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 [10]. 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) [15]

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 [3]; or in proof generation, *Measuring Systematic Generalization in Neural Proof Generation* [2], 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) [13]
- *Masked Language Modeling and the Distributional Hypothesis: Order Word Matters Pre-training for Little*, published at Empirical Methods of Natural Language Processing (EMNLP) 2021 [12]

This document does not discuss concurrent works on analyzing faithfulness and robustness in translations [9] (published at EMNLP 2021), proposed dataset on systematic reasoning on graph neural networks [16], or an unreferenced automatic dialog evaluation framework [14] (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 (Compositional Language Understanding with Text-based Relational Reasoning) - 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.

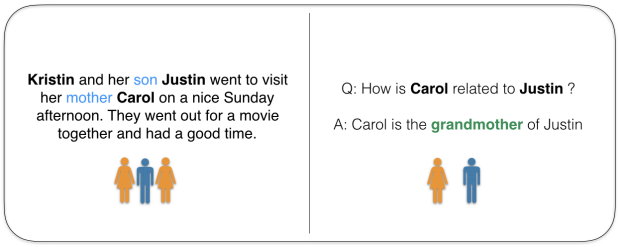


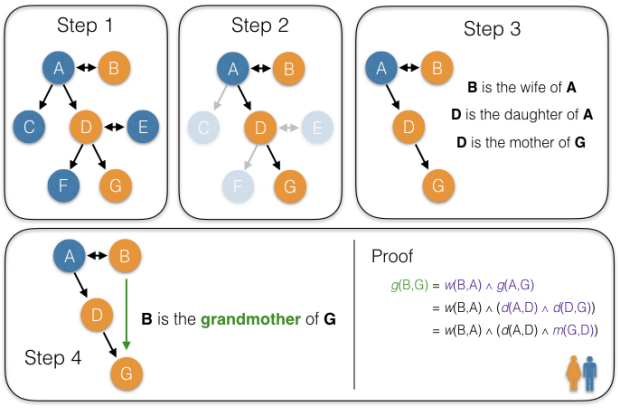
Figure 1: CLUTRR inductive reasoning task

3.2 Dataset Design

In order to design the CLUTRR benchmark, we build upon classic ILP task of inferring kinship relations. 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.** 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?

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 [15] for more details about the data generation process.

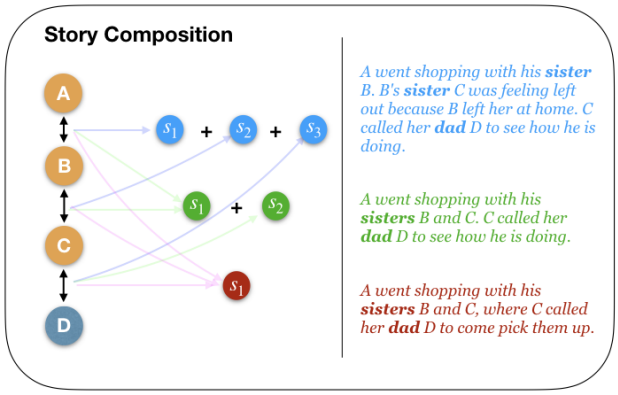


Figure 3: Generation of stories by composition in CLUTRR

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?.

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.

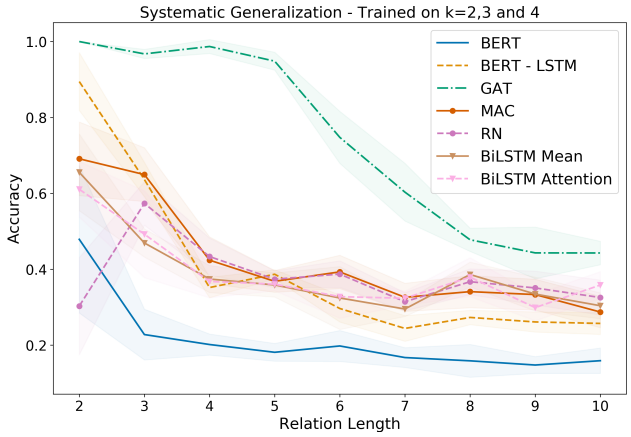


Figure 4: Systematic generalization results on CLUTRR, when trained on stories of length $k = 2, 3, 4$

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

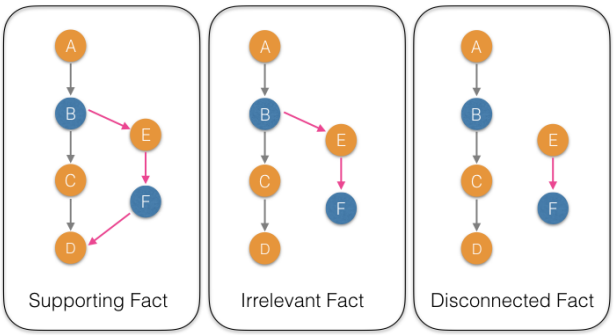


Figure 5: Noise generation methods in CLUTRR

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 the clean path. These disconnected facts

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

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. [3] 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. [2] 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

[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.

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 [Evans], *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 [Vaswani] 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.

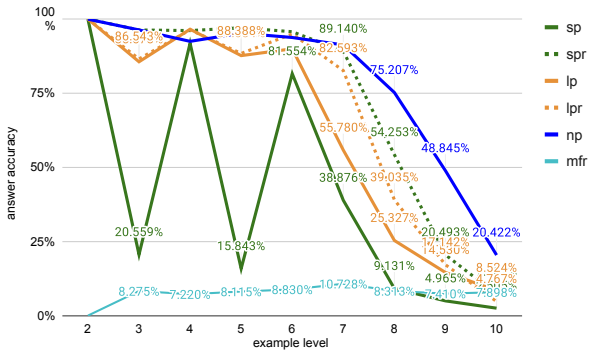


Figure 6: Systematic generalization issues in proof generation

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.

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

Here, we talk about two related contributions, UnNatural Language Inference [13] and pretraining with unnatural languages [12].

4.1 UnNatural Language Inference

4.2 Pre-training with Unnatural Languages

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) [1], named entity recognition (NER) [5] and semantic role labeling (SRL) [18]. More recently, syntax trees are used during pre-training of Transformer based models to imbibe better syntactical information by early and late fusion [11]. 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* [17], 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 [8], and it has been shown that these models leverage the statistical artifacts in NLI datasets, such as SNLI and MNLI [4]. 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 [6].

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 [7], (4) Write paper for EMNLP 2022.
Thesis preparation and submission Expected defense date: Fall 2022

References

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

[2] 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.

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

[4] 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.

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

[6] 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.

[7] 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.

[8] 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.

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

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

[11] 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.

[12] 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.

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

[14] 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.

[15] 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.

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

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

[18] 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.