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 , 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 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 .

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 [41]. 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 of semantic understanding , and contains relevant information about relations and world knowledge . 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 [Fodor]. 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 . 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 .

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) [48]

This document does not discuss in detail 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 [15]; or in proof generation, *Measuring Systematic Generalization in Neural Proof Generation* [14], published at Neural Information Processing Systems (NeurIPS) 2020.

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) [46]

- *Masked Language Modeling and the Distributional Hypothesis: Order Word Matters Pre-training for Little*, published at Empirical Methods of Natural Language Processing (EMNLP) 2021 [45]

This document does not discuss concurrent works on analyzing faithfulness and robustness in translations [38] (published at EMNLP 2021), proposed dataset on systematic reasoning on graph neural networks [49], or an unreferenced automatic dialog evaluation framework [47] (published at ACL 2020) conducted during my doctoral studies.

2 Background

2.1 Language Models

2.2 The rise of pre-training

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

[1] Work done as second author.

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 alearning 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.

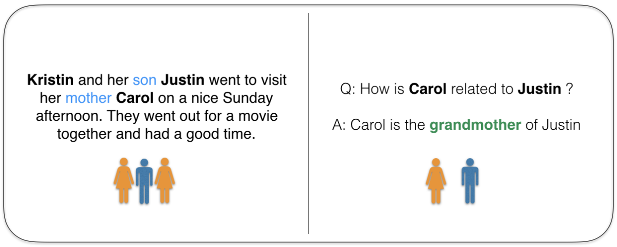

Kristin and her son **Justin** went to visit her mother **Carol** on a nice Sunday afternoon. They went out for a movie together and had a good time.
Q: How is **Carol** related to **Justin** ?
A: Carol is the **grandmother** of Justin

Figure 1: CLUTRR inductive reasoning task

3.2 Dataset Design

In order to design the CLUTRR benchmark, we build upon classic ILP task of inferring kinship relations [21, 33]. 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 two entities, whose relationship is *not* explicitly stated in the story. To generate these stories, we first design a knowledge base (KB) with rules specifying how kinship relations resolve, and we use the following steps to create semi-synthetic stories based on this knowledge base:

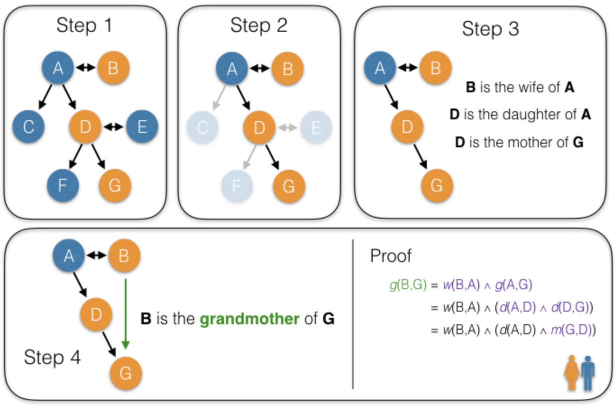

Step 1
A
B
C
D
E
F
G
Step 2
A
B
C
D
E
F
G
Step 3
A
B
D
G
B is the wife of **A**
D is the daughter of **A**
D is the mother of **G**
A
B
D
G
B is the **grandmother** of **G**
Step 4
Proof
g(B,G) = w(B,A) ∧ g(A,G)
= w(B,A) ∧ (d(A,D) ∧ d(D,G))
= w(B,A) ∧ (d(A,D) ∧ m(G,D))

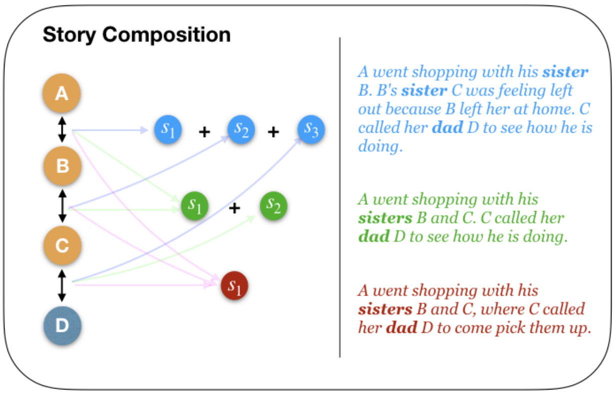
Figure 2: CLUTRR dataset design steps

- **Step 1.** Generate a random kinship graph that satisfies the rules in our KB.
- **Step 2.** Sample a target fact (i.e relation) to predict from the kinship graph
- **Step 3.** Apply backward chaining to sample a set of *k* facts that can prove the target relation (and optionally sample a set of "distracting" or "irrelevant" noise facts)
- **Step 4.** Convert the sampled facts into a natural language story through pre-specified text templates and crowd-sourced paraphrasing.

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. Since workers are given a set of facts logical facts to work from, they are able to combine and split multiple facts across separate sentences and construct diverse narratives (Figure 3). One challenge for data collection via AMT is that the number of possible stories generated by CLUTRR grows combinatorially as the number of supporting facts increases. This makes it infeasible to obtain a large number of paraphrased examples. To circumvent this issue and increase the flexibility of our benchmark, we reuse and compose AMT paraphrases to generate longer stories. In particular, we collected paraphrases for stories containing $k = 1,2,3$ supporting facts and then replaced the entities from these collected stories with placeholders in order to re-use them to generate longer semi-synthetic stories.

An example of a story generated by stitching together two shorter paraphrases is provided below:

> [Frank] went to the park with his father, [Brett]. [Frank] called his brother [Boyd] on the phone. He wanted to go out for some beers. [Boyd] went to the baseball game with his son [Jim].
> Q: What is [Brett] and [Jim]'s relationship?



*A went shopping with his **sister** B. B's **sister** C was feeling left out because B left her at home. C called her **dad** D to see how he is doing.*

*A went shopping with his **sisters** B and C. C called her **dad** D to see how he is doing.*

*A went shopping with his **sisters** B and C, where C called her **dad** D to come pick them up.*

Figure 3: Generation of stories by composition in CLUTRR

Thus, instead of simply collecting paraphrases for a fixed number of stories, we instead obtain a diverse collection of natural language templates that can be programmatically recombined to generate stories with various properties. Please refer to our paper [48] for more details about the data generation process.

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 $k = 2,3,\ldots,6$. 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?.

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.

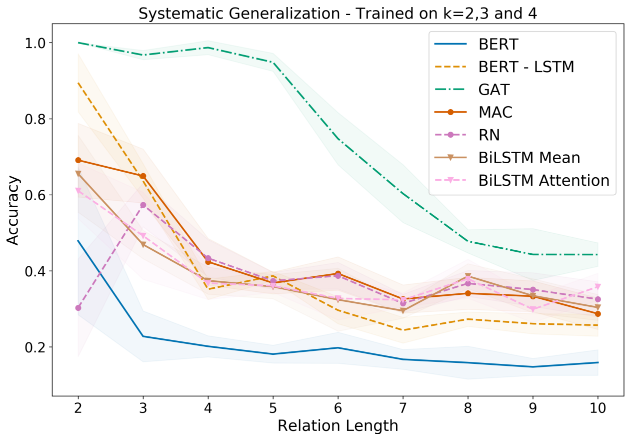


Figure 4: Systematic generalization results on CLUTRR, when trained on stories of length $k = 2,3,4$

Figure 4 illustrates the performance of different NLU models on this generalization task. For NLU models, we consider bidirectional LSTMs [22, 6] (with and without attention), as well as recently proposed models that aim to incorporate inductive biases towards relational reasoning: Relation Networks (RN) [43] and Compositional Memory Attention Network (MAC) [24]. We also use the large pretrained language model, BERT [11], 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) [58]. 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 generalization [27, 52]. 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 5).

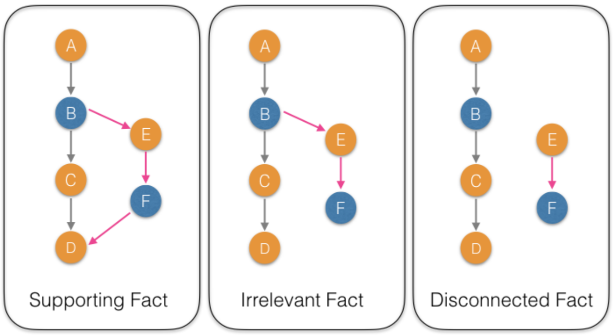


Figure 5: Noise generation methods in CLUTRR

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:

- *Irrelevant facts*: We add a path, which has exactly one shared end-point with the clean path. In this way, this is a *distractor* path, which contains facts that are connected to one of the entities in the target relation, but do not provide any information that could be used to help answer the query.
- *Supporting facts*: We add a path whose two end-points are on the clean path. The facts on this path are noise because they are not needed to answer the query, but they are supporting facts because they can, in principle, be used to construct alternative (longer) reasoning paths that connect the two target entities.
- *Disconnected facts*: We add paths which neither originate nor end in any entity on

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

the clean path. These disconnected facts involve entities and relations that are completely unrelated to the target query.

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:

1. 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.)

2. 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***. [15] 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].

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. [14] In this work, we extend our systematicity analysis to language generation using Transformer Language Models

[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 BERT models to run the same experiments.

(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 [12] , *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 [56] 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.

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.

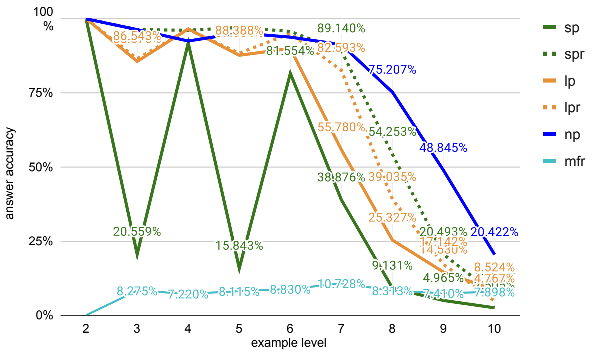


Figure 6: Systematic generalization issues in proof generation

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 [57] models—such as RoBERTa [29], BART [28], and GPT-2 and -3 [40, 4]–on many NLU tasks [60, 59]. The success of these models has prompted serious investigation, leading claims that a language modeling (LM) objective can capture syntactic information [20, 25, 61, 64], with their self-attention layers being capable of surprisingly effective learning [41].

In the following section, we discuss two related works [46, 45] 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, [46] 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., [7]) 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" [18, 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"* ([5, 44, 55, 1, 50, 51, 62], 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 [19]. 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 [8, 9]. This task, often also called Natural Language Inference (NLI; [3], 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.

Main findings. 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 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 [3], MultiNLI [63] and ANLI [36])— and one Chinese one, OCNLI [23]. 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 to construct permuted examples?

For any sentence pair in the NLI dataset, we use a permutation function $\mathscr{F}$ which essentially permutes the word order of a sentence with the restriction that no words maintain their original position. Thus, if a sentence S contains w words, then the total number of available permutations of S are $(w-1)!$. This allows us to inspect *multiple* permutations per hypothesis-premise pair. Concretely, we always test using 100 unique permutations for each hypothesis-premise pair in the NLI dataset. If a given NLU model (RoBERTa/BART/DistilBERT/InferSent/ConvNet/BiLSTM) assigns the gold label to *any one* permutation of an hypothesis-premise pair, we mark that example as correct to compute Maximum Accuracy, Ω_{max}. Now, this metric is strict as it does not allow for any permutation to be processed by the NLU model. We further relax this metric to Ω_{rand}, where a hypothesis-premise is marked correct if more than 1/3rd of its permutations are assigned the gold label by the model. Going down this route, we can define metrics until $\Omega_{1.0}$, where *all* permutations of an example are assigned the gold label. A graphical representation of the metrics are provided in Figure. Additionally, we also compute the probability of acceptance given a base condition: how many permutations are assigned gold label when the example satisfies this base condition. We use the original model accuracy as the base condition - allowing us to compute P^c (probability of correctness)/P^f (probability of flips) - given the example is originally predicted correctly/incorrectly by the model, what is the probability of its permutations to be assigned the gold label.

How many permuted examples does the model accept?

We find Ω_{max} is very high for models trained and evaluated on MNLI (in-domain generalization), reaching **98.7%** on MNLI dev. and test sets (in RoBERTa, compared to $\mathscr{A}$ of 90.6% (table:main). Recall, human accuracy is approximately 92% on MNLI dev., **(author?)** 35). This shows that there exists at least one permutation (usually many more) for almost all examples in D_{test} such that model M predicts the gold label. We also observe high Ω_{rand} at 79.4%, showing that there are many examples for which the models outperform even a random baseline in accepting permuted sentences (see app˙sec:threshold for more Ω values.)

Evaluating out-of-domain generalization with ANLI dataset splits resulted in an Ω_{max} value that is notably higher than $\mathscr{A}$ (89.7% Ω_{max} for RoBERTa compared to 45.6% $\mathscr{A}$). 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 q permutations). For out-of-domain generalization, Ω_{rand} decreases considerably (36.4% Ω_{rand} on A1), which means fewer permutations are accepted by the model. Next, recall that a classic bag-of-words model would have $\mathscr{P}^c = 100$ and $\mathscr{P}^f = 0$. No model performs strictly like a classic bag of words although they do perform somewhat BOW-like ($\mathscr{P}^c >> \mathscr{P}^f$ for all test splits, fig:comb˙plot). We find this BOW-likeness to be higher for certain non-Transformer models, (InferSent) as they exhibit higher $\mathscr{P}^c$ (84.2% for InferSent compared to 70.7% for RoBERTa on MNLI). We extended the experiments to the Original Chinese NLI dataset [23, 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, Ω_{max}, and Ω_{rand}.

How does humans perform on permuted examples?

We expect humans to struggle with unnatural data, given our intuitions and the sentence superiority findings (but see **(author?)** 32). 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

[4] Concurrent work by gupta-etal-2021-bert 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.

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 [10, 39, 16, 34].

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 [45]. 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, **?**) on various corpora with permuted word order while preserving some degree of distributional information, and examine their downstream 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, **?** , plus English Wikipedia) from **?**]. We denote the model trained on the original, un-modified BookWiki corpus as $\mathscr{M}_{\mathbb{N}}$ (for "natural"). We use two types of word order randomization methods: permuting words at the sentence level, and resampling words at the corpus level. Specifically, given a sentence S containing N words, we permute the sentence using a seeded random function $\mathscr{F}_1$ such that no word can remain in its original position. In total, there exist $(N-1)!$ possible permutations of a given sentence. We randomly sample a single permutation per sentence, to keep the total dataset size similar to the original. We extend this function to 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, $\mathscr{M}_{\texttt{1}}$, $\mathscr{M}_{\texttt{2}}$, $\mathscr{M}_{\texttt{3}}$, $\mathscr{M}_{\texttt{4}}$ 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 [45]. 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, $\mathscr{M}_{\texttt{UG}}$ 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 ($\mathscr{M}_{\texttt{NP}}$) and inspect the strength of inductive bias only by using a randomly initialized RoBERTa model ($\mathscr{M}_{\texttt{RI}}$).

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 [**?**] benchmark is a collection of 9 datasets for evaluating natural language understanding systems, of which we use Corpus of Linguistic Acceptability [CoLA, **?**], Stanford Sentiment Treebank [SST, **?**], Microsoft Research Paragraph Corpus [MRPC, **?**], Quora Question Pairs (QQP)[5], Multi-Genre NLI [MNLI, 63], Question NLI [QNLI, **? ?**], Recognizing Textual Entailment [RTE, **? ? ? ?**]. The PAWS task [**?**] 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 word order shuffled models perform remarkably close to the natural word order pre-trained model in case of all tasks. $\mathscr{M}_{\texttt{1}}$, the model pre-trained on completely shuffled sentences, is on average only 3.3 points lower than $\mathscr{M}_{\texttt{N}}$ on the accuracy-based tasks, and within 0.3 points of $\mathscr{M}_{\texttt{N}}$ on QQP. Even on PAWS, which was designed to require knowledge of word order, $\mathscr{M}_{\texttt{1}}$ is within 5 points of $\mathscr{M}_{\texttt{N}}$. Randomizing n-grams instead of words during pre-training results in a (mostly) smooth increase on these tasks: $\mathscr{M}_{\texttt{4}}$, the model pre-trained on shuffled 4-grams, trails $\mathscr{M}_{\texttt{N}}$ by only 1.3 points on average, and even comes within 0.2 points of $\mathscr{M}_{\texttt{N}}$ on PAWS. We observe a somewhat different pattern on CoLA, where $\mathscr{M}_{\texttt{2}}$ does almost as well as $\mathscr{M}_{\texttt{N}}$ and outperforms $\mathscr{M}_{\texttt{3}}$ and $\mathscr{M}_{\texttt{4}}$, though we also observe very high variance across random seeds for this task. Crucially, we observe that $\mathscr{M}_{\texttt{1}}$ outperforms $\mathscr{M}_{\texttt{NP}}$ 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

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

accuracy when fine-tuned on shuffled data and tested on normal word order data, both for $\mathscr{M}_{\texttt{N}}$ and for shuffled models $\mathscr{M}_1$ through $\mathscr{M}_4$. 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. **?**), as well as PAWS, suggesting that these tasks are the most word order reliant.

How does parametric 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 **?**] 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: $\mathscr{M}_1 \approx \mathscr{M}_{\texttt{UG}}$¡ $\mathscr{M}_2$¡ $\mathscr{M}_3$¡ $\mathscr{M}_4$¡ $\mathscr{M}_{\texttt{N}}$ (tab:pareto˙dependency). Surprisingly, $\mathscr{M}_{\texttt{UG}}$ probing scores seem to be somewhat better than $\mathscr{M}_1$ (though with large overlap in their standard deviations), even though $\mathscr{M}_{\texttt{UG}}$ cannot learn information related to either word order or co-occurrence patterns.

We also investigate the suite of 10 probing tasks [**?**] available in the SentEval toolkit [**?**]. This suite contains a range of semantic, syntactic and surface level tasks. **?**] utilize this set of probing tasks to arrive at the conclusion that "*BERT embeds a rich hierarchy of linguistic signals: surface information at the bottom, syntactic information in the middle, semantic information at the top*". We re-examine this hypothesis by using the same probing method and comparing against models trained with random word order. We observe that the $\mathscr{M}_{\texttt{N}}$ 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. However, $\mathscr{M}_{\texttt{N}}$ does score higher for two out of three syntactic tasks. Even for these two syntactic tasks, the gap among $\mathscr{M}_{\texttt{UG}}$ and $\mathscr{M}_{\texttt{N}}$ is much higher than $\mathscr{M}_1$ and $\mathscr{M}_{\texttt{N}}$. 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.

Is the effect similar in case of non-parametric probing?

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 [**? ? ?**] using the formulation of **?**] and **?**]. Since these probes do not contain any learnable parameters, they are called "non-parametric". For each probe, the objective is for a pre-trained model to provide higher probability to a grammatically correct word than to an incorrect one.

We observe for the **?**] and **?**] datasets that the gap between the $\mathscr{M}_{\texttt{N}}$ and randomization models is relatively large. The **?**] dataset shows a smaller gap between $\mathscr{M}_{\texttt{N}}$ and the randomization models. While some randomization models (e.g., $\mathscr{M}_2$, $\mathscr{M}_3$, and $\mathscr{M}_4$) performed quite similarly to $\mathscr{M}_{\texttt{N}}$ according to the parametric probes, they all are markedly worse than $\mathscr{M}_{\texttt{N}}$ 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 [**?**]: 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) [13], named entity recognition (NER) [26] and semantic role labeling (SRL) [54]. More recently, syntax trees are used during pre-training of Transformer based models to imbibe better syntactical information by early and late fusion [42]. 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* [53], 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 [37], and it has been shown that these models leverage the statistical artifacts in NLI datasets, such as SNLI and MNLI [17]. 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 [30].

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 [31], (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] 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.

[4] 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.

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

[6] 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.

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

[8] 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.

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

[10] 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.

[11] 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.

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

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

[14] 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.

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

[16] 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.

[17] 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.

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

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

[20] 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.

[21] 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.

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

[23] 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.

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

[25] 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.

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

[27] 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.

[28] 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.

[29] 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.

[30] 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.

[31] 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.

[32] 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.

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

[34] 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.

[35] N. Nangia and S. R. Bowman. Human vs. muppet: A conservative estimate of human performance on the GLUE benchmark. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4566–4575, Florence, Italy, July 2019. Association for Computational Linguistics.

[36] 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.

[37] 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.

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

[39] 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.

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

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

[42] 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.

[43] 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.

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

[45] 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.

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

[47] 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.

[48] 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.

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

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

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

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

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

[54] 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.

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

[56] 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.

[57] 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.

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

[59] 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.

[60] 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.

[61] 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.

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

[63] 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.

[64] 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.