The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)

Analyzing Compositionality-Sensitivity of NLI Models

Yixin Nie,* Yicheng Wang,* Mohit Bansal
Department of Computer Science
University of North Carolina at Chapel Hill
{yixin1, yicheng, mbansal}@cs.unc.edu

Abstract

Success in natural language inference (NLI) should require a model to understand both lexical and compositional semantics. However, through adversarial evaluation, we find that several state-of-the-art models with diverse architectures are over-relying on the former and fail to use the latter. Further, this compositionality unawareness is not reflected via standard evaluation on current datasets. We show that removing RNNs in existing models or shuffling input words during training does not induce large performance loss despite the explicit removal of compositional information. Therefore, we propose a compositionality-sensitivity testing setup that analyzes models on natural examples from existing datasets that cannot be solved via lexical features alone (i.e., on which a bag-of-words model gives a high probability to one wrong label), hence revealing the models' actual compositionality awareness. We show that this setup not only highlights the limited compositional ability of current NLI models, but also differentiates model performance based on design, e.g., separating shallow bag-of-words models from deeper, linguistically-grounded tree-based models. Our evaluation setup is an important analysis tool: complementing currently existing adversarial and linguistically driven diagnostic evaluations, and exposing opportunities for future work on evaluating models' compositional understanding.

1 Introduction

Natural Language Inference (NLI) is a task in which a system is asked to classify the relationship between a pair of premise and hypothesis as one of either entailment, contradiction, or neutral. This task is considered to be the basis of many downstream, higher-level NLP tasks that require complex natural language understanding such as question-answering and summarization. Large annotated datasets such as the Stanford Natural Language Inference Bowman et al. (2015) (SNLI) and the Multi-Genre Natural Language Inference Williams, Nangia, and Bowman (2018) (MNLI) have promoted the development of many different neural NLI models, including encoding and co-attention models using both sequential and recursive representations Nie and Bansal (2017); Choi, Yoo, and Lee (2017); Chen et al. (2017); Gong, Luo, and Zhang (2017); Ghaeini et al. (2018); Parikh et al. (2016); Wang, Hamza, and Florian (2017), all achieving near human-level performance on standard datasets.

Despite their high performance, it is unclear if these models employ semantic understanding of natural language to classify these pairs, or are simply performing word/phrase-level pattern matching. We first conduct investigative experiments with rule-based adversaries and show empirically that seven state-of-the-art NLI models, spanning a variety of architectures, are all unable to recognize simple semantic differences when the word-level information remains unchanged (e.g., the swapping of the subject and object or the addition of the same modifier to different governors).[1] Their failure on these examples contrasts sharply with their high performance on the standard evaluation set, indicating that standard evaluation does not sufficiently assess sentence-level understanding.

Next, to further show the insufficiency of standard evaluation for testing compositional understanding, we conduct two additional experiments in which the compositional information is removed or diluted. Firstly, we train and evaluate the state-of-the-art models with their RNNs replaced by fully-connected layers. Secondly, we train these models with input words shuffled and evaluate them on the original evaluation datasets. In both of the experiments, models are still able to achieve high performance on the standard evaluation datasets, demonstrating that standard evaluation is unable to sufficiently separate models with compositionality understanding capabilities from those without.

We also show the limitation of adversarial evaluation setups by demonstrating their narrow scope: each type of adversary is only capable of testing a model's ability to process one specific type of compositional semantics. We show via adversarial training that success on one type of adversary does not generalize to other types of adversaries, but instead induces errors caused by over-fitting the training data. Thus, while adversarial evaluation is useful for exposing the issue of lexical over-stability, it is not a robust measure of models' ability to understand semantic compositional information.

* indicates equal contribution.

[1] We release our adversaries and compositionality-sensitivity testing setups in the supplementary. The code will be released at https://github.com/easonnie/analyze-compositionality-sensitivity-NLI.

Hence, in order to analyze a model's abilities to reason beyond the lexical level and reveal its sensitivity to compositional differences, we present a 'compositionality-sensitivity' testing setup: we select examples for which a bag-of-words model is misguided (assigns a high probability to one wrong label), this allows us to directly measure how much compositional information the model takes into consideration. By effectively punishing models' over-reliance on lexical features, this testing setup could encourage the development of models that are sensitive to compositional clues regarding the pairs' logical relationships.

We show that although our seven models and their variants have comparable performance on standard evaluation (SNLI and MNLI), our new compositionality-sensitivity testing differentiates them by their capability to capture compositional information (e.g., bag-of-words-like models perform worse than sequential models, which in turn perform worse than syntactic-tree based models). Unlike adversarial evaluation, this setup uses natural examples that are not confined to any specific linguistic context or domain by leveraging existing NLI datasets to the largest extent possible for compositional testing. We hope that this new setup could inspire the collections of datasets that control for lexical-features, to explicitly evaluate the compositional ability of NLI models.

We end by discussing how our compositionality-sensitivity evaluation setup complements other recently proposed evaluation setups. Specifically, while certain linguistically-driven diagnostic datasets are useful in testing for a model's performance in a specific realistic setting, model-driven evaluations such as ours gives insight into why models succeed and fail in these specific linguistic scenarios. The main contributions of this paper are three-fold: 1) we introduce two new adversarial setups that expose current state-of-the-art models' inability to process simple sentence-level semantics when lexical features give no information; 2) we rigorously test and expose the limits of standard and adversarial evaluations; 3) we propose a novel compositionality-sensitivity test that analyzes a model's ability to recognize compositional semantics beyond the lexical level, and show its effectiveness in separating models based on architecture.

2 Models and Motivation

In this section, we start with an overview of the designs of seven recently proposed, high-performing natural language inference models and outline several commonalities that are counter-intuitive. We argue that these shared traits hint at their over-reliance on lexical features for prediction, and they are not modeling the compositional nature of language. This motivates our further investigation in later sections.

2.1 NLI Models

Many different models have been proposed for the NLI task; they all fall under one of two broad categories: sentence encoding-based (sentence encoders) or co-attention based. Sentence encoders independently encode each sentence as a fixed-length vector, and then make a prediction, while co-attention models make inferences by jointly processing the premise and hypothesis. The constraint of independent processing for sentence encoders was put in place to encourage the development of effective fixed-length sentence representations generalizable to higher-level tasks. However, co-attention models with recursive or sequential modeling have achieved much better performance on popular NLI datasets. In this paper, we analyze 7 different models spanning both categories, which are, or were, state-of-the-art in their respective category.[2] We give a brief description of each model below (see Table 1):

Model	SNLI	Type	Representation
RSE	86.47	Enc	Sequential
G-TLSTM	85.04	Enc	Recursive (latent)
DAM	85.88	CoAtt	Bag-of-Words
ESIM	88.17	CoAtt	Sequential
S-TLSTM	88.10	CoAtt	Recursive (syntax)
DIIN	88.10	CoAtt	Sequential
DR-BiLSTM	88.28	CoAtt	Sequential

Table 1: Summary of the models we evaluate, including their performance, type, and sentence representation.'Enc' = Sentence Encoder 'CoAtt' = Co-Attention Model

RSE Residual Sentence Encoder Nie and Bansal (2017) is an encoding-based model that first uses multiple layers of residually-connected BiLSTM to encode the tokens in a sentence and then obtain the sentence's fixed-length representation by max pooling over RNN's hidden states from all timesteps. It is one of the top performing sentence encoders on the Multi-NLI dataset.

G-TLSTM Gumbel-TreeLSTM Choi, Yoo, and Lee (2017) is a recursive encoding-based model that learns latent-tree representations for sentences via reinforcement learning.

DAM Decomposable Attention Model Parikh et al. (2016) is a light-weight co-attention model that performs cross-attention at the word level with decomposable matrices.

ESIM Enhanced Sequential Inference Model Chen et al. (2017) is a strong co-attention model that uses BiLSTM to encode tokens within each sentence, and perform cross-attention on these encoded token representations.

S-TLSTM Syntactic TreeLSTM Chen et al. (2017) is identical to ESIM except it encodes sentence tokens via a TreeLSTM based on the dependency parse instead of sequential BiLSTM. It is the highest performing NLI model with a recursive component.

DIIN Densely Interactive Inference Network Gong, Luo, and Zhang (2017) is a novel co-attention model that extracts phrase-level alignment features using densely connected convolutional layers a word-level interactive matrix.

DR-BiLSTM Dependent Reading Bidirectional LSTM Ghaeini et al. (2018) is a model that modifies on ESIM with a dependent reading mechanism that encodes each sentence conditioned on the other.

We were able to obtain original implementations for RSE, G-TLSTM, S-TLSTM and DIIN. We used our own im-

[2] https://nlp.stanford.edu/projects/snli/
https://repeval2017.github.io/shared/

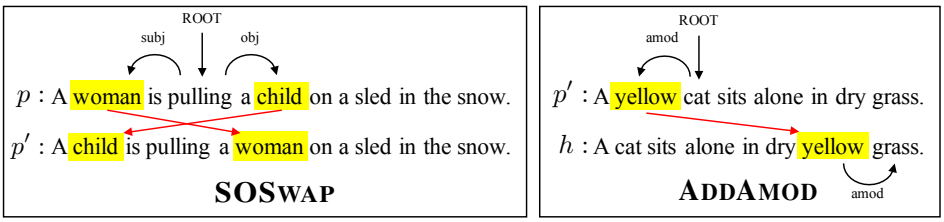

Figure 1: Examples of adversaries generated for our experiments. On the left, we have an example of the SOSWAP adversary, where the swapped subject and object are marked in yellow in p'. On the right, we have an example of the ADDAMOD adversary, where the added adjective modifier is marked in yellow.

Model	SNLI dev	SOSWAP E	SOSWAP **C**	SOSWAP N	ADDAMOD E	ADDAMOD C	ADDAMOD **N**
RSE	86.5	**92.5**	2.1	5.5	**95.2**	0.2	4.6
G-TLSTM	85.9	**97.2**	1.2	1.5	**95.9**	1.2	2.9
DAM	85.0	**99.7**	0.3	0.0	**99.9**	0.0	0.1
ESIM	88.2	**96.4**	2.1	1.5	**85.6**	9.6	4.8
S-TLSTM	88.1	**92.1**	4.4	3.5	**90.4**	1.1	8.5
DIIN	88.1	**84.9**	4.5	10.6	**55.0**	0.4	44.6
DR-BiLSTM	88.3	**89.7**	5.5	4.8	**82.1**	8.9	9.0
Human	-	2	**84**	14	10	2	**88**

Table 2: Model performance on SNLI and % of predictions on the adversarial test sets. E, C, N indicate the classification where E is entailment, C is contradiction and N is neutral. (Note that SOSWAP mostly creates contradictory pairs, while ADDAMOD mostly creates neutral pairs).

plementation for the other three models and were able to achieve comparable results on standard evaluation sets.

2.2 Motivation

Many top-performing sentence encoders (such as RSE) use max-pooling as the final layer to encode the sentences Nangia et al. (2017), and except DIIN, most top-performing co-attention models calculate cross-alignment on the RNN hidden state of each token. These design trends are counterintuitive because max-pooling and attention mechanisms are communicative operations which are not affected by word order, making the RNN layers the only way for the models to capture the compositional structure of sentences. However, past studies have shown that RNNs (especially sequential RNNs) are insufficient for effectively capturing logical compositional structure that is often present in NLI Evans et al. (2018); Nangia and Bowman (2018). These indicate an over-focus on lexical information in neural NLI approaches which is very different from how humans approach the task.

3 Adversarial Evaluation

We test our intuition that the models do not sufficiently capture the compositional nature of sentences by evaluating them on a couple of rule-based adversaries, where we change the semantics of the sentences by perturbing the compositionality without modifying any lexical features. We found that none of the models were able to successfully use the compositional difference to reason with these examples.

3.1 Adversarial Examples

To test our hypothesis that models are over-reliant on word-level information and have limited ability to process compositional structures, we created adversarial test sets composed of pairs of sentences whose logical relations cannot be extracted from lexical information alone. Specifically, we conduct experiments with the following two types of adversarial data in which we change the semantics of the sentence by only modifying its compositional structure:

SOSWAP Adversaries We take a premise from the SNLI dataset, p, that contains a subject-verb-object structure, and create the hypothesis p' by swapping the subject and object. This results in a contradictory pair as the semantic roles of the premise are swapped in the hypothesis. An example is shown on the left side of Fig. 1. We were able to create 971 examples of this type.

ADDAMOD Adversaries In this setup, we take a premise from the SNLI dataset, p, that has at least two different noun entities. We then pick an adjective modifier from the SNLI dataset that has been used to describe both nouns, and create the premise p' by adding the modifier to one of the nouns, and the hypothesis h by adding it to the other. This results in a neutral pair as the hypothesis contains additional information and is neither implied nor refuted by the premise. An example of this is shown on the right side of Fig. 1. We were able to create 1783 examples of this type.

The intuition behind both of the adversaries described above is that, while the semantic difference resulting from compositional change is obvious for humans, the two input sentences will be almost identical for models that take no compositional information into consideration.[3]

3.2 Adversarial Evaluation Results

We trained our 7 models on the SNLI training set and tested them on the adversarial test sets – the results are shown in Table 2. To ensure that the intuitions behind our adversarial generation algorithms were correct, we conducted human evaluation for a sample of 100 examples for each eval-

[3]To create the adversarial data, we used the Stanford Parser Chen and Manning (2014) from CoreNLP 3.8.0 to get the dependency parse of the sentences, on which we apply our strategires.

Model	SNLI			MNLI Matched			MNLI MisMatched		
	Original	BoW	WS	Original	BoW	WS	Original	BoW	WS
RSE	86.47	85.02	–	72.80	70.02	–	74.00	71.10	–
ESIM	88.17	82.37	86.79	76.16	68.98	73.70	76.22	69.77	74.20
DR-BiLSTM	88.28	82.81	86.90	76.90	70.11	73.27	77.49	70.70	73.25

Table 3: The 'Original' columns show results for vanilla models on the resp. validation sets. The 'BoW' column show results for BoW-like variants created replacing their RNNs with fully-connected layers. The 'WS' columns show results for models trained with shuffled input sentences.

	SOSWAP E/C/N	ADDAMOD E/C/N
None	96.4/2.1/1.5	85.6/9.6/4.8
SOSWAP	0.9/99.1/0.0	66.7/26.9/6.5
ADDAMOD	73.1/1.0/25.9	0.3/0.1/99.6

Table 4: The percentages of predicting E/C/N by ESIM with different types of adversarial training, where an underlined number indicates the accuracy on the correct label.

uation set.[4] On both experiments, despite a majority of the examples being marked as non-entail by our human evaluators, the models classified them overwhelmingly as entailment, indicating the models' inability to recognize or process compositional semantic information[5]. The models' poor performance on these adversarial test sets contrasts sharply with their high performance on standard evaluation, raising doubts on the effectiveness and reliability of standard evaluation. However, adversarial evaluation as done here has its own issues. We discuss problems with current evaluation further in the next section.

4 Limitations of Existing Evaluations

In this section, we show that models' performance on standard evaluation does not reflect their compositional understanding capabilities, which we suspect leads to the lack of focus on this type of modeling in the current literature.

4.1 Regular Evaluation Limitations

The gap in model performance between standard evaluation and adversarial evaluation (see Table 2) indicates the limitations of regular evaluation at testing a model's ability to process sentences' compositional structure. More importantly, regular evaluation fails to *separate or differentiate* models that are relying on lexical pattern-matching from those with deeper compositional understanding. To further illustrate this point, we conduct the following two experiments in which we intentionally force the models to be unaware of compositional information by either removing RNN connections in their architectures or by randomizing word order during training.

RNN Replacement: We create strong bag-of-words-like models by replacing RNN layers in RSE, ESIM, and DR-BiLSTM with fully-connected layers, and train them on the standard training set.

Word-Shuffled Training: We train the ESIM and DR-BiLSTM models with the words of the two input sentences shuffled, such that the compositional information is diluted and hard to learn.

The results of these models and their corresponding variants on SNLI, MNLI matched, and MNLI mismatched development set are shown in Table 3, where we see that their performance is not too far from that of their original, recurrent counterparts. To be specific, there is roughly 6-7 points drop in accuracy when RNNs connections are removed and only 2-3 points drop when words are shuffled during training. These counter-intuitive findings indicate that even a model which only considers shallow lexical features is able to get a decent result on standard evaluation, despite using a mechanism that is very different from human reasoning.

4.2 Adversarial Evaluation Limitations

Although, rule-based adversaries were able to expose the models' lack of knowledge of compositional semantics, they have their own limitations and do not serve well as a general analysis tool. Due to the recursive nature of language, there are infinitely many ways for compositional information to affect a sentence's meaning, but each type of rule-based adversary only tests for one specific compositional rule. Thus, success on one type of adversary only demonstrates knowledge of that single rule, and does not indicate general knowledge of compositionality rules. The easiest way to see this is via adversarial training and data-augmentation: we trained the ESIM model with data augmentation from either type of adversaries,[6] and re-evaluated the retrained models on both SOSWAP and ADDAMOD. As shown in Table 4, while adversarial data-augmentation leads to improvement on the same type of adversary, it does not generalize to other types of adversaries. In fact, we see that focusing on one type of adversarial performance may lead to over-fitting that par-

[4]The adversaries are intended to use to highlight models' compositional unawareness and motivate further analysis rather than to be a general-purpose evaluation set. Human evaluation results indicates that the the majority of the data are correct.

[5]Note that DIIN's relatively high performance on ADDAMOD is likely due to its convolutional structure successfully capturing the modifier relationship, but we see that it still fails on adversaries with longer-range dependency requirements such as SOSWAP.

[6]We add 20,000 adversarial examples into training at each epoch. Adv-Training data was created from SNLI training set while Adv-Evaluation set was created from SNLI dev set.

ticular adversary, and hurt overall robustness. For example, in Table 4, we see that adversarial training with SOSWAP leads to an increase in incorrect 'contradiction' predictions on ADDAMOD, and adversarial training with ADDAMOD actually leads to a decrease in performance on SOSWAP while incorrectly increasing 'neutral' predictions.[7] These results indicate that models' success on an enumerable set of adversarial evaluation is still far from validating its general compositional ability.

Thus, we propose an alternative evaluation strategy that leverages existing data to evaluate a model's general compositional understanding capabilities.

5 Compositionality-Sensitivity Testing

In this section, we first formulate the role of compositionality in the context of NLI task, and then propose a compositionality-sensitivity testing setup as an analysis tool to explicit reveal how much compositional information the models take into consideration for inference.

5.1 Problem Formulation

NLI is a complex task with many variables – almost all previous approaches model the task as the distribution $p(y \mid x)$ of the logical relation y conditioned on the pair of input sentences $x = (\mathrm{P}, \mathrm{H})$, where $y \in \{$entailment, contradiction, neutral$\}$ and P, H are the premise and hypothesis, respectively. This conditional distribution is often parameterized by some neural models and trained end-to-end by maximizing the probability of 'ground-truth' label. For the sake of studying models' insensitivity to compositional information, we consider a factorization of the two input sentences as tuples (S_p, Π_h) and (S_h, Π_h), where S_p and S_h are the sets of tokens that make up the premise P and hypothesis H as lexical factors, and Π_p and Π_h are the sets of compositional rules that combine those tokens into meaningful sentences as compositional factors.[8] A perfect modeling of NLI that is capable of taking all lexical and compositional information into account is formalized as Eqn. 1, whereas a entirely bag-of-words (BoW) model is formalized as Eqn. 2.

$$p(y \mid x) = f_\theta(S_p, S_h, \Pi_p, \Pi_h) \quad (1)$$

$$p(y \mid x) = g_\theta(S_p, S_h) \quad (2)$$

The models we discuss are neither perfect models nor entirely BoW models, but rather a combination of both, where they are able to detect and use some lexical features and some semantic rules:

$$p(y \mid x) = \hat{f}_\theta(\tilde{S}_p, \tilde{S}_h, \tilde{\Pi}_p, \tilde{\Pi}_h) \quad (3)$$

where $\tilde{S}_p \subseteq S_p$ and $\tilde{S}_h \subseteq S_h$ are the sets of lexical features of the sentences that the model is capable of using, and similarly $\tilde{\Pi}_p \subseteq \Pi_p$ and $\tilde{\Pi}_h \subseteq \Pi_h$ are sets of compositional rules that the model is capable of using. The issue we explored in previous sections is that current models are overly relying on S_p and S_h, but have limited ability to detect and use Π_p and Π_h. In other words, $\tilde{\Pi}_p \ll \Pi_p$ and $\tilde{\Pi}_h \ll \Pi_h$. For instance, the adversaries we created Sec. 3.2 have sentence pairs which have the same lexical elements but different compositional structures, i.e., $S_p = S_h$ but $\Pi_p \neq \Pi_h$. To an entirely BoW model (Eqn. 2), this looks identical to the scenario where the same sentence is repeated twice. Thus in those cases, inferences necessarily require knowledge of compositional information. This provides intuition into our new evaluation setup: *In order to evaluate models' compositionality-sensitivity, we need to evaluate their performance on data which can not be solved by lexical features alone, i.e., cannot be solved by an entirely BoW model.* We thus seek to evaluate models on a subset of the standard evaluation set that fits this criterion.

5.2 Approximating BoW Model

To obtain such a subset, we must first approximate an entirely BoW model. Specifically, we use a softmax regression classifier that takes in only lexical features for prediction. More formally,

$$v = h(x) \quad (4)$$

$$p(c \mid x) = \frac{\exp(w_c^\top v)}{\sum_{c' \in L} \exp(w_{c'}^\top v)} \quad (5)$$

where h is a function that maps the raw input pair x to its lexical feature vector $v \in \mathbb{R}^d$, $p(c|x)$ is the probability given to label c by the softmax regression classifier. The lexical feature vector v is an indicator vector that contains the following lexical features from the input pair:

- Unigrams appearance within the premise.
- Unigrams appearance within the hypothesis.
- Word pairs (cross-unigrams) where one appears in the premise and the other in the hypothesis.

For unigram and cross-unigram features, we only pick words that are nouns, verbs, adjectives or adverbs to reduce sparsity. We train the regression model on both SNLI and MNLI and use it to approximate an entirely BoW model.

5.3 Lexically-Misleading Score

Since the softmax regression classifier we used is not an entirely BoW model, i.e., it does not capture and use all aspects of lexical semantics. Examples that it predicted incorrectly might still be solvable with the correct lexical information. Thus, to preserve the integrity of our evaluation, we further remove examples that the softmax regression classifier is ambivalent about, and only look at examples where the

[7] While it is true that we can use data augmentation from both types of adversaries to improve performance on both types of evaluations, we can easily come up with a third, different type of adversary (e.g., swapping the verbs between the main sentence and a clause) that is still difficult for the 2-adversarially trained model. Enumerating rule-based adversaries to cover all frequently-used compositional structural changes in a language is prohibitively costly, as generating high-quality (natural and grammatical) data following a single rule already takes tons of time and resources.

[8] Due to the complexity of language, lexical elements are often intertwined with compositional rules and this factorization of p will make Π_p intractable in practice. However, we isolate compositional factors from lexical factors in order to analysis of model behavior.

regression model was confidently wrong, i.e., cases where they were 'misled' by lexical features. We do so because in cases where the regression has insufficient lexical knowledge (e.g., rare/unseen words), it is likely going to give a less confident prediction, whereas in cases where the model was misled, it had the lexical knowledge to make a decision, and hence a wrong prediction indicates the need for compositional knowledge.

Formally, we define the **Lexically-Misleading Score** (LMS) of an NLI datapoint (x, c^*) as:

$$f_{LMS}(x, c^*) = \max_{c \in L \setminus \{c^*\}} p(c \mid x) \quad (6)$$

where c^* is the ground truth label, $p(c \mid x)$ is the probability generated by our regression model, and $L = \{\text{entailment, contradiction, neutral}\}$ is the label set. In other words, f_{LMS} of a data point is the maximum probability the regression gave on an incorrect label. The idea behind LMS is that: *the more lexically misleading an example is, the more confident we are that compositional information is required to solve it.* We thus use LMS to select examples from existing evaluation sets for our evaluation.

5.4 Subsampling and Testing

Given a standard evaluation set and associated 'ground-truth' labels, $D = \{(x_i, c_i)\}_{i=1}^{N}$, we create CS_λ, the compositionality-sensitivity evaluation set of confidence λ:

$$\text{CS}_\lambda = \{(x_i, c_i) \in D \mid f_{LMS}(x_i, c_i) \geq \lambda\}$$

The choice of λ represents a trade-off between being confident about the individual examples' ability to test compositionality-sensitivity and keeping a decent sample size of evaluation data. CS_0 is equivalent to testing on the entire evaluation dataset, whereas $\text{CS}_{0.95}$ (in a 3-way classifier) gives us an extremely small evaluation set (e.g. $\text{CS}_{0.95}$ on SNLI only has 148 examples) with highly misleading lexical features. Empirically, we found that for SNLI and MNLI, $\lambda = 0.7$ gives a good balance between size of the evaluation set and its ability to test compositionality-sensitivity (e.g., $\text{CS}_{0.7}$ on SNLI has 999 examples). Fig. 2 shows examples of sentence pairs with high LMS from the SNLI validation set that were in $\text{CS}_{0.7}$ for SNLI.[9]

5.5 Usage and Limitations

It is worth noting that we do not wish this subset to be used as a benchmark for models to compete on, but rather an analysis tool to explicitly reveal models' compositionality-awareness. Even though the testing setup has its own limitations such as data sparsity and noisiness, it still serves as an initial step to highlight the problem of compositionality understanding (and gain some important insights into models' behaviors, as shown below), which has been largely unexplored in the current neural literature. But more importantly, we hope that this inspires future works on data collection that explicitly address the issue by adding compositionality requirements and lexical-feature balancing into the collection process.

[9]We release the LMS values of the SNLI and MNLI development set in the supplementary materials.

5.6 Evaluation of Existing Models on CS_λ

Table 5 shows the performance of our seven models re-evaluated with CS_λ at different λ values.

General Trend: We see that in general, model performance decreases as λ increases, whereas human performance[10] suffered much less with increasing λ values. This is consistent with our hypothesis that there are significant differences between human-style deep reasoning (with both lexical and compositional knowledge) and inference by current models, which overly relies on lexical information. We also noticed that for all the models on SNLI, MNLI matched, and MNLI mismatch dev set, there is a big gap between the accuracy on the whole dev set and those on $\text{CS}_{0.7}$. This demonstrates that our models have very limited ability to utilize or even recognize compositional information for semantic understanding. These findings indicate the space and need for further research on structured sentence modeling.

Seqential Model vs. Structured Model: The results on $\text{CS}_{0.7}$ differentiates models based on their architectures. More importantly, it explicitly reveals models' compositional understanding which is otherwise largely hidden on the standard evaluation. Specifically, the results for ESIM and S-TLSTM (row 3 and 4) give a clear comparison between sequential and recursive modeling since the two models have the same architecture with the exception that ESIM uses sequential RNN and S-TLSTM uses recursive RNN to encode the sentences. We see that S-TLSTM is better than ESIM on $\text{CS}_{0.7}$, despite ESIM getting better results on all three standard evaluation datasets. This indicates that the recursive model with additional syntactic tree input does in fact induce more compostional understanding ability, which is completely invisible if we merely focus on the results of standard evaluation. Moreover, DIIN (row 5) obtains the best results on all the $\text{CS}_{0.7}$ subsets, substantially surpassing that of DR-BiLSTM (row 6), the most powerful sequential model in the table. This is also consistent with the intuition that DIIN's convolutional network and phrase-level alignment provide much more compositional information than simple RNN-based sequential models. Another interesting fact is the difference in performance between a recursive model trained with explicit external linguistic supervision (S-TLSTM) and one trained via latent tree learning (G-TLSTM). We see that S-TLSTM is able to capture compositional information more effectively than G-TLSTM (row 2), which is consistent with findings from diagnostic datasets regarding recursive modeling Nangia and Bowman (2018).

Necessity of Compositional Information: In the lower side of the table (row 9-13), we evaluate models with either severed RNNs connections or word-shuffled training data. The results represent models with limited compositional accessibility or awareness. As expected, the results on $\text{CS}_{0.7}$ are similar to or even below the majority vote even though their performance on standard evaluation is on the same level as

[10]We approximate human performance by the mechanism proposed by Gong, Luo, and Zhang (2017): we choose one of the annotator labels (out of 5) and compare it against the ground truth. Due to noisy data collection procedure, the actual ceiling of human performance should be much higher than this value.

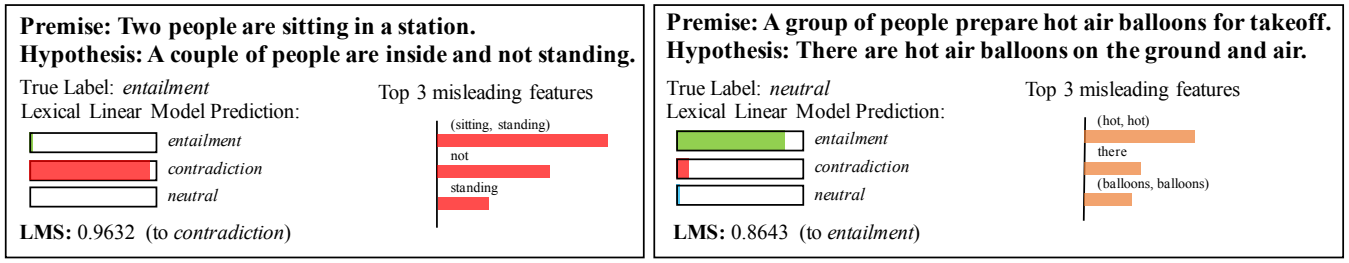

Figure 2: Two examples with high LMS. Correct prediction for the 1st example requires recognizing that 'not standing' and 'sitting' are the same state, rather than focusing on the superficial lexical clues such as 'not' and the cross unigram ('sitting', 'standing') that both mislead to 'contradiction'. For the 2nd example, word-overlap misleads the classifier to predict 'entailment'.

	Model	SNLI				MNLI (Matched)				MNLI (MisMatched)			
		Whole Dev	$CS_{0.5}$	$CS_{0.6}$	**$CS_{0.7}$**	Whole Dev	$CS_{0.5}$	$CS_{0.6}$	**$CS_{0.7}$**	Whole Dev	$CS_{0.5}$	$CS_{0.6}$	**$CS_{0.7}$**
1	RSE	86.47	59.01	55.59	52.73	72.80	48.48	43.57	39.62	74.00	49.30	45.84	40.85
2	G-TLSTM	85.88	57.27	53.68	50.28	70.70	45.32	41.20	38.14	70.81	46.33	42.03	38.87
3	**ESIM**	88.17	62.76	58.58	55.28	76.16	52.76	49.96	48.31	76.22	54.06	51.26	48.32
4	**S-TLSTM**	88.10	64.60	60.57	**57.51**	76.06	53.92	51.54	**48.90**	76.04	55.60	52.40	**50.61**
5	**DIIN**	88.08	64.28	60.57	**57.17**	78.70	59.49	56.12	**54.05**	78.38	59.79	57.44	**53.66**
6	DR-BiLSTM	88.28	62.92	58.50	55.28	76.90	55.26	52.72	50.07	77.49	57.39	55.37	53.04
7	Human	88.32	81.87	80.40	80.76	88.45	86.00	86.03	86.45	89.30	85.53	85.35	84.45
8	Majority Vote	33.82	42.13	42.96	43.27	35.45	36.23	35.04	35.20	35.22	34.22	35.39	34.00
	Models in which compositional information removed or diluted												
9	RSE (BoW)	85.02	52.82	47.93	43.60	70.02	40.69	34.57	31.66	71.10	43.66	38.60	34.30
10	ESIM (BoW)	82.37	48.64	44.18	40.49	68.98	38.59	33.44	30.34	69.77	41.00	35.93	32.32
11	DR-BiLSTM (BoW)	82.81	48.97	44.33	41.38	70.11	37.97	33.07	28.42	70.70	40.73	35.09	30.79
12	ESIM (WS)	86.79	58.41	50.61	45.49	73.70	44.20	41.20	41.09	74.20	49.39	45.39	41.77
13	DR-BiLSTM (WS)	86.90	58.46	50.39	44.77	73.27	45.77	41.20	37.85	73.25	46.33	42.03	38.26

Table 5: Results of models, human, and majority-vote baseline on different levels of compositionality-sensitivity testing. Results of models with limited compositional information are in the bottem on the table.

that of the original models, indicating that compositionality understanding is required to obtain a good result on $CS_{0.7}$.

6 Related Work and Discussion

Over-Stability: Jia and Liang (2017) used adversarial evaluation to show that models trained on the Stanford QA Dataset Rajpurkar et al. (2016) were reliant on syntactic similarity for answering, revealing the over-stability of QA models. With similar motivation, we study the task of NLI by showing that models are overly focused on lexical features and have limited ability of compositionality.

Existing Analysis on NLI: Previous work on analyzing NLI models Glockner, Shwartz, and Goldberg (2018); Carmona, Mitchell, and Riedel (2018) has focused on models' limited ability in identifying lexical semantics that were rare or unseen during training. Our work complements theirs by demonstrating models' limited understanding of compositionality encoded in the sentences. Gururangan et al. (2018), Poliak et al. (2018b) and Tsuchiya (2018) concurrently showed hypothesis bias in NLI and RTE datasets. In particular, Gururangan et al. (2018) also proposed to evaluate models on a harder and better subset of standard evaluation. We instead focus on exposing models' compositionality-insensitivity by selecting our evaluation dataset based on LMS (lexically-misleading score).

Compositionality: Nangia and Bowman (2018) introduce a dataset to study the ability of latent tree models. Evans et al. (2018) introduce a dataset of logical entailments for measuring models' ability to capture the structure of logical expressions against an entailment prediction task. Dasgupta et al. (2018) study the inference behavior of models using sentence embedding on a compositional comparisons dataset. However, we conduct a rigorous study on compositionality-sensitivity, covering a broader range of NLI models and show how to use a filtered subset of existing NLI datasets to test models' compositional ability.

Linguistic Diagnostic Evaluation: Multiple linguistic diagnostic datasets have been published to test NLI models' ability to process certain linguistic phenomena such as coreference, double negation, etc. Williams, Nangia, and Bowman (2018); Nangia et al. (2017); Poliak et al. (2018a); Wang et al. (2018). These datasets are helpful in that they explore the potential usefulness of existing models by demonstrating their abilities in specific scenarios. However, the way models approach language might not have any linguistic grounding. Consider an example where the premise is 'We can't not go to sleep.' and hypothesis is 'We have to go to sleep.' Understanding the first sentence should need compositionality for

processing the double negation. However, given that the example's LMS score is only 0.2376 (which means our BoW regression model was able to solve this correctly), models can solve this particular example via a lexical feature shortcut. Thus, a model resolving of a specific linguistic problem does not necessarily indicate its understanding of the linguistic rule and its generalizability to other compositionality rules. Thus, our work complements linguistic diagnostic datasets well, since a model performing well on both our evaluation and the linguistic diagnostic setup is likely using compositional rules (i.e., human-like reasoning) rather than other pattern-matching procedures to obtain seemingly-compositional behavior.

7 Conclusion

In this paper, we show that current NLI models achieve misleadingly high results on standard evaluation due to its inability to test models' compositional semantic understanding. We further show that typical adversarial evaluation is also limited in terms of evaluating generalizability. Therefore, to encourage the design of models with general understanding capabilities, we propose our compositionality-sensitivity testing that evaluates using compositional information which is not confined to any specific type. Our work complements other recent advancements in evaluation in the community and we hope that this not only encourages the development of structured compositional-aware models, but also highlights the need of more lexical-feature-controlled data collections for semantically demanding tasks (e.g. NLI), such that they will require not only distributional semantics, which is often captures via large scale unsupervised learning, but also compositional semantics, which tend to be overlooked but a harder problem in the community.

Acknowledgments

We thank the reviewers for their helpful comments. This work was supported by faculty research awards from Verisk, Google, and Facebook.

References

Bowman, S. R.; Angeli, G.; Potts, C.; and Manning, C. D. 2015. A large annotated corpus for learning natural language inference. In *EMNLP*.

Carmona, V. I. S.; Mitchell, J.; and Riedel, S. 2018. Behavior analysis of nli models: Uncovering the influence of three factors on robustness. *NAACL*.

Chen, D., and Manning, C. 2014. A fast and accurate dependency parser using neural networks. In *EMNLP*.

Chen, Q.; Zhu, X.; Ling, Z.-H.; Wei, S.; Jiang, H.; and Inkpen, D. 2017. Enhanced lstm for natural language inference. In *ACL*.

Choi, J.; Yoo, K. M.; and Lee, S.-g. 2017. Learning to compose task-specific tree structures. In *AAAI*.

Dasgupta, I.; Guo, D.; Stuhlmüller, A.; Gershman, S. J.; and Goodman, N. D. 2018. Evaluating compositionality in sentence embeddings. *arXiv preprint arXiv:1802.04302*.

Evans, R.; Saxton, D.; Amos, D.; Kohli, P.; and Grefenstette, E. 2018. Can neural networks understand logical entailment? *ICLR*.

Ghaeini, R.; Hasan, S. A.; Datla, V.; Liu, J.; Lee, K.; Qadir, A.; Ling, Y.; Prakash, A.; Fern, X. Z.; and Farri, O. 2018. Dr-bilstm: Dependent reading bidirectional lstm for natural language inference. *NAACL*.

Glockner, M.; Shwartz, V.; and Goldberg, Y. 2018. Breaking nli systems with sentences that require simple lexical inferences. *NAACL*.

Gong, Y.; Luo, H.; and Zhang, J. 2017. Natural language inference over interaction space. *ICLR*.

Gururangan, S.; Swayamdipta, S.; Levy, O.; Schwartz, R.; Bowman, S. R.; and Smith, N. A. 2018. Annotation artifacts in natural language inference data. *NAACL*.

Jia, R., and Liang, P. 2017. Adversarial examples for evaluating reading comprehension systems. *EMNLP*.

Nangia, N., and Bowman, S. R. 2018. Listops: A diagnostic dataset for latent tree learning. *ACL-SRW*.

Nangia, N.; Williams, A.; Lazaridou, A.; and Bowman, S. R. 2017. The repeval 2017 shared task: Multi-genre natural language inference with sentence representations. *RepEval*.

Nie, Y., and Bansal, M. 2017. Shortcut-stacked sentence encoders for multi-domain inference. *RepEval*.

Parikh, A. P.; Täckström, O.; Das, D.; and Uszkoreit, J. 2016. A decomposable attention model for natural language inference. In *EMNLP*.

Poliak, A.; Haldar, A.; Rudinger, R.; Hu, J. E.; Pavlick, E.; White, A. S.; and Van Durme, B. 2018a. Towards a unified natural language inference framework to evaluate sentence representations. *arXiv preprint arXiv:1804.08207*.

Poliak, A.; Naradowsky, J.; Haldar, A.; Rudinger, R.; and Van Durme, B. 2018b. Hypothesis only baselines in natural language inference. **SEM*.

Rajpurkar, P.; Zhang, J.; Lopyrev, K.; and Liang, P. 2016. Squad: 100,000+ questions for machine comprehension of text. *EMNLP*.

Tsuchiya, M. 2018. Performance impact caused by hidden bias of training data for recognizing textual entailment. *LREC*.

Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; and Bowman, S. R. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *EMNLP*.

Wang, Z.; Hamza, W.; and Florian, R. 2017. Bilateral multi-perspective matching for natural language sentences. *IJCAI*.

Williams, A.; Nangia, N.; and Bowman, S. R. 2018. A broad-coverage challenge corpus for sentence understanding through inference. *NAACL*.