

Linguistically-Informed Transformations (LIT[🔥]): A Method for Automatically Generating Contrast Sets

{Chuanrong Li[♡], Lin Shengshuo[♡], Leo Z. Liu[♣], Xinyi Wu[♡], Xuhui Zhou[♡] }*,
Shane Steinert-Threlkeld[♡]

[♡]Department of Linguistics, University of Washington

[♣]Paul G. Allen School of Computer Science & Engineering, University of Washington
{licor, shuo2019, xywu, xuhui, shanest}@uw.edu,
zeyuliu2@cs.washington.edu

Abstract

Although large-scale pretrained language models, such as BERT and RoBERTa, have achieved superhuman performance on in-distribution test sets, their performance suffers on out-of-distribution test sets (e.g., on contrast sets). Building contrast sets often requires human-expert annotation, which is expensive and hard to create on a large scale. In this work, we propose a Linguistically-Informed Transformation (LIT) method to automatically generate contrast sets, which enables practitioners to explore linguistic phenomena of interests as well as compose different phenomena. Experimenting with our method on SNLI and MNLi shows that current pretrained language models, although being claimed to contain sufficient linguistic knowledge, struggle on our automatically generated contrast sets. Furthermore, we improve models' performance on the contrast sets by applying LIT to augment the training data, without affecting performance on the original data.¹

1 Introduction

Large-scale pretrained language models have given remarkable improvements to a wide range of NLP tasks (Peters et al., 2018; Howard and Ruder, 2018; Devlin et al., 2019; Liu et al., 2019; Radford et al., 2019). However, the results are questionable, since those models take advantage of lexical cues (and other heuristics) in the datasets, which can make them right for wrong reasons (Gururangan et al., 2018; McCoy et al., 2019). Therefore, the concept of evaluating models on contrast sets (Gardner et al., 2020) and the creation of generalization tests (Kaushik et al., 2020) is critical for building a robust NLP system. Those test sets are usually

*Equal Contribution from bracket authors

¹See code in https://github.com/leo-liuzy/LIT_auto-gen-contrast-set

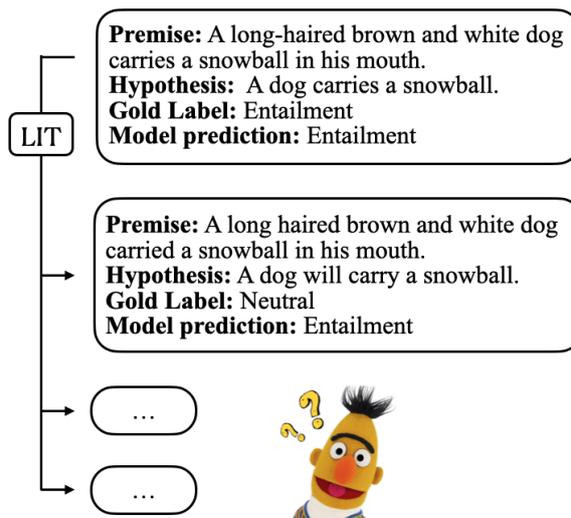


Figure 1: Example of BERT making wrong prediction on LIT-transformed data but correct prediction on the original datum. The detailed transformed datum includes a premise modified to past tense and a hypothesis with future tense. The true label correspondingly changes to *neutral*. LIT also generates multiple transformation results at once for a single original datum; we include only one detailed example here for simplicity of the illustration.

manually created, which requires significant human effort, and so is hard to do on a large scale.

In this work, we propose Linguistically-Informed Transformations (LIT) to create contrast sets automatically. Our method can perturb the original examples and generate various types of contrastive examples, with a wide choice of linguistic phenomena. Furthermore, our tool supports compositional generalization tests. Namely, researchers can choose transformations from a set of basic linguistic phenomena and modify original sentences with an arbitrary combination of those basic transformations.

To demonstrate the utility of LIT, we focus on the natural language inference (NLI) task, a central

task to many NLP applications. We apply LIT to generate contrast sets for SNLI (Bowman et al., 2015) and MNLi (Williams et al., 2018) using seven linguistic phenomena. Human experts’ rating show that our generated data is high-quality for basic transformations and for most of the compositional transformations. See Appendix B for more details.

With our generated contrast sets, we show that pretrained language models, despite having ‘seen’ huge quantities of raw text data, fail on simple linguistic perturbations. As shown with an example in Figure 1, ‘decoupling’ tenses of the premise and hypothesis breaks BERT’s prediction. Our analysis not only shows the inadequate coverage of SNLI and MNLi datasets but also reveals the deficiency of current pretraining-and-finetuning paradigms. Compared to previous work showing that BERT is not robust and fails to generalize on out-of-distribution test sets (McCoy et al., 2019; Zhou et al., 2019; Jin et al., 2019b), our method provides a more fine-grained picture showing on which phenomenon the models fail. In summary, our contributions are:

- We provide a method for automatically generating phenomenon-specific contrast sets, which helps NLP practitioners better understand pre-trained language models.
- We further apply LIT to augment SNLI and MNLi training data, which improves models’ performance on out-of-distribution test sets without sacrificing the models’ performance on the in-distribution test set.
- We demonstrate that, in the current pretraining paradigm, traditional linguistic methods are valuable for their ability to measure and promote robustness and consistency in data-driven models.

After discussing several areas of related work in Section 2, we describe LIT in step-by-step detail (Section 3). We then apply LIT to SNLI and MNLi (4.1) before evaluating BERT and RoBERTa on both simple (4.2) and compositional (4.4) transformations. We conclude (Section 5) by discussing limitations of LIT and future directions.

2 Related Work

NLI Model Diagnosis Our work builds on works diagnosing and improving NLI models with

automatically augmented instances (McCoy et al., 2019; Min et al., 2020). While most of these works apply simple methods such as templates to generate new instances, which limits the phenomena covered, our method has a wider coverage and can be easily extended.

Contrast Sets Contrast sets (Gardner et al., 2020) serve to evaluate a models’ true capabilities by evaluating on out-of-distribution data since previous in-distribution test sets often have systematic gaps, which inflate models’ performance on a task (Gururangan et al., 2018; Geva et al., 2019). The idea of contrast sets is to modify a test instance to a minimum degree while preserving the original instance’s syntactic/semantic artifacts and changing the label. Typically, the authors of the dataset create the contrast set manually. We show that a precision grammar, namely ERG (Copestake and Flickinger, 2000), can be used to automate this process while preserving the authors’ benefit of choosing the perturbations of interest.

Adversarial Datasets Another line of work addressing the problem of current models’ super-human performance on in-distribution test sets focuses on adversarial methods. Bras et al. (2020) uses an adversarial filtering algorithm to reduce spurious bias in the dataset to avoid models relying on such patterns. Dinan et al. (2019) shows that a human-in-the-loop adversarial training framework significantly improves models’ robustness. And Jin et al. (2019a) shows that current pretrained language models are not robust under simple lexical manipulations. Adversarial methods generate test instances automatically, which can be applied to augment the training data (Jin et al., 2019a; Dinan et al., 2019). However, these adversarial methods introduce specific models in the loop, which might also bias the test set.

3 Generating Contrast Sets

We propose a new Linguistically-Informed Transformation (LIT) method for large-scale automatic generation of contrast sets. LIT 1) parses the input sentence for both syntax and semantics, 2) produces transformed syntax and semantics for each linguistic phenomenon, 3) generates perturbed sentences corresponding to the transformed syntax/semantics, 4) and selects the best surface sentence for each phenomenon. The full pipeline is shown in Figure 2. Note that we expand the definition of

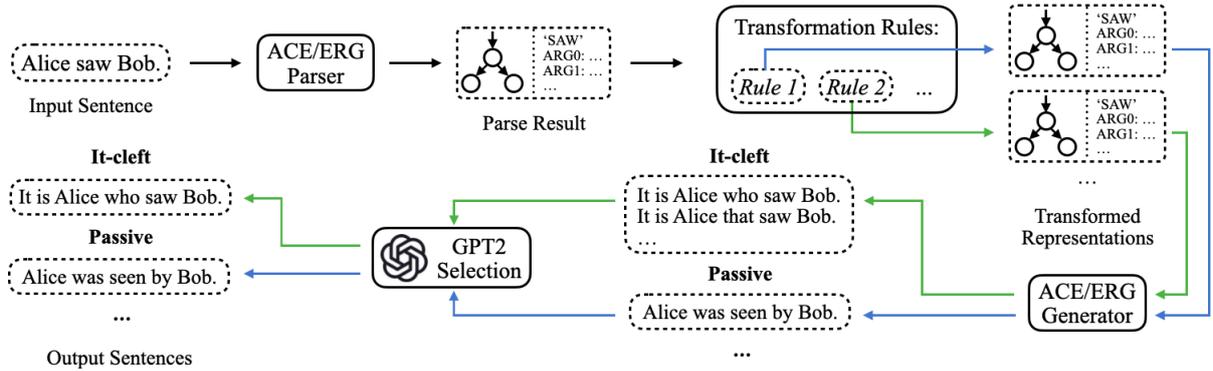


Figure 2: General pipeline of LIT system exemplified with one input sentence. The parse result includes both syntax and semantics. The transformation rules produce one transformed representation per phenomenon. A set of sentences, all grammatical according to ERG, is generated for each transformed representation. One sentence per phenomenon is selected as the final output sentence. We include two “Rule”s for illustration purpose; LIT includes more transformation rules and can be extended for more phenomena.

contrast sets in Gardner et al. (2020). We not only apply our generated contrast sets for evaluation but also for augmentation. We also no longer restrict that the perturbations necessarily lead to the change of the labels.

LIT contains seven phenomenon-specific transformation rules for modifying the parse results and can be further extended; LIT also allows the composition of different transformation rules for complicated perturbations involving multiple linguistic phenomena.

3.1 Parse and Generation

LIT utilizes an existing grammar implementation for parsing and generation, namely the English Resource Grammar (ERG, Copestake and Flickinger 2000). ERG is a linguistically motivated broad-coverage grammar for English in the Head-Driven Phrase Structure Grammar framework (HPSG, Pollard and Sag 1994; Sag et al. 2003) covering 82.6% of sentences in Wall Street Journal (WSJ) sections in the Penn Treebank (Marcus et al., 1993). ERG is processing-neutral, meaning that it is not limited to either parsing or generation, and can handle both with a grammar processor. In this work, we use the ACE parser² as the processor for ERG grammar.

3.2 Transformation

The core part and original contributions of our LIT system are the transformation rules; each rule modifies the parse results from ERG and the ACE parser for one linguistic phenomenon. An ERG parse result includes an HPSG syntax tree and a seman-

tic representation in Minimal Recursion Semantics (MRS, Copestake et al. 2005). An MRS representation consists of a bag of *elementary predicates* (EPs), each with a *handle* for reference, a set of *handle constraints* that specify relations between handles, a *top* indicating the topmost EP, and an *index* variable for the event described by the entire sentence. Every variable has a set of features such as tense and numbers indicating the properties of the entities or the events.

In what follows, we illustrate the application of the transformation rule for it-cleft construction applied to the sentence *Alice saw Bob.*; see Appendix A for a full list of rules.

- (1) Original parse result:

```
[ TOP: h0
  INDEX: e2
  [ e SF: prop TENSE: past ... ]
  RELS: < [ proper_q LBL: h4 ARG0: x3 ... ]
          [ named LBL: h7 ARG0: x3 CARG: "Alice" ]
          [ _see_v_1 LBL: h1 ARG0: e2 ARG1: x3 ARG2: x9 ... ]
          [ proper_q LBL: h10 ARG0: x9 ... ]
          [ named LBL: h13 ARG0: x9 CARG: "Bob" ] >
  HCONS: < h0 qeq h1 h5 qeq h7 h11 qeq h13 > ]
```

TOP: label of topmost EP
INDEX: the variable associated with the sentential event
RELS: bag of EPs
LBL: label variable for the EP
HCONS: constrains between labels for EPs; *qeq* denotes a scoping relation

- (2) Inserting it-cleft EP

```
[ _be_v_itcleft LBL: h14 ARG0: e15 ARG1: x3 ARG2: h1 ]
```

- (3) Connecting top handle *h0* to it-cleft EP handle (LBL)

```
HCONS: < h0 qeq h14 h5 qeq h7 h11 qeq h13 > ]
```

- (4) Change sentential semantic index to it-cleft EP’s ARG0

```
INDEX: e15
[ e SF: prop TENSE: pres ... ]
```

- (5) Final result

²<http://sweaglesw.org/linguistics/ace/>

```

[ TOP: h0
INDEX: e15
[ e SF: prop TENSE: pres MOOD: indicative PROG: - PERF: - ]
RELS: < [ proper_q LBL: h4 ... ]
[ named LBL: h7 ARG0: x3 CARG: "Alice" ]
[ _see_v_1 LBL: h1 ARG0: e2 ARG1: x3 ARG2: x9 ]
[ proper_q LBL: h10 ARG0: x9 ... ]
[ named LBL: h13 ARG0: x9 CARG: "Bob" ]
[ _be_v_itcleft LBL: h14
ARG0: e15 ARG1: x3 ARG2: h1 ] >
HCONS: < h0 qeq h14 h5 qeq h7 h11 qeq h13 > ]

```

For each parse result, LIT generates one transformation for each linguistic phenomenon, obtaining a set of simple transformations. Each transformation result in this set can also be fed into LIT as a new base for transformation, allowing different rules to be stacked and producing compositions of transformations. LIT uses all transformation results to generate surface sentences.

3.3 Surface Sentence Selection

Selection by ERG: One of the advantages of the LIT system is that the grammar backbone ensures the acceptability of the generated data. The ACE parser only generates grammatical sentences, according to the ERG. Consequently, ill-formed LIT-transformed results are automatically rejected at the generation phase without additional efforts from the users and developers; for instance, even though LIT may produce a representation that would correspond to **Alice may will see Bob*.³, such a surface string will not be generated since the ERG does not accept it.

In practice, ERG slightly overgenerates and allows certain ungrammatical strings. Such cases are likely too rare to affect the overall quality of the dataset and can often be filtered out during post-selection. ERG also cannot rule out grammatically well-formed but semantically unnatural sentences, which limits the data quality for certain constructions, especially for passives. As a sanity check, we had expert annotators evaluate the generated data and found high agreement on the grammaticality of generated data; the full details are in Appendix B.

Post-Selection by Pretrained Language Models: ERG often permits multiple strings for a single representation since the meaning-to-form mapping is not unique in natural languages. To select the candidate sentence for a specific transformation, LIT employs GPT-2 (Radford et al., 2018) to rank multiple surface sentences generated from the same representation and selects the best one according to their perplexity scores.

³* means ungrammatical

3.4 Phenomena Covered

LIT is capable of perturbing sentences for seven linguistic transformations: polar questions, it-clefts, tense and aspect, modality, negation, passives and subject-object swapping. Examples for each transformation are shown in Appendix A. LIT also allows different transformations to be stacked where possible. LIT can be further extended for more linguistic transformations, and any extension to the LIT system would also receive all of the aforementioned benefits from ACE and ERG.

3.5 Comparison with Other Approaches

Flexibility: LIT covers certain simple constructions that can be handled with a template-based approach, for instance, the subject-object swapping in McCoy et al. (2019). LIT is, however, not limited to template-generated examples and is capable of perturbing naturally-occurring instances.

Plausibility: One special property setting LIT apart from other automatic dataset-construction methods is that LIT uses existing linguistic theories resources as its backbone. The use of ERG enables LIT to control data plausibility without human annotation from scratch.

Modularity: LIT consists of multiple modules: parsing and generation, transformation, and post-selection. Extending with more transformation rules, updating ERG (which is still under active development), and including other language models for post-selection can all be handled in the system without major modification to other modules, allowing LIT to be reused for different works.

Model Agnostic: LIT employs traditional linguistic methods for transforming sentences, and the role of language models is limited to selecting the best one from the strings generated by ERG. Contrasting to models trained on specific datasets, the ERG grammar behind LIT does not introduce bias from any specific architecture or dataset. This increases the utility of contrast sets generated with LIT as they are likely to be used for testing data-driven models.

3.6 Sentence Coverage

LIT successfully transformed 21.0% of the sentence pairs in MNLI and 19.7% in SNLI, with at least one transformed result for each sentence in the pair. The number of transformed sentence pairs by phenomenon is shown in Table 2.

Transformation	Label	Sentence 1	Sentence 2
o;o	Contradiction	Alice is driving a car.	Alice is playing piano.
i;i	Unchanged	It is Alice who is driving a car.	It is Alice who is playing piano.
pa;pa	Unchanged	A car is being driven by Alice.	Piano is being played by Alice.
f;p	Neutral	Alice will be driving a car.	Alice was playing piano.
m;o	Neutral	Alice may be driving a car.	Alice is playing piano.
f;p +i	Neutral	It is Alice who will be driving a car.	It is Alice who was playing piano.
f;p +pa	Neutral	A car will be driven by Alice.	Piano will be played by Alice.

Table 1: Examples for label rules used for determining labels of generated data for different transformations. We use **a;b + c** to denote compositional transformation where the premise is transformed with both **a** and **b**, whereas the hypothesis is transformed with **b** and **c**.

	# MNLI ex.			# SNLI ex.	
	train	m.	mm.	train	dev
o;o	392k	10k	10k	550k	10k
i;i	13k	1k	1k	65k	1k
pa;pa	3k	236	353	16k	586
f;p	3k	221	208	6k	111
p;f	3k	262	260	7k	142
m;o	13k	1k	1k	48k	905
p;f +i	4k	288	303	7k	122
p;f +pa	719	61	82	1k	45
f;p +i	4k	259	270	6k	91
f;p +pa	727	59	72	1k	37

Table 2: Number of examples (ex.) of different transformation rules in MNLI and SNLI parse results. For MNLI, we report example count in training (**train**) set and development (**dev**) set. For SNLI, we report example count in training (**train**) set, matched development (**m.**) set, and mismatched development (**mm.**) set.

4 Experiments

Using LIT, we evaluate whether large pretrained models ‘understand’ certain linguistic phenomena through testing them on transformed SNLI and MNLI instances. Specifically, we investigate whether BERT and RoBERTa can successfully predict transformed instances on modality (may), tenses (past; future), passivization, it cleft, and their compositions correctly and consistently. In the following section, we first discuss how we set up our tasks, and then we present our results on simple transformations and composed transformations, respectively.

4.1 Setup

For the purpose of this paper, we formulate our experiment settings as follows. Specifically, each

instance in SNLI/MNLI consists of a hypothesis (e.g., *Some men are playing a sport.*), a premise (e.g. *A soccer game with multiple males playing.*) and their corresponding relationship label (entailment). A dual transformed instance is obtained by applying LIT to either the hypothesis or premise, which may or may not change the label of their relationship (i.e., entailment, neutral, and contradiction).

4.1.1 Transforming NLI Datasets with LIT

While LIT does not produce labels after transformation, we apply two label-changing, two label-preserving, and the relevant compositional transformations listed in Table 1, with one example per transformation. Note that **o;o** means we do not modify the instance.

- **Modality** is used to talk about possibilities and necessities beyond what is actually true and is central to natural language semantics (Kratzer, 1991). We investigate models’ ability to understand the uncertainty expressed in the text by adding ‘may’ to the instance. Thus, a ‘contradiction’ or ‘entailment’ relationship label is changed to ‘neutral’ logically. Specifically, we consider adding ‘may’ to the premise (**m;o**). Note that one can also add ‘may’ to the hypothesis, which we leave for future work.
- **Tenses** are used to evaluate sentences at times other than the time of utterance. To probe whether models are able to perform temporal reasoning, we transformed the instances by assigning past tense to hypothesis and future tense to premise (**p;f**) or vice versa (**f;p**), which changes the ‘contradiction’ and ‘entailment’ label to ‘neutral’.

		f;p	p;f	i;i	pa;pa	m;o	p;f +i	p;f +pa	f;p +i	f;p +pa
ORI	Acc@Ori	93.21	91.60	91.86	95.28	90.90	90.97	93.44	94.21	94.92
	Acc@Ctr	5.43	41.98	85.17	90.99	15.13	34.38	32.79	6.18	6.78
	Consistency	4.98	34.35	91.04	93.99	10.19	28.82	29.51	5.79	5.08
		f;p	p;f	i;i	pa;pa	m;o	p;f +i	p;f +pa	f;p +i	f;p +pa
AUG	Acc@Ori	93.67	93.13	91.86	94.85	92.19	90.97	93.44	94.98	94.92
	Acc@Ctr	99.10	99.62	89.80	91.85	99.11	87.50	98.36	76.06	94.92
	Consistency	92.76	92.75	95.06	94.42	91.49	78.47	91.80	71.04	89.83

Table 3: Consistency and accuracies of `roberta-large` over different linguistic phenomena in MNLI. We first train two model separately on the original (ORI) training set and augmented (AUG) training set. Then, we evaluate the trained models on **m.** and **mm.** for each phenomena. In this table, we report accuracy on the original sentence pair (Acc@Ori), accuracy on the transformed sentence pair (Acc@Ctr), and the model’s consistency. Each accuracy/consistency has the format (**m./mm.**).

- **Label-preserving Transformations** do not require inferring the label after transformation, which serves to test models’ ability to stay consistent with its prediction after some linguistic perturbations. Here, we experiment on passivization (**pa**) and it-cleft (**i**).
- **Compositional Transformations** help us further evaluate models’ ‘understanding’ of certain linguistic phenomenon. If the models robustly ‘understand’ phenomenon α and β , composing both should not pose problems to the models. Specifically, we consider adding passivization and it cleft to **p;f** and **f;p** transformations. They are denoted as **p;f +i**, **p;f +p**, **f;p +i**, and **f;p +p** respectively.

The statistics for our generated dataset are shown in Table 2. We train two models on two training set. The original (ORI) training set includes untransformed SNLI training data, whilst the augmented (AUG) training set includes LIT-transformed data with all non-compositional transformations listed in Table 2. We test both models’ accuracy and consistency for all transformations in the same table.

We use a set of rules to infer the labels of generated pairs (see Table 1) based on the types of transformation and the original labels. For instance, originally entailment pairs will turn neutral when ‘may’ is inserted since the ‘may’ modality discharges the truth value of original propositions. ‘Decoupling’ the tenses of originally present-tense pairs for past/future tense pairs also turns the label to neutral, for events at different times are less likely to affect each other.

We hypothesize that NLI tasks follow logic rules completely and our following experiments also con-

form to that hypothesis, which legitimize our label-inferring rules. However exceptions to such rules may occur: *Alice died* nevertheless contradicts *Alice will be eating*, since dying is an event preventing future action of its agent. Annotation by three experts of 100 randomly chosen transformed pairs shows that 79% human agreement with the inferred label, with 92% for label-preserving transformations and 76% for label-changing transformations. Future work will explore refinements of our label-assignment procedure.

4.1.2 Probing Models

For pretrained language models, we use models from HuggingFace (Wolf et al., 2019). In this paper, we use `bert-base-uncased`, `bert-large-uncased` (Devlin et al., 2019), `roberta-base`, and `roberta-large` (Liu et al., 2019). For all models, we use Adam to optimize the parameters with an initial learning rate of 5×10^{-5} . For all the fine-tuning, we use the same seed and train with batch size 32 for 3 epochs, the same setting used in (Devlin et al., 2019). In this paper, since we never use the development set for early stopping or hyper-parameter tuning (and since MNLI doesn’t have a publicly available test set), we evaluate our models on the development set. Note that MNLI has matched (**m.**) and mismatched (**mm.**) test examples, which are derived from the same and different sources as those in the training set, respectively.

4.1.3 Evaluation Metrics

To fully evaluate models’ performance, we use both accuracy and consistency. While accuracy measures how well a model can accurately predict

		MNLI	aug-MNLI	SNLI	aug-SNLI
ORI	bert-base-uncased	84.31/84.79	69.47/69.05	90.97	46.96
	bert-large-uncased	86.54/86.46	71.28/70.34	91.78	47.72
	roberta-base	88.00/87.60	71.95/70.58	91.86	47.72
	roberta-large	90.01/90.34	73.78/73.04	92.83	46.34
AUG	bert-base-uncased	84.62/84.45	86.60/85.73	90.86	94.34
	bert-large-uncased	86.24/86.37	88.14/87.98	91.49	96.00
	roberta-base	87.51/87.52	89.66/89.45	92.13	95.05
	roberta-large	90.14/89.84	91.47/91.04	92.53	95.93

Table 4: Accuracy on MNLI and SNLI datasets. MNLI results have the format (m./mm.). SNLI results are on SNLI dev.

test instances, consistency measures how robust a model under certain perturbations. We report accuracy on the original test set (Acc@Ori), accuracy on the generated contrast set (Acc@Ctr), and the consistency score (defined below). Note that test sets for different phenomena might be different since we only choose the test instances to be included for each phenomenon if LIT produces contrast instances corresponding to the phenomenon.

Consistency In addition to using accuracy to measure models’ performance, recent research pays attention to consistency, which provides another perspective to probe models’ competence in the real world (Trichelair et al., 2018; Zhou et al., 2019; Gardner et al., 2020). If a model is robust for the given task, then its performance on original and transformed data should be consistent. For instance, a human is expected to be consistent over the understanding of both a simple sentence and its it-cleft counterpart. We thus measure consistency by comparing the model’s prediction on original and transformed data. We define consistency for a dual test instance as the match between labels assigned on original and transformed data instances. Specifically, we define the model to be consistent if a model makes the same label prediction (whether correct or not) for a dual test instance as for the original, and inconsistent otherwise.⁴ We evaluate the model consistency for each type of linguistic transformation to investigate the models’ robustness to different linguistic phenomena, and to examine the differences between the difficulties of different linguistic structures for the models.

⁴Note that this contrasts with what Gardner et al. (2020) call *contrast consistency*, where both predictions additionally have to be both correct.

4.2 Simple Transformations

By perturbing the test instances with our predefined transformations, we aim to probe pre-trained language models’ relevant linguistic knowledge and robustness towards those transformations.

As shown in Table 3, RoBERTa, trained on ORI of MNLI, performs worse on contrast sets, especially for label-changing transformations. Label-preserving transformations do not hurt models’ performance as much as label-changing transformations. We observed similar trends for other models (see Appendix C. This observation is aligned with (McCoy et al., 2019)), which suggests that models are relying on lexical overlaps to infer the relationship between premise and hypothesis.

Another observation is that RoBERTa does not achieve high consistency in any of the simple transformations. The poor and inconsistent performance of RoBERTa on our contrast sets shows that even though the model can perform very well on the in-distribution test set, there is still a systematic gap for future models to overcome.

4.3 Applying LIT for Data Augmentation

Having shown that pre-trained language models do not generalize well to our generated contrast sets, we ask whether we can ‘teach’ models to recognize those phenomena and make correct predictions accordingly.

We do this by fine-tuning models on the augmented training data together with the original data. As shown in Table 4, we observe that, when training on the augmented data, models preserve their performance on the original test set while improving significantly on the out-of-distribution test sets.

Taking a closer look over the specific phenomenon in Table 3, models’ performance increases significantly on label-changing contrast

sets. This indicates that models improve in terms of ‘understanding’ the role of modality (may) and tenses in natural language inference. Arguably, models may simply memorize the ‘trick’ that modality (may) and tenses (past to future) are associated with label ‘neutral.’ However, we successfully show that we could enable models to learn those ‘tricks’ through data augmentation. Future work will probe whether models fine-tuned on our augmented data are relying on such heuristics.

The models’ performance also increases slightly for label-preserving transformations. However, their consistency does not increase for every transformation, which suggests that data augmentation alone may not suffice for building robust models.

4.4 Compositional Transformations

We further investigate the models’ performance when multiple transformation rules are composed together and applied to a single sentence. We probe models fine-tuned on the original dataset and the dataset augmented with only simple transformations with our compositional test sets. If a model learns the linguistic phenomenon systematically, it should perform well on these compositional transformations even without training. This resembles the zero-shot tests on tasks like SCAN (Lake and Baroni, 2018), but applied to naturally occurring linguistic data.⁵

The bottom-right quadrant of Table 3 shows that RoBERTa performs very well on compositional transformations when it is fine-tuned only on simple transformations, in some cases ($\mathbf{p}; \mathbf{f} + \mathbf{pa}$) even performing better than on the simple transformation data. Again, we observed similar results across all models (see Appendix C). This suggests that it has learned something systematic about the transformations in the augmented dataset.

For both $\mathbf{p}; \mathbf{f}$ and $\mathbf{f}; \mathbf{p}$, RoBERTa performs worse when additionally composing with it-clefts than with passivization. This suggests that there are differences in the level of systematicity learned for the different transformations, a phenomenon which future work will investigate in more detail.

5 Discussion and Analysis

With LIT, we reveal that current high-performance NLI models still suffer from understanding simple linguistic phenomena. They can be trained to un-

⁵See Andreas (2020) for a complementary, heuristic-driven approach to compositional data augmentation.

derstand these phenomena in a way that appears systematic. In the remainder, we discuss the limitations of LIT, applying LIT to investigate the systematic deficiency of current large-scale datasets, and potential applications of LIT to tasks other than NLI.

5.1 Limitations of LIT

One major limitation of LIT is the dependency on ERG, which took more than twenty years of human labor and is specifically for English. It is possible to swap ERG/ACE parser with data-driven parsers and generators trained on semantic graphbanks, including the DeepBank (Flickinger et al., 2012) which uses the same representation frameworks, potentially extending the method to other languages where a broad-coverage hand-crafted grammar is unavailable. Using data-driven models, however, does re-introduce possible model bias and uncertainty of robustness. Nevertheless, once such a resource is available, LIT provides a method of transforming sentences for data augmentation and integrating linguistic knowledge into a data-driven NLP pipeline.

Future work will also involve expanding the phenomena covered by LIT by generating new transformation rules (cf. 3.4). One potential extension is the insertion of control and raising verbs:

- (6) Alice voted for Bob.
 - a. Alice seemed to have voted for Bob.
 - b. Alice wished to vote for Bob.
 - c. Alice persuaded Carol to vote for Bob.

LIT also has a limited coverage, successfully transforming about 20% of the instances in SNLI (see Section 3.4). The limited coverage may introduce bias in the generated dataset; for instance, the ERG grammar is more likely to fail when parsing complicated sentences. Nevertheless, we provide a proof of concept that the method can be used to augment data and probe for understanding of the linguistic phenomena of interest here; a higher recall grammar will only improve the situation, and can be easily integrated.

5.2 Analysing Sentence Types in Datasets

In addition to constructing contrast sets, we also used LIT to directly analyze the sentence types in the transformable portion of SNLI and MNLI to investigate the effects of data bias on pretrained models probed in our work. For MNLI, we found

that 46.4% sentences are in present tense, 32.2% in past tense and only 2.95% in future tense; 7.27% sentences are passive, 0.580% have *may* modality and 0.227% are it-cleft sentences. We found no passive/future or future/passive tense pairs. The lack of sentences with *may* modality and mismatched tense pairs may account for the low performance for those transformations before fine-tuning on them. It-cleft transformation does not change the meaning and labels, which may explain the high performance despite its rarity in the original data. Note that LIT can only detect linguistic phenomena in sentences parsable with ERG (see Section 3.6), but such functionality can still provide important insights on datasets and can be further explored in future works.

6 Conclusion

We propose Linguistically-Informed Transformations (LIT), a general method to generate contrast sets using an existing linguistic resource. We apply LIT to transform NLI datasets and evaluate current state-of-the-art NLI models. We reveal the systematic gap between current NLI models and an ideal NLI model for NLP practice, which comes from the inadequate coverage of the linguistic phenomenon of SNLI and MNLI. We further show that models can be further improved by using LIT to augment the training data. Furthermore, models fine-tuned on simple transformations perform very well on compositional transformations, suggesting that fine-tuning provides some systematic understanding of these phenomena.

References

- Jacob Andreas. 2020. [Good-enough compositional data augmentation](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7556–7566, Online. Association for Computational Linguistics.
- 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.
- Ronan Le Bras, Swabha Swayamdipta, Chandra Bhagavatula, Rowan Zellers, Matthew E. Peters, Ashish Sabharwal, and Yejin Choi. 2020. [Adversarial filters of dataset biases](#).
- Ann Copestake and Dan Flickinger. 2000. [An open source grammar development environment and broad-coverage English grammar using HPSG](#). In *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC'00)*, Athens, Greece. European Language Resources Association (ELRA).
- Ann Copestake, Dan Flickinger, Carl Pollard, and Ivan A. Sag. 2005. [Minimal Recursion Semantics: An Introduction](#). *Research on Language and Computation*, 3(2-3):281–332.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.
- Emily Dinan, Samuel Humeau, Bharath Chintagunta, and Jason Weston. 2019. [Build it break it fix it for dialogue safety: Robustness from adversarial human attack](#). *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*.
- Dan Flickinger, Yi Zhang, and Valia Kordoni. 2012. Deepbank. a dynamically annotated treebank of the wall street journal. In *Proceedings of the 11th International Workshop on Treebanks and Linguistic Theories*, pages 85–96.
- Matt Gardner, Yoav Artzi, Victoria Basmova, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, Nitish Gupta, Hanna Hajishirzi, Gabriel Ilharco, Daniel Khashabi, Kevin Lin, Jiangming Liu, Nelson F. Liu, Phoebe Mulcaire, Qiang Ning, Sameer Singh, Noah A. Smith, Sanjay Subramanian, Reut Tsarfaty, Eric Wallace, Ally Quan Zhang, and Ben Zhou. 2020. Evaluating nlp models via contrast sets. *ArXiv*, abs/2004.02709.
- Mor Geva, Yoav Goldberg, and Jonathan Berant. 2019. [Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1161–1166, Hong Kong, China. 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, New Orleans, Louisiana. Association for Computational Linguistics.
- Jeremy Howard and Sebastian Ruder. 2018. [Universal language model fine-tuning for text classification](#). In

- Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.
- Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2019a. [Is bert really robust? a strong baseline for natural language attack on text classification and entailment](#).
- Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2019b. [Is bert really robust? natural language attack on text classification and entailment](#). *ArXiv*, abs/1907.11932.
- Divyansh Kaushik, Eduard Hovy, and Zachary Lipton. 2020. [Learning the difference that makes a difference with counterfactually-augmented data](#). In *International Conference on Learning Representations*.
- Angelika Kratzer. 1991. Modality. in semantics: An international handbook of contemporary research.
- Brenden M. Lake and Marco Baroni. 2018. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *ICML*.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692.
- Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. [Building a large annotated corpus of English: The Penn Treebank](#). *Computational Linguistics*, 19(2):313–330.
- Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. [Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.
- Junghyun Min, R. Thomas McCoy, Dipanjan Das, Emily Pitler, and Tal Linzen. 2020. [Syntactic data augmentation increases robustness to inference heuristics](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Seattle, Washington. Association for Computational Linguistics.
- Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. [Deep contextualized word representations](#). In *Proc. of NAACL*.
- Carl Jesse Pollard and Ivan A. Sag. 1994. *Head-driven phrase structure grammar*. Studies in contemporary linguistics. Center for the Study of Language and Information ; University of Chicago Press, Stanford : Chicago.
- Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. [Improving language understanding by generative pre-training](#).
- Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. [Language models are unsupervised multitask learners](#).
- Ivan A Sag, Thomas Wasow, and Emily M Bender. 2003. *Syntactic Theory: A Formal Introduction*, volume 152 of *CSLI Lecture Notes*. CSLI Publications.
- Paul Trichelair, Ali Emami, Adam Trischler, Kaheer Suleman, and Jackie Chi Kit Cheung. 2018. [How reasonable are common-sense reasoning tasks: A case-study on the winograd schema challenge and swag](#).
- Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. [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.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R’emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. [Huggingface’s transformers: State-of-the-art natural language processing](#). *ArXiv*, abs/1910.03771.
- Xuhui Zhou, Yue Zhang, Leyang Cui, and Dandan Huang. 2019. [Evaluating commonsense in pre-trained language models](#).