] consists of predicting whether a given pair of sentences are paraphrases. This dataset contains both paraphrase and non-paraphrase pairs with high lexical overlap, which are generated by controlled word swapping and back translation. Since even a small word swap and perturbation can drastically modify the meaning of the sentence, we hypothesize the randomized pre-trained models will struggle to attain a high performance on PAWS. We fine-tune all models according to RoBERTa best practices.

We observe that fine-tuning on word order shuffled pre-trained models perform remarkably close to the natural word order pre-trained model in case of all tasks. $\mathscr{M}_{1}$, the model pre-trained on completely shuffled sentences, is on average only 3.3 points lower than $\mathscr{M}_{\mathtt{N}}$ on the accuracy-based tasks, and within 0.3 points of $\mathscr{M}_{\mathtt{N}}$ on QQP. Even on PAWS, which was designed to require knowledge of word order, $\mathscr{M}_{1}$ is within 5 points of $\mathscr{M}_{\mathtt{N}}$. Randomizing n-grams instead of words during pre-training results in a (mostly) smooth increase on these tasks: $\mathscr{M}_{4}$, the model pre-trained on shuffled 4-grams, trails $\mathscr{M}_{\mathtt{N}}$ by only 1.3 points on average, and even comes within 0.2 points of $\mathscr{M}_{\mathtt{N}}$ on PAWS. We observe a somewhat different pattern on CoLA, where $\mathscr{M}_{2}$ does almost as well

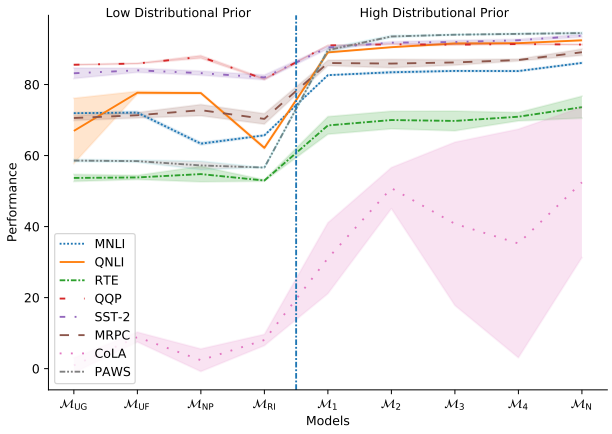

Figure 7: Fine-tuning results on various word shuffled pre-trained models.

[5]http://data.quora.com/First-Quora-Dataset-Release-Question-Pairshttp://data.quora.com/First-Quora-Dataset-Release-Question-Pairs

as $\mathscr{M}_{\mathtt{N}}$ and outperforms $\mathscr{M}_{\mathtt{3}}$ and $\mathscr{M}_{\mathtt{4}}$, though we also observe very high variance across random seeds for this task. Crucially, we observe that $\mathscr{M}_{\mathtt{1}}$ outperforms $\mathscr{M}_{\mathtt{NP}}$ by a large margin. This shows that positional embeddings are critical for learning, even when the word orders themselves are not natural. Overall, these results confirm our hypothesis that RoBERTa's strong performance on downstream tasks can be explained for a large part by the distributional prior.

Where does the models learn word order, in pre-training or in fine-tuning?

There are two possible explanations for the above results: either the tasks do not need word order information to be solved, or any necessary word order information can be acquired during fine-tuning. To examine this question, we permute the word order during fine-tuning as well. Concretely, for each task, we construct a unigram order-randomized version of each example in the fine-tuning training set using $\mathscr{F}_1$. We then fine-tune our pre-trained models on this shuffled data and evaluate task performance.

We observe for some taks (QQP, QNLI, MNLI, SST-2 and MRPC) the accuracy is still significantly high when fine-tuned on shuffled data, suggesting that purely lexical information is quite useful on its own. On the other hand, for the rest of the datasets (CoLA, PAWS, RTE) we observe noticable drops in accuracy when fine-tuned on shuffled data and tested on normal word order data, both for $\mathscr{M}_{\mathtt{N}}$ and for shuffled models $\mathscr{M}_{\mathtt{1}}$ through $\mathscr{M}_{\mathtt{4}}$. This suggests both that word order information is useful for these tasks, and that shuffled models must be learning to use word order information during fine-tuning. Having word order during fine-tuning is especially important for achieving high accuracy on CoLA, RTE (cf. [68]), as well as PAWS, suggesting that these tasks are the most word order reliant.

How does syntactical probes behave when evaluated on word order shuffled pre-trained models?

Since majority of the works investigating syntax representation use probes, we also employ similar mechanisms to test whether these probes are capable to differentiate the pre-trained models trained on shuffled word order from natural word order. In case of parametric probes (i.e probes having a learnable component), we use the Pareto optimality probing framework [69] to investigate syntax using Dependency Parsing as an auxilliary task. By investigating the Dependency parsing task on two datasets (Penn Tree Bank and Universal Dependencies EWT), we observe that the results *follow a similar trend as our downstream fine-tuning results* (Table 4). Surprisingly, $\mathscr{M}_{\mathtt{UG}}$ probing scores seem to be somewhat better than $\mathscr{M}_{\mathtt{1}}$ (though with large overlap in their standard deviations), even though $\mathscr{M}_{\mathtt{UG}}$ cannot learn information related to either word order or co-occurrence patterns.

We also find similar inconsistencies when analyzed with a suite of 10 probing tasks [15] available in the SentEval toolkit [14]. This suite contains a range of semantic, syntactic and surface level tasks. We observe that the $\mathscr{M}_{\mathtt{N}}$ pre-trained model scores better than the unnatural word order models for *only one out of five semantic tasks and in none of the lexical tasks*, unlike the claim of syntax understanding of BERT in the literature [44]. However, $\mathscr{M}_{\mathtt{N}}$ does score higher for two out of three syntactic tasks. Even for these two syntactic tasks, the gap among $\mathscr{M}_{\mathtt{UG}}$ and $\mathscr{M}_{\mathtt{N}}$ is much higher than $\mathscr{M}_{\mathtt{1}}$ and $\mathscr{M}_{\mathtt{N}}$. These results show that while natural word order is useful for at least some probing tasks, the distributional prior of randomized models alone is enough to achieve a reasonably high accuracy on syntax sensitive probing.

Model	UD EWT		PTB	
	MLP	Linear	MLP	Linear
$\mathscr{M}_{\mathtt{N}}$	80.41 +/- 0.85	66.26 +/- 1.59	86.99 +/- 1.49	66.47 +/- 2.77
$\mathscr{M}_{\mathtt{4}}$	78.04 +/- 2.06	65.61 +/- 1.99	85.62 +/- 1.09	66.49 +/- 2.02
$\mathscr{M}_{\mathtt{3}}$	77.80 +/- 3.09	64.89 +/- 2.63	85.89 +/- 1.01	66.11 +/- 1.68
$\mathscr{M}_{\mathtt{2}}$	78.22 +/- 0.88	64.96 +/- 2.32	84.72 +/- 0.55	64.69 +/- 2.50
$\mathscr{M}_{\mathtt{1}}$	69.26 +/- 6.00	56.24 +/- 5.05	79.43 +/- 0.96	57.20 +/- 2.76
$\mathscr{M}_{\mathtt{UG}}$	74.15 +/- 0.93	65.69 +/- 7.35	80.07 +/- 0.79	57.28 +/- 1.42

Table 4: Unlabeled Attachment Score (UAS) (mean and std) on the dependency parsing task (DEP) on two datasets, UD EWT and PTB, using the Pareto Probing framework [69].

From our results so far, it is unclear whether parametric probing meaningfully distinguishes models trained with corrupted sentence order from those trained with normal orders. Thus, we also investigate non-parametric probes [51, 54, 30, 27, 103]. Since these probes do not contain any learnable parameters, they are called "non-parametric". We observe for some datasets [51, 54] the gap between the $\mathscr{M}_{\mathtt{N}}$ and randomization models is relatively large. While some randomization models (e.g., $\mathscr{M}_{\mathtt{2}}$, $\mathscr{M}_{\mathtt{3}}$, and $\mathscr{M}_{\mathtt{4}}$)

performed quite similarly to $\mathscr{M}_{\mathtt{N}}$ according to the parametric probes, they all are markedly worse than $\mathscr{M}_{\mathtt{N}}$ according to the non-parametric ones. This suggests that non-parametric probes identify certain syntax-related modeling failures that parametric ones do not.

4.2.3 Discussion & Conclusion

The assumption that word order information is crucial for any classical NLP pipeline (especially for English) is deeply ingrained in our understanding of syntax itself [10]: without order, many linguistic constructs are undefined. Our fine-tuning results in subsec:glue˙results and parametric probing results in subsec:param˙probing, however, suggests that MLMs do not need to rely much on word order to achieve high accuracy, bringing into question previous claims that they learn a "classical NLP pipeline". These results should hopefully encourage the development of better, more challenging tasks that require sophisticated reasoning, and harder probes to narrow down what exact linguistic information is present in the representations learned by our models.

5 Future Work & Timeline

Until now, most of the work in my doctoral studies have been focused on developing methods to detect the issue of systematicity plaguing neural NLU models. To complete my thesis, I thereby plan to work towards a couple of methods to improve robustness and systematicity of NLU models.

5.1 Unsupervised syntax learning by mutually exclusive training using word order

In our prior work on word order (Section 4), we observed that NLU models are largely distributional - they understand the collection of words in a sentence but have limited understanding on the order of words. This poses a problem - representations of random permutations of the sentence having no grounded meaning will still be identified by the NLU models to contain syntactic and semantic information. Due to this distributional effect, we posit that syntax understanding of NLU models are still primitive, mostly restricted to higher order information. Thus, it is imperative to develop mechanisms to imbibe the required syntactical information within the sentence representation, such that it is systematic. One can use syntactic features such as dependency parses to imbibe information about syntax in the sentence representation using auxilliary supervison loss. In literature, such syntactical information has shown to be effective in downstream tasks, such as Relation Extraction (RE) [25], named entity recognition (NER) [46] and semantic role labeling (SRL) [89]. More recently, syntax trees are used during pre-training of Transformer based models to imbibe better syntactical information by early and late fusion [76]. However, such direct supervision models to imbibe syntactical information is difficult as it requires access to preferably human-annotated syntax parses of sentences, which raises questions on the viability of such approaches for real world applications. Even so, limited studies have been performed to investigate systematicity issues of those models trained with supplementary syntactical signal.

Therefore, we propose an alternate, unsupervised mechanism to imbibe syntax information within the sentence representations by leveraging word order. Concretely, we use an auxilliary objective to the model to recognize correct and incorrect permutations of a given sentence alongside the task objective. Now, in the strictest sense there is only one correct ordering of a sentence which conveys the intended meaning. However, natural language (English) allows for a degree of flexibility in word order. Thus, we plan to leverage the idea of *separable permutations* [88], where a subset of permutations can be treated as positive signal which can be reconstructed from the CCG parse of the given sentence. This auxilliary training loss could potentially inform the model to be systematic in understanding syntax, and thereby reduce the distributional, bag-of-words behavior of the encoder representations.

5.2 Nonsensical data augmentation for better systematic generalization

Systematic generalization is an issue which plagues many NLU tasks, in particular Natural Language Inference (NLI). Generalization to out-of-domain examples is poor [63], and it has been shown that these models leverage the statistical artifacts in NLI datasets, such as SNLI and MNLI [33]. One reason why models tend to overfit on the training data is the exposure bias to specific nouns/verbs/entities during training. When subjected to systematic stress test, the NLU models tend to be brittle as they fail to learn syntax of the training signal by fixating on the rare words and artifacts. Thus, we propose a dynamic data augmentation training scheme for NLU models where we repeat the training examples with word replacements from the same syntactic family. Overall, a sentence might lose its intended meaning (hence, "nonsensical") - however if the same operation is conducted on the premise, the entailment logic remains unchanged. Concretely, given a lexical item in a sentence, we randomly replace that item with another belonging to the same syntactic family, equally in both premise and hypothesis sentences. For empirical reasons we will restrict this replacement to specific family of lexicons (proper nouns, verbs) which typically form as rare elements in the dataset. We also plan to include a probabilistic model for this replacement which replaces lexicons based on their corpus probability to ensure uniformity in training. By systematically replacing the lexicons in a different context one can potentially increase the training data to reduce exposure bias problem. Similar methods have been devised for mitigating gender bias previously in the literature with varying success [55].

5.3 Timeline

Unsupervised syntax learning by mutually exclusive training using word order (1) Investigate CCG parsing to generate separable permutations on the fly, (2) Representational analysis on separable and non separable permutations, (3) Train auxillary loss with either direct supervision or partial gradient based methods such as Meta Learning, (4) Write paper for ACL 2022 or TACL 2022
Nonsensical data augmentation for better systematic generalization (1) Investigate lexicon replacements by using syntactic parsers in a given dataset, (2) Analyze how the distribution of rare elements change in the training corpus by using this kind of replacement, (3) Test on out-of-domain data and NLI stress test sets, such as HANS [56], (4) Write paper for EMNLP 2022.
Thesis preparation and submission Expected defense date: Fall 2022

References

[1] A. D. Baddeley, G. J. Hitch, and R. J. Allen. Working memory and binding in sentence recall. *Journal of Memory and Language*, 2009.

[2] E. M. Bender and A. Koller. Climbing towards NLU: On meaning, form, and understanding in the age of data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5185–5198, Online, July 2020. Association for Computational Linguistics.

[3] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[4] L. Bentivogli, P. Clark, I. Dagan, and D. Giampiccolo. The fifth PASCAL recognizing textual entailment challenge. In *TAC*, 2009.

[5] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26, 2013.

[6] S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning. 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, Sept. 2015. Association for Computational Linguistics.

[7] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020.

[8] J. M. Cattell. The time it takes to see and name objects. *Mind*, os-XI(41):63–65, 01 1886.

[9] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, 2014.

[10] N. Chomsky. *Syntactic structures*. Walter de Gruyter, 1957.

[11] N. Chomsky. *The minimalist program*. Cambridge, Massachusetts: The MIT Press, 1995.

[12] R. Collobert and J. Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167, 2008.

[13] C. Condoravdi, D. Crouch, V. de Paiva, R. Stolle, and D. G. Bobrow. Entailment, intensionality and text understanding. In *Proceedings of the HLT-NAACL 2003 Workshop on Text Meaning*, pages 38–45, 2003.

[14] A. Conneau and D. Kiela. SentEval: An evaluation toolkit for universal sentence representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA).

[15] A. Conneau, G. Kruszewski, G. Lample, L. Barrault, and M. Baroni. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[16] I. Dagan, O. Glickman, and B. Magnini. The pascal recognising textual entailment challenge. In *Machine Learning Challenges Workshop*. Springer, 2005.

[17] I. Dagan, O. Glickman, and B. Magnini. The PASCAL recognising textual entailment challenge. In *Machine Learning Challenges Workshop*, pages 177–190. Springer, 2005.

[18] I. Dasgupta, D. Guo, A. Stuhlmüller, S. J. Gershman, and N. D. Goodman. Evaluating compositionality in sentence embeddings. In *Proceedings of Annual Meeting of the Cognitive Science Society*, 2018.

[19] D. Demszky, K. Guu, and P. Liang. Transforming question answering datasets into natural language inference datasets. *arXiv preprint arXiv:1809.02922*, 2018.

[20] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[21] W. B. Dolan and C. Brockett. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*, 2005.

[22] A. Ettinger. 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, Dec. 2020.

[23] R. Evans and E. Grefenstette. Learning explanatory rules from noisy data. Nov. 2017.

[24] J. A. Fodor and Z. W. Pylyshyn. Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28(1-2):3–71, 1988.

[25] K. Fundel, R. Küffner, and R. Zimmer. Relex—relation extraction using dependency parse trees. *Bioinformatics*, 23(3):365–371, 2007.

[26] D. Giampiccolo, B. Magnini, I. Dagan, and W. B. Dolan. The third PASCAL recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing*, pages 1–9, 2007.

[27] Y. Goldberg. Assessing BERT's Syntactic Abilities. *CoRR*, page 4, 2019.

[28] N. Gontier, K. Sinha, S. Reddy, and C. Pal. Measuring Systematic Generalization in Neural Proof Generation with Transformers. *arXiv:2009.14786 [cs, stat]*, Oct. 2020.

[29] E. Goodwin, K. Sinha, and T. J. O'Donnell. Probing Linguistic Systematicity. *arXiv:2005.04315 [cs]*, Aug. 2020.

[30] K. Gulordava, P. Bojanowski, E. Grave, T. Linzen, and M. Baroni. Colorless green recurrent networks dream hierarchically. *arXiv:1803.11138 [cs]*, Mar. 2018.

[31] A. Gupta, G. Kvernadze, and V. Srikumar. BERT & family eat word salad: Experiments with text understanding. *AAAI*, 2021.

[32] S. Gururangan, S. Swayamdipta, O. Levy, R. Schwartz, S. Bowman, and N. A. Smith. 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, June 2018. Association for Computational Linguistics.

[33] S. Gururangan, S. Swayamdipta, O. Levy, R. Schwartz, S. R. Bowman, and N. A. Smith. Annotation Artifacts in Natural Language Inference Data. *arXiv:1803.02324 [cs]*, Apr. 2018.

[34] R. B. Haim, I. Dagan, B. Dolan, L. Ferro, D. Giampiccolo, B. Magnini, and I. Szpektor. The second PASCAL recognising textual entailment challenge. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, 2006.

[35] Z. S. Harris. Distributional structure. *Word*, 1954.

[36] I. Heim and A. Kratzer. *Semantics in generative grammar*. Blackwell Oxford, 1998.

[37] J. Hewitt and C. D. Manning. 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, June 2019. Association for Computational Linguistics.

[38] G. E. Hinton et al. Learning distributed representations of concepts. In *Proceedings of the eighth annual conference of the cognitive science society*, volume 1, page 12. Amherst, MA, 1986.

[39] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[40] J. Howard and S. Ruder. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*, 2018.

[41] H. Hu, K. Richardson, L. Xu, L. Li, S. Kübler, and L. Moss. OCNLI: Original Chinese Natural Language Inference. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3512–3526, Online, Nov. 2020. Association for Computational Linguistics.

[42] D. A. Hudson and C. D. Manning. Compositional attention networks for machine reasoning. In *International Conference on Learning Representations*, 2018.

[43] D. Hupkes, S. Veldhoen, and W. Zuidema. Visualisation and 'diagnostic classifiers' reveal how recurrent and recursive neural networks process hierarchical structure. *Journal of Artificial Intelligence Research*, 61:907–926, 2018.

[44] G. Jawahar, B. Sagot, and D. Seddah. 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, July 2019. Association for Computational Linguistics.

[45] G. Jawahar, B. Sagot, and D. Seddah. 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, 2019. Association for Computational Linguistics.

[46] R. Jia and P. Liang. Adversarial Examples for Evaluating Reading Comprehension Systems. 2016.

[47] Z. Jie and W. Lu. Dependency-guided lstm-crf for named entity recognition. *arXiv preprint arXiv:1909.10148*, 2019.

[48] D. Jin, Z. Jin, J. T. Zhou, and P. Szolovits. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8018–8025, 2020.

[49] D. Kaushik and Z. C. Lipton. How much reading does reading comprehension require? a critical investigation of popular benchmarks. *arXiv preprint arXiv:1808.04926*, 2018.

[50] B. Lake and M. Baroni. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *International Conference on Machine Learning*, pages 2879–2888, 2018.

[51] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer. 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, July 2020. Association for Computational Linguistics.

[52] T. Linzen, E. Dupoux, and Y. Goldberg. Assessing the ability of LSTMs to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics*, 4:521–535, 2016.

[53] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *arXiv*, 2019.

[54] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv:1907.11692 [cs]*, July 2019.

[55] R. Marvin and T. Linzen. Targeted syntactic evaluation of language models. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1202, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics.

[56] R. H. Maudslay, H. Gonen, R. Cotterell, and S. Teufel. It's All in the Name: Mitigating Gender Bias with Name-Based Counterfactual Data Substitution. *arXiv:1909.00871 [cs]*, Feb. 2020.

[57] R. T. McCoy, E. Pavlick, and T. Linzen. Right for the Wrong Reasons: Diagnosing Syntactic Heuristics in Natural Language Inference. *arXiv:1902.01007 [cs]*, June 2019.

[58] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. In Y. Bengio and Y. LeCun, editors, *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, 2013.

[59] F. Mollica, M. Siegelman, E. Diachek, S. T. Piantadosi, Z. Mineroff, R. Futrell, H. Kean, P. Qian, and E. Fedorenko. Composition is the core driver of the language-selective network. *Neurobiology of Language*, 1(1):104–134, 2020.

[60] S. Muggleton. Inductive logic programming. *New generation computing*, 8(4):295–318, 1991.

[61] A. Naik, A. Ravichander, C. Rose, and E. Hovy. Exploring numeracy in word embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3374–3380, Florence, Italy, July 2019. Association for Computational Linguistics.

[62] A. Naik, A. Ravichander, N. Sadeh, C. Rose, and G. Neubig. 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, Aug. 2018. Association for Computational Linguistics.

[63] Y. Nie, A. Williams, E. Dinan, M. Bansal, J. Weston, and D. Kiela. 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, July 2020. Association for Computational Linguistics.

[64] Y. Nie, A. Williams, E. Dinan, M. Bansal, J. Weston, and D. Kiela. Adversarial NLI: A New Benchmark for Natural Language Understanding. *arXiv:1910.14599 [cs]*, May 2020.

[65] P. Parthasarathi, K. Sinha, J. Pineau, and A. Williams. Sometimes We Want Translationese. *arXiv:2104.07623 [cs]*, Apr. 2021.

[66] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[67] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*, 2018.

[68] F. Petroni, T. Rocktäschel, P. Lewis, A. Bakhtin, Y. Wu, A. H. Miller, and S. Riedel. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*, 2019.

[69] T. M. Pham, T. Bui, L. Mai, and A. Nguyen. Out of Order: How important is the sequential order of words in a sentence in Natural Language Understanding tasks? *arXiv:2012.15180 [cs]*, Dec. 2020.

[70] T. Pimentel, N. Saphra, A. Williams, and R. Cotterell. Pareto Probing: Trading Off Accuracy for Complexity. *arXiv:2010.02180 [cs]*, Nov. 2020.

[71] A. Poliak, J. Naradowsky, A. Haldar, R. Rudinger, and B. Van Durme. 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, June 2018. Association for Computational Linguistics.

[72] X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, and X. Huang. Pre-trained Models for Natural Language Processing: A Survey. *Science China Technological Sciences*, 63(10):1872–1897, Oct. 2020.

[73] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding by generative pre-training. 2018.

[74] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 2019.

[75] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*, 2016.

[76] A. Rogers, O. Kovaleva, and A. Rumshisky. A Primer in BERTology: What we know about how BERT works. *arXiv:2002.12327 [cs]*, Nov. 2020.

[77] D. S. Sachan, Y. Zhang, P. Qi, and W. Hamilton. Do Syntax Trees Help Pre-trained Transformers Extract Information? *arXiv:2008.09084 [cs]*, Jan. 2021.

[78] A. Santoro, D. Raposo, D. G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, and T. Lillicrap. A simple neural network module for relational reasoning. In *Advances in neural information processing systems*, pages 4967–4976, 2017.

[79] E. Scheerer. Early german approaches to experimental reading research: The contributions of wilhelm wundt and ernst meumann. *Psychological Research*, 1981.

[80] K. Sinha, R. Jia, D. Hupkes, J. Pineau, A. Williams, and D. Kiela. Masked Language Modeling and the Distributional Hypothesis: Order Word Matters Pre-training for Little. *arXiv:2104.06644 [cs]*, Apr. 2021.

[81] K. Sinha, P. Parthasarathi, J. Pineau, and A. Williams. UnNatural Language Inference. *arXiv:2101.00010 [cs]*, June 2021.

[82] K. Sinha, P. Parthasarathi, J. Wang, R. Lowe, W. L. Hamilton, and J. Pineau. Learning an Unreferenced Metric for Online Dialogue Evaluation. *arXiv:2005.00583 [cs]*, May 2020.

[83] K. Sinha, S. Sodhani, J. Dong, J. Pineau, and W. L. Hamilton. CLUTRR: A Diagnostic Benchmark for Inductive Reasoning from Text. *arXiv:1908.06177 [cs, stat]*, Sept. 2019.

[84] K. Sinha, S. Sodhani, J. Pineau, and W. L. Hamilton. Evaluating Logical Generalization in Graph Neural Networks. *arXiv:2003.06560 [cs, stat]*, Mar. 2020.

[85] J. Snell and J. Grainger. The sentence superiority effect revisited. *Cognition*, 2017.

[86] J. Snell and J. Grainger. Word position coding in reading is noisy. *Psychonomic bulletin & review*, 26(2):609–615, 2019.

[87] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642, 2013.

[88] S. Sodhani, S. Chandar, and Y. Bengio. On Training Recurrent Neural Networks for Lifelong Learning. *arXiv e-prints*, Nov. 2018.

[89] M. Stanojević and M. Steedman. Formal Basis of a Language Universal. *Computational Linguistics*, 47(1):9–42, Apr. 2021.

[90] E. Strubell, P. Verga, D. Andor, D. Weiss, and A. McCallum. Linguistically-Informed Self-Attention for Semantic Role Labeling. *arXiv:1804.08199 [cs]*, Nov. 2018.

[91] I. Tenney, D. Das, and E. Pavlick. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy, July 2019. Association for Computational Linguistics.

[92] H. Toyota. Changes in the constraints of semantic and syntactic congruity on memory across three age groups. *Perceptual and Motor Skills*, 2001.

[93] M. Tsuchiya. 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, May 2018. European Language Resources Association (ELRA).

[94] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

[95] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[96] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018.

[97] A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. In H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, editors, *NeurIPS*, 2019.

[98] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman. 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, Nov. 2018. Association for Computational Linguistics.

[99] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018.

[100] A. Warstadt and S. R. Bowman. Can neural networks acquire a structural bias from raw linguistic data? In *Proceedings of the 42nd Annual Virtual Meeting of the Cognitive Science Society*, 2020.

[101] A. Warstadt, A. Singh, and S. R. Bowman. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641, 2019.

[102] Y. Wen, J. Snell, and J. Grainger. Parallel, cascaded, interactive processing of words during sentence reading. *Cognition*, 2019.

[103] A. Williams, N. Nangia, and S. Bowman. 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, June 2018. Association for Computational Linguistics.

[104] T. Wolf. Some additional experiments extending the tech report "Assessing BERT's Syntactic Abilities" by Yoav Goldberg. page 7, 2019.

[105] Z. Wu, Y. Chen, B. Kao, and Q. Liu. Perturbed masking: Parameter-free probing for analyzing and interpreting BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4166–4176, Online, July 2020. Association for Computational Linguistics.

[106] Y. Zhang, J. Baldridge, and L. He. Paws: Paraphrase adversaries from word scrambling. *arXiv preprint arXiv:1904.01130*, 2019.

[107] Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27, 2015.