Right for the Wrong Reasons: Diagnosing Syntactic Heuristics in Natural Language Inference

Anonymous ACL submission

Abstract

A machine learning system can score well on a given test set by relying on heuristics that are effective for frequent example types but break down in more challenging cases. We study this issue within natural language inference (NLI), the task of determining whether one sentence entails another. We hypothesize that statistical NLI models may adopt three fallible syntactic heuristics: the lexical overlap heuristic, the subsequence heuristic, and the constituent heuristic. To determine whether models have adopted these heuristics, we introduce a controlled evaluation set called HANS (Heuristic Analysis for NLI Systems), which contains many examples where the heuristics fail. We find that models trained on MNLI, including the state-of-the-art model BERT, perform very poorly on HANS, suggesting that they have indeed adopted these heuristics. We conclude that there is substantial room for improvement in NLI systems, and that the HANS dataset can motivate and measure progress in this area.

1 Introduction

Neural networks excel at learning the statistical patterns in a training set and applying them to similar test cases. This strength can also be a weakness: statistical learners such as standard neural network architectures are prone to adopting shallow heuristics that succeed for the majority of training examples, instead of learning the underlying generalizations that they are intended to capture. If such heuristics often yield correct outputs, the loss function provides little incentive for a model to learn deeper generalizations.

This issue has been documented across domains in artificial intelligence. In computer vision, for example, neural networks trained to recognize objects are misled by contextual heuristics: they recognize monkeys in typical settings with high accuracy, yet they label a monkey holding a guitar as a human, presumably because guitars only occur with humans in the training set (Wang et al., 2018). Similar heuristics arise in visual question answering systems (Agrawal et al., 2016).

The current paper addresses this issue in the domain of natural language inference (NLI), the task of determining whether a **premise** sentence entails (i.e. implies the truth of) a **hypothesis** sentence (Condoravdi et al., 2003; Dagan et al., 2006; Bowman et al., 2015). As in other domains, neural NLI models have been shown to learn shallow heuristics, in this case based on the presence of specific words (Naik et al., 2018; Sanchez et al., 2018). For example, a model might assign a label of *contradiction* to any input containing the word *not*, since *not* often appears in examples of contradiction.

The focus of our work is on heuristics based on superficial **syntactic** properties. Consider this sentence pair, which has the target label *entailment*:

(1) *Premise:* The judge was paid by the actor.
Hypothesis: The actor paid the judge.

An NLI system that labels this example correctly might do so not by reasoning about the meanings of these sentences, but rather by assuming that the premise entails any hypothesis whose words all appear in the premise (Dasgupta et al., 2018; Naik et al., 2018). Crucially, if the model is using this heuristic, it will predict *entailment* for (2) as well, even though that label is incorrect in this case:

(2) *Premise:* The actor was paid by the judge.
Hypothesis: The actor paid the judge.

We introduce a new evaluation set called HANS (Heuristic Analysis for NLI Systems),[1] designed to diagnose the use of such fallible structural

[1] Anonymized GitHub repository with data and code: https://github.com/hansanon/hans

Heuristic	Definition	Example
Lexical overlap	Assume that a premise entails all hypotheses constructed from words in the premise	**The doctor** was **paid** by **the actor**. $\xrightarrow[\text{WRONG}]{}$ The doctor paid the actor.
Subsequence	Assume that a premise entails all of its contiguous subsequences.	The doctor near **the actor danced**. $\xrightarrow[\text{WRONG}]{}$ The actor danced.
Constituent	Assume that a premise entails all complete subtrees in its parse tree.	If **the artist slept**, the actor ran. $\xrightarrow[\text{WRONG}]{}$ The artist slept.

Table 1: The heuristics targeted by the HANS dataset, along with examples of incorrect entailment predictions that these heuristics would lead to.

heuristics. We target three heuristics, defined in Table 1. While these heuristics often yield correct labels, they clearly do not correspond to the normative rules of natural language inference, and thus fail on certain examples. We design our dataset around such examples, so that models that employ these heuristics are guaranteed to fail on particular subsets of the dataset, rather than simply show lower overall accuracy.

We evaluate four popular NLI models, including the state-of-the-art BERT model (Devlin et al., 2018), on the HANS dataset. All models performed substantially below chance on this dataset, barely exceeding 0% accuracy in most cases. We conclude that their behavior is consistent with the hypothesis that they have adopted these heuristics.

Contributions: This paper has two main contributions. First, we introduce the HANS dataset, an NLI evaluation set that tests specific hypotheses about invalid heuristics that NLI models are likely to learn. Second, we use this dataset to illuminate interpretable shortcomings in state-of-the-art models trained on MNLI (Williams et al., 2018b); these shortcoming may arise from inappropriate model inductive biases, from insufficient signal provided by training datasets, or both. These results indicate that there is substantial room for improvement for these models and datasets, and that HANS can serve as a tool for motivating and measuring improvement in this area.

2 Syntactic Heuristics

We focus on three heuristics: the lexical overlap heuristic, the subsequence heuristic, and the constituent heuristic, all defined in Table 1. These heuristics form a hierarchy: the constituent heuristic is a special case of the subsequence heuristic, which in turn is a special case of the lexical overlap heuristic. Table 2 in the next page gives examples where each heuristic succeeds and fails.

There are two reasons why we expect these heuristics to be adopted by a statistical learner trained on standard NLI training datasets such as SNLI (Bowman et al., 2015) or MNLI (Williams et al., 2018b). First, the MNLI training set contains far more examples that support the heuristics than examples that contradict them:[2]

Heuristic	Supporting Cases	Contradicting Cases
Lexical overlap	2,158	261
Subsequence	1,274	72
Constituent	1,004	58

Since MNLI contains data from multiple genres, we conjecture that the scarcity of contradicting examples is not an idiosyncratic property of a particular genre, but rather a general property of NLI training sets generated in the crowdsourcing approach followed by MNLI. Given this asymmetry, then, we conjecture that our syntactic heuristics would be attractive to statistical learners that do not have strong linguistic priors.

The second reason we might expect current NLI models to adopt these heuristics is that their input representations may make them susceptible to these heuristics. The lexical overlap heuristic disregards the order of the words in the sentence and considers only their identity, so it is likely to be adopted by bag-of-words NLI models (e.g., Parikh et al. 2016). The subsequence heuristic considers linearly adjacent chunks of words, so one might expect it to be adopted by standard RNNs, which

[2] In this table, the lexical overlap counts include the subsequence counts, which include the constituent counts.

Heuristic	Premise	Hypothesis	Label
Lexical overlap heuristic	The banker near the judge saw the actor.	The banker saw the actor.	E
	The lawyer was advised by the actor.	The actor advised the lawyer.	E
	The doctors visited the lawyer.	The lawyer visited the doctors.	N
	The judge by the actor stopped the banker.	The banker stopped the actor.	N
Subsequence heuristic	The artist and the student called the judge.	The student called the judge.	E
	Angry tourists helped the lawyer.	Tourists helped the lawyer.	E
	The judges heard the actors resigned.	The judges heard the actors.	N
	The senator near the lawyer danced.	The lawyer danced.	N
Constituent heuristic	Before the actor slept, the senator ran.	The actor slept.	E
	The lawyer knew that the judges shouted.	The judges shouted.	E
	If the actor slept, the judge saw the artist.	The actor slept.	N
	The lawyers resigned, or the artist slept.	The artist slept.	N

Table 2: Examples of sentences used to test the three heuristics. The *label* column shows the correct label for the sentence pair; *E* stands for *entailment* and *N* stands for *non-entailment*. A model relying on the heuristics would label all examples as *entailment* (incorrectly for those marked as N).

process sentences in linear order. Finally, the constituent heuristic appeals to components of the parse tree, so one might expect to see it adopted by tree-based NLI models (Bowman et al., 2016).

3 Dataset Composition

For each heuristic, we generated five templates for examples that support the heuristic and five templates for examples that refute it. Below is one template for the subsequence heuristic; see Appendix A for a full list of templates.

(3) The N_1 P the N_2 V. $\nrightarrow$ The N_2 V.
The lawyer by the actor ran. $\nrightarrow$ *The actor ran.*

We generated 1,000 examples from each template, for a total of 10,000 examples per heuristic. Since some heuristics are special cases of others, the examples for a given heuristic are ones that can be classified as instances of that heuristic but not of a more narrowly defined heuristic. Specifically, for the lexical overlap examples, every word in the hypothesis is in the premise, but the hypothesis is not a subsequence or constituent of the premise; for the subsequence examples, the hypothesis is a subsequence, but not a constituent, of the premise.

3.1 Dataset Controls

Plausibility: One advantage of generating examples from templates—instead of, e.g., modifying naturally-occurring examples—is that we can ensure the plausibility of all generated sentences. For example, we do not generate cases such as *The student read the book* $\nrightarrow$ *The book read the student*, which could ostensibly be solved using a hypothesis-plausibility heuristic. To achieve this, we drew our core vocabulary from Ettinger et al. (2018), where every noun was a plausible subject of every verb or a plausible object of every transitive verb. Some templates required expanding this core vocabulary; in those cases, we manually curated the additions to ensure plausibility.

Selectional criteria: Some of our example types depend on specific subcategorization frames for a particular verb. For example, (4) requires awareness of the fact that *believed* can take a clause (*the lawyer saw the officer*) as its complement:

(4) The doctor believed the lawyer saw the officer.
$\nrightarrow$ The doctor believed the lawyer.

It is arguably unfair to expect a model to understand this example if it had only ever encountered *believe* with a noun phrase object (e.g., *I believed* ***the man***). To control for this issue, we only chose verbs that appeared at least 50 times in the MNLI training set in all relevant frames.

4 Experimental Setup

Since HANS is designed to probe for structural heuristics, we selected three models that exemplify popular strategies for representing the input sentence: DA, a bag-of-words model; ESIM, which uses a sequential structure; and SPINN,

which uses a parse tree. In addition to these three models, we included BERT, the current state-of-the-art on MNLI. The following paragraphs provide more details on these models.

DA: The Decomposable Attention model (DA; Parikh et al., 2016) uses a form of attention to align words in the premise and hypothesis and to make predictions based on the aggregation of this alignment. It uses no word order information and can thus be viewed as a bag-of-words model.

ESIM: The Enhanced Sequential Inference Model (ESIM; Chen et al., 2017) uses a modified bidirectional LSTM to encode sentences. We use the variant with a sequential encoder, rather than the tree-based Hybrid Inference Model (HIM).

SPINN: The Stack-augmented Parser-Interpreter Neural Network (SPINN; Bowman et al., 2016) is tree-based, creating sentence representations by combining phrases based on a syntactic parse. We use the SPINN-PI-NT variant, which takes a parse tree as an input (rather than learning to parse). During training, we use the parses provided with MNLI; for the HANS evaluation set, our templates include templates for each sentence's parse. These parse templates are based on parses from the Stanford PCFG Parser 3.5.2 (Klein and Manning, 2003), the same parser used to generate the parses in MNLI. Based on manual inspection, this parser generally provided correct parses for HANS examples.

BERT: The Bidirectional Encoder Representations from Transformers model (BERT; Devlin et al., 2018) is a Transformer model that uses attention, rather than recurrence, to process sentences. It is the current state-of-the-art model for MNLI. We use the `bert-base-uncased` pre-trained model and fine-tune it on MNLI.

Implementation and evaluation: For DA and ESIM, we used the implementations from AllenNLP (Gardner et al., 2017). For SPINN[3] and BERT,[4] we used code from the GitHub repositories for the papers about those models.

We trained all models on MNLI. MNLI uses three labels (*entailment*, *contradiction*, and *neutral*). We chose to annotate HANS with two labels only (*entailment* and *non-entailment*) because the distinction between *contradiction* and *neutral* was often unclear for our cases.[5] For evaluating a model on HANS, we took the highest-scoring label out of *entailment*, *contradiction*, and *neutral*; we then translated *contradiction* or *neutral* labels to *non-entailment*. An alternate approach would have been to add the *contradiction* and *neutral* scores to determine a score for *non-entailment*; we found little difference between these approaches, since the models almost always assigned more than 50% of the label probability to a single label.[6]

5 Results

The results of our experiments are shown in Table 3. All models achieved high scores on the MNLI test set, replicating the accuracies found in past work (DA: Gururangan et al. 2018; ESIM: Williams et al. 2018b; SPINN: Williams et al. 2018a; BERT: Devlin et al. 2018).

On the HANS dataset, all models almost always assigned the correct label in the cases where the label is *entailment*, i.e. where the correct answer is in line with the hypothesized heuristics. However, they all performed poorly—with accuracies less than 10% in most cases, when chance is 50%—on the cases where the heuristics make incorrect predictions. Thus, despite their high scores on the MNLI test set, all four models behaved in a way consistent with the use of the heuristics targeted in HANS, and not with the correct rules of inference.

Comparison of models: Both DA and ESIM had near-zero performance across all three heuristics. These models might therefore make no distinction between the three heuristics, but instead treat them all as the same phenomenon, i.e. lexical overlap. Indeed, for DA, this must be the case, as this model does not have access to word order; ESIM does in theory have access to word order information but does not appear to use it here.

SPINN had the best performance on the subsequence cases. This might be due to the tree-

[3] https://github.com/stanfordnlp/spinn; we used the NYU fork at https://github.com/nyu-mll/spinn

[4] https://github.com/google-research/bert

[5] For example, with *The actor was helped by the judge ↛ The actor helped the judge*, it is possible that the actor did help the judge, pointing to a label of *neutral*; yet the premise does pragmatically imply that the actor did not help the judge, meaning that this pair could also fit the non-strict definition of *contradiction* used in NLI annotation.

[6] We also tried training the models on MNLI with *neutral* and *contradiction* collapsed into *non-entailment*; this gave similar results as collapsing after training (Appendix C) .

			Correct: *Entailment*			Correct: *Non-entailment*		
Model	Model class	MNLI	Lexical	Subseq.	Const.	Lexical	Subseq.	Const.
DA	Bag-of-words	0.72	0.99	1.00	0.97	0.01	0.02	0.03
ESIM	RNN	0.77	0.98	1.00	0.99	0.01	0.02	0.01
SPINN	TreeRNN	0.67	0.96	0.96	0.93	0.02	0.06	0.11
BERT	Transformer	0.84	0.95	0.99	0.98	0.16	0.04	0.16

Table 3: Results. The MNLI column reports accuracy on the MNLI test set. The remaining columns report accuracies on 6 sub-components of the HANS evaluation set; each sub-component is defined by its correct label (either *entailment* or *non-entailment*) and the heuristic it addresses.

based nature of its input: since the subsequences targeted in these cases were explicitly chosen not to be constituents, they do not form cohesive units in SPINN's input in the way they do for sequential models. SPINN also outperformed DA and ESIM on the constituent cases, suggesting that SPINN's tree-based representations moderately helped it learn how specific constituents contribute to the overall sentence. Finally, SPINN did worse than the other models on constituent cases where the correct answer is *entailment*. This moderately greater balance between accuracy on entailment and non-entailment cases further indicates that SPINN is less likely than the other models to assume that constituents of the premise are entailed; this harms its performance in cases where that assumption happens to lead to the correct answer.

BERT did slightly worse than SPINN on the subsequence cases, but performed noticeably less poorly than all other models at both the constituent and lexical overlap cases (though it was still far below chance). Its performance particularly stood out for the lexical overlap cases, suggesting that some of BERT's success at MNLI may be due to a greater tendency to incorporate word order information compared to other models.

Analysis of particular example types: In the cases where a model's performance on a heuristic was perceptibly above zero, accuracy was not evenly spread across subcases (for case-by-case results, see Appendix B). For example, within the lexical overlap cases, BERT achieved 39% accuracy on conjunction (e.g., *The actor and the doctor saw the artist* $\nrightarrow$ *The actor saw the doctor*) but 0% accuracy on subject/object swap (*The judge called the lawyer* $\nrightarrow$ *The lawyer called the judge*). Within the constituent heuristic cases, BERT achieved 49% accuracy at determining that a clause embedded under *if* and other conditional words is not entailed (*If the doctor resigned, the lawyer danced* $\nrightarrow$ *The doctor resigned*), but 0% accuracy at identifying that the clause outside of the conditional clause is also not entailed (*If the doctor resigned, the lawyer danced* $\nrightarrow$ *The lawyer danced*).

6 Discussion

Independence of heuristics: Though each heuristic is most closely related to one class of model (e.g., the constituent heuristic is related to tree-based models), the results show that all models tested fail on cases illustrating all three heuristics. This finding is unsurprising since these heuristics are closely related to each other, meaning that an NLI model may adopt all of them, even the ones that do not specifically target that class of model. For example, since the subsequence and constituent heuristics are special cases of the lexical overlap heuristic, all models can fail on cases illustrating all heuristics, because all models have access to individual input words.

Though the heuristics form a hierarchy—the constituent heuristic is a subcase of the subsequence heuristic, which is a subcase of the lexical overlap heuristic—this hierarchy does not necessarily predict the performance of our models. For example, BERT performed worse on the subsequence heuristic than on the constituent heuristic, even though the constituent heuristic is a special case of the subsequence heuristic. Such behavior has two possible causes. First, it could be due to the specific cases we chose for each heuristic: the cases chosen for the subsequence heuristic may be inherently more challenging than the cases chosen for the constituent heuristic, even though the constituent heuristic as a whole is a subset of the subsequence one. Alternately, it is possible for a model to adopt a more general heuristic (e.g., the subsequence heuristic) but to make an exception

for some special cases (e.g., the cases to which the constituent heuristic could apply). That is, a model might learn to use the subsequence heuristic unless the subsequence in question is a constituent.

Assigning blame: architecture or training set? The behavior of a trained model depends on both the training set and the model's architecture. The models' poor results on HANS could therefore arise from architectural limitations, from insufficient signal in the MNLI training set, or from both.

The fact that SPINN did markedly better at the constituent and subsequence cases than ESIM and DA, even though the three models were trained on the same dataset, suggests that MNLI does contain some signal that can counteract the appeal of the syntactic heuristics tested by HANS. SPINN's structural inductive biases allow it to leverage this signal, but the other models' biases do not.

Other sources of evidence suggest that the models' failure is due in large part to insufficient signal from the MNLI training set, rather than the models' representational capacities alone. The BERT model we used (`bert-base-uncased`) was found by Goldberg (2019) to achieve strong results in syntactic tasks such as subject-verb agreement prediction, a task that minimally requires a distinction between the subject and direct object of a sentence (Linzen et al., 2016; Gulordava et al., 2018; Marvin and Linzen, 2018). Despite this evidence that BERT has access to relevant syntactic information, its accuracy was 0% on the subject-object swap cases (e.g., *The doctor saw the lawyer* $\nrightarrow$ *The lawyer saw the doctor*). We believe it is unlikely that our fine-tuning step on MNLI, a much smaller corpus than the corpus BERT was trained on, substantially changed the model's representational capabilities. Even though the model most likely had access to information about subjects and objects, then, MNLI did not make it clear how that information applies to inference. Supporting this conclusion, McCoy et al. (2019) found little evidence of compositional structure in the InferSent model, which was trained on SNLI, even though the same model type (an RNN) was able to learn clear compositional structure when trained on tasks that underscored the need for such structure. These results further suggest that the training set rather than the architecture might be to blame for the models' poor compositional behavior.

Finally, our BERT-based model differed from the other models in that it was pretrained on a massive amount of data on a masking task and a next-sentence classification task, followed by fine-tuning on MNLI, while the other models were only trained on MNLI; we therefore cannot rule out the possibility that BERT's comparative success at HANS was due to the greater amount of data it has encountered rather than any architectural features.

Is the dataset too difficult? It is possible that the models performed poorly because HANS is simply unfairly hard. To assess the difficulty of our dataset, we obtained human judgments on a subset of HANS from 95 participants on Amazon Mechanical Turk as well as 3 expert annotators (linguists who were unfamiliar with HANS: 2 graduate students and 1 postdoctoral researcher). Details of the human experiments are in Appendix E. The average accuracy was 76% for Mechanical Turk participants and 97% for expert annotators. Nangia and Bowman (2019) found that Mechanical Turk participants had an accuracy of 92% on examples from MNLI, indicating that HANS is more challenging for humans than MNLI is. The difficulty of some of our examples is in line with past psycholinguistic work in which humans have been shown to incorrectly answer comprehension questions for some of our subsequence subcases. For example, in an experiment in which participants read the sentence *As Jerry played the violin gathered dust in the attic*, some participants answered *yes* to the question *Did Jerry play the violin?* (Christianson et al., 2001).

Crucially, although Mechanical Turk annotators found HANS to be harder overall than MNLI, their accuracy was similar whether the correct answer was *entailment* (75% accuracy) or *non-entailment* (77% accuracy). The contrast between the balance in the human errors across labels and the stark imbalance in the models' errors (Table 3) indicates that human errors are unlikely to be driven by the heuristics targeted in the current work.

7 What does it take to succeed on HANS?

The failure of the models we tested raises the question of what it would take to do well on HANS. One possibility is that a different type of model would perform better. For example, it is plausible that a model based on hand-coded rules would handle HANS well. However, since most models we tested are in theory capable of handling HANS's examples but failed to do so when trained on MNLI, it is likely that performance could also

	Correct: →			Correct: ↛		
Model	Lex.	Subseq.	Const.	Lex.	Subseq.	Const.
DA	0.94	0.98	0.96	0.26	0.74	1.00
ESIM	0.99	1.00	1.00	1.00	1.00	1.00
SPINN	0.92	1.00	0.99	0.90	1.00	1.00
BERT	1.00	1.00	1.00	1.00	1.00	1.00

Table 4: HANS accuracies for models trained on MNLI plus examples of all 30 categories in HANS.

be significantly improved by training the same architectures on a dataset in which these heuristics are less successful.

To that end, we retrained all four models on the MNLI training set augmented with a dataset structured exactly like HANS (i.e., using the same set of thirty subcases) but containing no specific examples that also appeared in HANS. Our additions comprised 30,000 examples, roughly 8% of the number present in the original MNLI training set (392,702 examples). We then tested these models on HANS; the results are in Table 4. The models performed very strongly, with BERT achieving 100% accuracy across the board; the one exception to the overall success was that the DA model performed poorly on subcases for which a bag-of-words representation was inadequate. Of course, augmenting the training set in this way is not a generally viable solution, since it would require additional augmentation for every element of an infinite set of possible heuristics; but these results do show that, to prevent a model from learning a heuristic, one viable approach is to use a training set that does not support this heuristic.[7]

These results leave open the possibility that these models are simply memorizing HANS's thirty subcases. To address this, we retrained our models on MNLI augmented with subsets of HANS but withholding some subcases, then we tested them on the withheld subcases. The results from one of these experiments, using BERT, are in Table 5. There were some successful cases of transfer; e.g., BERT performed well on the withheld categories with sentence-initial adverbs, regardless of whether the correct label was *non-entailment* or *entailment*. Such successes suggest that these models are able to learn from some specific subcases that they should rule out the broader heuristics; in this case, the non-withheld example types plausibly informed BERT not to indiscriminately follow the constituent heuristic, allowing it to instead base its judgments on the specific adverbs in question (e.g., *certainly* vs. *probably*). However, models did not always transfer successfully; e.g., BERT had 0% accuracy on the withheld category of passives. Thus, though the models do seem to be able to rule out the extremely broad versions of the heuristics and transfer that knowledge to some new cases, they may still fall back to behavior consistent with the heuristics for other cases. For results on all experiments involving withheld categories, see Appendix D.

[7] This experiment is only an initial exploration and leaves open many questions about the conditions under which a model will adopt a heuristic, such as what ratio of supporting examples to contradicting examples is required.

Withheld category	Accuracy
Lexical overlap: Conjunctions (↛) *The doctors saw the author and the tourists.* ↛ *The author saw the tourists.*	0.08
Lexical overlap: Passives (→) *The authors were supported by the actor.* → *The actor supported the authors.*	0.00
Subsequence: NP/Z (↛) *Before the actors presented the doctors arrived.* ↛ *The actors presented the doctors.*	0.57
Subsequence: PP on object (→) *The authors called the judges near the doctor.* → *The authors called the judges.*	0.98
Constituent: Adverbs (↛) *Probably the artists saw the authors.* ↛ *The artists saw the authors.*	0.84
Constituent: Adverbs (→) *Certainly the lawyers advised the manager.* → *The lawyers advised the manager.*	0.96

Table 5: Accuracies for BERT fine-tuned on MNLI augmented with most HANS example categories except withholding the categories in this table.

8 Related Work

8.1 Analyzing trained models

This project relates to an extensive body of research on exposing and understanding weaknesses in models' learned behavior and representations. In the NLI literature, Poliak et al. (2018b) and Gururangan et al. (2018) show that, due to biases in NLI datasets, it is possible to achieve reasonable success at NLI by only looking at the hypothe-

sis. Other recent works address possible ways in which NLI models might use fallible heuristics, focusing on semantic phenomena, such as lexical inferences (Glockner et al., 2018) or quantifiers (Geiger et al., 2018), or biases based on specific words (Sanchez et al., 2018). Our work focuses instead on *structural* phenomena, following the proof-of-concept work done by Dasgupta et al. (2018). Our focus on using NLI to address how models capture structure follows some older work about using NLI for the evaluation of parsers (Rimell and Clark, 2010; Mehdad et al., 2010).

Outside NLI, multiple projects have used classification tasks to understand what linguistic and/or structural information is captured by vector encodings of sentences (e.g., Adi et al., 2017; Ettinger et al., 2018; Conneau et al., 2018). We instead choose the behavioral approach of using task performance on critical cases. Unlike the classification approach, this approach is agnostic to the type of model being used; our dataset could be used to evaluate a symbolic NLI system just as easily as a neural one, whereas classification tasks only work for models with vector representations. This flexibility has motivated the use of NLI as a way to reframe a diverse array of other linguistic tasks (Poliak et al., 2018a; White et al., 2017).

8.2 Structural heuristics

Similar to our lexical overlap heuristic, Dasgupta et al. (2018), Nie et al. (2018), and Kim et al. (2018) also tested NLI models on specific phenomena where word order matters; however, we use a larger set of phenomena to study lexical overlap in general rather than within specific phenomena. Naik et al. (2018) also find evidence that NLI models use a lexical overlap heuristic, but our approach is substantially different from theirs.[8]

Several of our subcases of the subsequence heuristic are inspired by psycholinguistics research on garden path sentences (Bever, 1970; Frazier and Rayner, 1982) and local coherence (Tabor et al., 2004); these works have aims similar to ours but are concerned with the representations used by humans rather than neural networks.

Finally, all of our constituent heuristic subcases depend on the implicational behavior of specific words. Several past works (Pavlick and Callison-Burch, 2016; Rudinger et al., 2018; White et al., 2018; White and Rawlins, 2018) have studied such behavior for verbs (e.g., *He knows it is raining* entails *It is raining*, while *He believes it is raining* does not). We extend that approach by including other types of words with specific implicational behavior, namely conjunctions (*and, or*), prepositions that take clausal arguments (*if, because*), and adverbs (*definitely, supposedly*). MacCartney and Manning (2009) also discuss the implicational behavior of these various types of words within NLI.

8.3 Generalization

Our work suggests that test sets drawn from the same distribution as the training set may be inadequate for assessing whether a model has learned the intended task. Instead, it is also necessary to evaluate on a generalization set that departs from the training distribution. McCoy et al. (2018) found a similar result for the task of question formation; different architectures that all succeeded on the test set failed on the generalization set in different ways, showing that the test set alone was not sufficient to determine what the models had learned. This effect can arise not just from different architectures but also from different initializations of the same architecture (Weber et al., 2018).

9 Conclusions

Statistical learners such as neural networks closely track the statistical regularities in their training sets. This process makes them vulnerable to adopting heuristics that are valid for frequent cases but fail on less frequent ones. We have investigated three such heuristics that we hypothesize NLI models are likely to learn. To evaluate whether NLI models do behave consistently with these heuristics, we have introduced the HANS dataset, on which models using these heuristics are guaranteed to fail. We find that four existing NLI models perform very poorly on HANS, suggesting that their high accuracies on NLI test sets may be due to the exploitation of invalid heuristics rather than deeper understanding of language. These results indicate that, despite the impressive accuracies of these models on standard evaluations, there is still much progress to be made and that targeted, challenging evaluations, such as HANS, are important for determining whether models are learning what they are intended to learn.

[8] Naik et al. (2018) diagnose the lexical overlap heuristic by appending *and true is true* to existing MNLI hypotheses, which decreases lexical overlap but does not change the sentence pair's label. We instead generate new sentence pairs for which the words in the hypothesis all appear in the premise.

References

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2017. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. In *International Conference on Learning Representations*.

Aishwarya Agrawal, Dhruv Batra, and Devi Parikh. 2016. Analyzing the behavior of visual question answering models. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1955–1960. Association for Computational Linguistics.

Thomas G. Bever. 1970. The cognitive basis for linguistic structures. *Cognition and the development of language*, 279(362):1–61.

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

Samuel R. Bowman, Jon Gauthier, Abhinav Rastogi, Raghav Gupta, Christopher D. Manning, and Christopher Potts. 2016. A fast unified model for parsing and sentence understanding. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1466–1477. Association for Computational Linguistics.

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

Kiel Christianson, Andrew Hollingworth, John F Halliwell, and Fernanda Ferreira. 2001. Thematic roles assigned along the garden path linger. *Cognitive Psychology*, 42(4):368–407.

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

Alexis Conneau, Germán Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136. Association for Computational Linguistics.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2006. The PASCAL Recognising Textual Entailment Challenge. In *Proceedings of the First International Conference on Machine Learning Challenges: Evaluating Predictive Uncertainty Visual Object Classification, and Recognizing Textual Entailment*, MLCW'05, pages 177–190, Berlin, Heidelberg. Springer-Verlag.

Ishita Dasgupta, Demi Guo, Andreas Stuhlmüller, Samuel J Gershman, and Noah D Goodman. 2018. Evaluating compositionality in sentence embeddings. In *Proceedings of the 40th Annual Conference of the Cognitive Science Society*, pages 1596–1601, Madison, WI.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:1810.04805*.

Allyson Ettinger, Ahmed Elgohary, Colin Phillips, and Philip Resnik. 2018. Assessing composition in sentence vector representations. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1790–1801. Association for Computational Linguistics.

Lyn Frazier and Keith Rayner. 1982. Making and correcting errors during sentence comprehension: Eye movements in the analysis of structurally ambiguous sentences. *Cognitive Psychology*, 14(2):178–210.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke S. Zettlemoyer. 2017. AllenNLP: A Deep Semantic Natural Language Processing Platform.

Atticus Geiger, Ignacio Cases, Lauri Karttunen, and Christopher Potts. 2018. Stress-testing neural models of natural language inference with multiply-quantified sentences. *arXiv preprint arXiv:1810.13033*.

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

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

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

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. 2018. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112. Association for Computational Linguistics.

Juho Kim, Christopher Malon, and Asim Kadav. 2018. Teaching syntax by adversarial distraction. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 79–84. Association for Computational Linguistics.

Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*.

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

Bill MacCartney and Christopher D Manning. 2009. *Natural language inference*. Ph.D. thesis, Stanford University.

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

R. Thomas McCoy, Robert Frank, and Tal Linzen. 2018. Revisiting the poverty of the stimulus: Hierarchical generalization without a hierarchical bias in recurrent neural networks. In *Proceedings of the 40th Annual Conference of the Cognitive Science Society*, pages 2093–2098, Madison, WI.

R. Thomas McCoy, Tal Linzen, Ewan Dunbar, and Paul Smolensky. 2019. RNNs implicitly implement tensor-product representations. In *International Conference on Learning Representations*.

Yashar Mehdad, Alessandro Moschitti, and Fabio Massimo Zanzotto. 2010. Syntactic/semantic structures for textual entailment recognition. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 1020–1028. Association for Computational Linguistics.

Aakanksha Naik, Abhilasha Ravichander, Norman Sadeh, Carolyn Rose, and Graham Neubig. 2018. Stress test evaluation for natural language inference. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2340–2353. Association for Computational Linguistics.

Nikita Nangia and Samuel R Bowman. 2019. A conservative human baseline estimate for GLUE: People still (mostly) beat machines.

Yixin Nie, Yicheng Wang, and Mohit Bansal. 2018. Analyzing compositionality-sensitivity of nli models. *arXiv preprint arXiv:1811.07033*.

Ankur Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. 2016. A decomposable attention model for natural language inference. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2249–2255. Association for Computational Linguistics.

Ellie Pavlick and Chris Callison-Burch. 2016. Tense manages to predict implicative behavior in verbs. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2225–2229. Association for Computational Linguistics.

Adam Poliak, Aparajita Haldar, Rachel Rudinger, J. Edward Hu, Ellie Pavlick, Aaron Steven White, and Benjamin Van Durme. 2018a. Collecting diverse natural language inference problems for sentence representation evaluation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 67–81. Association for Computational Linguistics.

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

Laura Rimell and Stephen Clark. 2010. Cambridge: Parser evaluation using textual entailment by grammatical relation comparison. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 268–271. Association for Computational Linguistics.

Rachel Rudinger, Aaron Steven White, and Benjamin Van Durme. 2018. Neural models of factuality. 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 731–744. Association for Computational Linguistics.

Ivan Sanchez, Jeff Mitchell, and Sebastian Riedel. 2018. Behavior analysis of NLI models: Uncovering the influence of three factors on robustness. 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 1975–1985. Association for Computational Linguistics.

Whitney Tabor, Bruno Galantucci, and Daniel Richardson. 2004. Effects of merely local syntactic coherence on sentence processing. *Journal of Memory and Language*, 50(4):355–370.

Jianyu Wang, Zhishuai Zhang, Cihang Xie, Yuyin Zhou, Vittal Premachandran, Jun Zhu, Lingxi Xie, and Alan Yuille. 2018. Visual concepts and compositional voting. *Annals of Mathematical Sciences and Applications*, 3(1):151–188.

Noah Weber, Leena Shekhar, and Niranjan Balasubramanian. 2018. The fine line between linguistic generalization and failure in seq2seq-attention models. In *Proceedings of the Workshop on Generalization in the Age of Deep Learning*, pages 24–27. Association for Computational Linguistics.

Aaron Steven White, Pushpendre Rastogi, Kevin Duh, and Benjamin Van Durme. 2017. Inference is everything: Recasting semantic resources into a unified evaluation framework. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 996–1005. Asian Federation of Natural Language Processing.

Aaron Steven White and Kyle Rawlins. 2018. The role of veridicality and factivity in clause selection. In *Proceedings of the 48th Annual Meeting of the North East Linguistic Society,*.

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

Adina Williams, Andrew Drozdov, and Samuel R. Bowman. 2018a. Do latent tree learning models identify meaningful structure in sentences? *Transactions of the Association of Computational Linguistics*, 6:253–267.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018b. 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. Association for Computational Linguistics.