Can neural networks understand monotonicity reasoning?

Hitomi Yanaka[1,2]**, Koji Mineshima**[2]**, Daisuke Bekki**[2]**,**
Kentaro Inui[1,3]**, Satoshi Sekine**[1]**, Lasha Abzianidze**[4]**, and Johan Bos**[4]
[1]RIKEN, [2]Ochanomizu University, [3]Tohoku University, Japan
[4]University of Groningen, Netherlands
`{hitomi.yanaka, satoshi.sekine}@riken.jp,`
`mineshima.koji@ocha.ac.jp,bekki@is.ocha.ac.jp,`
`inui@ecei.tohoku.ac.jp,{l.abzianidze, johan.bos}@rug.nl`

Abstract

Monotonicity reasoning is one of the important reasoning skills for any intelligent natural language inference (NLI) model in that it requires the ability to capture the interaction between lexical and syntactic structures. Since no test set has been developed for monotonicity reasoning with wide coverage, it is still unclear whether neural models can perform monotonicity reasoning in a proper way. To investigate this issue, we introduce the Monotonicity Entailment Dataset (MED). Performance by state-of-the-art NLI models on the new test set is substantially worse, under 55%, especially on downward reasoning. In addition, analysis using a monotonicity-driven data augmentation method showed that these models might be limited in their generalization ability in upward and downward reasoning.

1 Introduction

Natural language inference (NLI), also known as recognizing textual entailment (RTE), has been proposed as a benchmark task for natural language understanding. Given a premise P and a hypothesis H, the task is to determine whether the premise semantically entails the hypothesis (Dagan et al., 2013). A number of recent works attempt to test and analyze what type of inferences an NLI model may be performing, focusing on various types of lexical inferences (Glockner et al., 2018; Naik et al., 2018; Poliak et al., 2018) and logical inferences (Bowman et al., 2015b; Evans et al., 2018).

Concerning logical inferences, monotonicity reasoning (van Benthem, 1983; Icard and Moss, 2014), which is a type of reasoning based on word replacement, requires the ability to capture the interaction between lexical and syntactic structures. Consider examples in (1) and (2).

(1) a. **All** [_workers ↓_] [_joined for a French dinner ↑_]

b. _All workers joined for a dinner_

c. _All new workers joined for a French dinner_

(2) a. **Not all** [_new workers ↑_] _joined for a dinner_

b. _Not all workers joined for a dinner_

A context is **upward entailing** (shown by [... ↑]) that allows an inference from (1a) to (1b), where _French dinner_ is replaced by a more general concept _dinner_. On the other hand, a **downward entailing** context (shown by [... ↓]) allows an inference from (1a) to (1c), where _workers_ is replaced by a more specific concept _new workers_. Interestingly, the direction of monotonicity can be reversed again by embedding yet another downward entailing context (e.g., _not_ in (2)), as witness the fact that (2a) entails (2b). To properly handle both directions of monotonicity, NLI models must detect monotonicity operators (e.g., _all_, _not_) and their arguments from the syntactic structure.

For previous datasets containing monotonicity inference problems, FraCaS (Cooper et al., 1994) and the GLUE diagnostic dataset (Wang et al., 2019) are manually-curated datasets for testing a wide range of linguistic phenomena. However, monotonicity problems are limited to very small sizes (FraCaS: 37/346 examples and GLUE: 93/1650 examples). The limited syntactic patterns and vocabularies in previous test sets are obstacles in accurately evaluating NLI models on monotonicity reasoning.

To tackle this issue, we present a new evaluation dataset[1] that covers a wide range of monotonicity reasoning that was created by crowdsourcing and collected from linguistics publications (Section 3). Compared with manual or automatic construction, we can collect naturally-occurring examples by crowdsourcing and well-designed ones from linguistics publications. To enable the evaluation of

---

[1]The dataset will be made publicly available at https://github.com/verypluming/MED.

skills required for monotonicity reasoning, we annotate each example in our dataset with linguistic tags associated with monotonicity reasoning.

We measure the performance of state-of-the-art NLI models on monotonicity reasoning and investigate their generalization ability in upward and downward reasoning (Section 4). The results show that all models trained with SNLI (Bowman et al., 2015b) and MultiNLI (Williams et al., 2018) perform worse on downward inferences than on upward inferences.

In addition, we analyzed the performance of models trained with an automatically created monotonicity dataset, HELP (Yanaka et al., 2019). The analysis with monotonicity data augmentation shows that models tend to perform better in the same direction of monotonicity with the training set, while they perform worse in the opposite direction. This indicates that the accuracy on monotonicity reasoning depends solely on the majority direction in the training set, and models might lack the ability to capture the structural relations between monotonicity operators and their arguments.

2 Monotonicity

As an example of a monotonicity inference, consider the example with the determiner *every* in (3); here the premise P entails the hypothesis H.

(3) P: *Every* [$_{NP}$ *person* ↓] [$_{VP}$ *bought a movie ticket* ↑]
H: *Every young person bought a ticket*

Every is downward entailing in the first argument (NP) and upward entailing in the second argument (VP), and thus the term *person* can be more specific by adding modifiers (*person* ⊒ *young person*), replacing it with its hyponym (*person* ⊒ *spectator*), or adding conjunction (*person* ⊒ *person and alien*). On the other hand, the term *buy a ticket* can be more general by removing modifiers (*bought a movie ticket* ⊑ *bought a ticket*), replacing it with its hypernym (*bought a movie ticket* ⊑ *bought a show ticket*), or adding disjunction (*bought a movie ticket* ⊑ *bought or sold a movie ticket*). Table 1 shows determiners modeled as binary operators and their polarities with respect to the first and second arguments.

There are various types of downward operators, not limited to determiners (see Table 2). As shown in (4), if a propositional object is embedded in a downward monotonic context (e.g., *when*), the polarity of words over its scope can be reversed.

Determiners	First argument	Second argument
every, each, all	downward	upward
some, a, a few, many, several, proper noun	upward	upward
any, no, few, at most X, fewer than X, less than X	downward	downward
the, both, most, this, that	non-monotone	upward
exactly	non-monotone	non-monotone

Table 1: Determiners and their polarities.

Category	Examples
determiners	*every, all, any, few, no*
negation	*not, n't, never*
verbs	*deny, prohibit, avoid*
nouns	*absence of, lack of, prohibition*
adverbs	*scarcely, hardly, rarely, seldom*
prepositions	*without, except, but*
conditionals	*if, when, in case that, provided that, unless*

Table 2: Examples of downward operators.

(4) P: *When* [*every* [$_{NP}$ *young person* ↑] [$_{VP}$ *bought a ticket* ↓]], [*that shop was open*]
H: *When* [*every* [$_{NP}$ *person*] [$_{VP}$ *bought a movie ticket*]], [*that shop was open*]

Thus, the polarity (↑ and ↓), where the replacement with more general (specific) phrases licenses entailment, needs to be determined by the interaction of monotonicity properties and syntactic structures; polarity of each constituent is calculated based on a monotonicity operator of functional expressions (e.g., *every*, *when*) and their function-term relations.

3 Dataset

3.1 Human-oriented dataset

To create monotonicity inference problems, we should satisfy three requirements: (a) detect the monotonicity operators and their arguments; (b) based on the syntactic structure, induce the polarity of the argument positions; and (c) replace the phrase in the argument position with a more general or specific phrase in natural and various ways (e.g., by using lexical knowledge or logical connectives). For (a) and (b), we first conduct polarity computation on a syntactic structure for each sentence, and then select premises involving upward/downward expressions.

For (c), we use crowdsourcing to narrow or broaden the arguments. The motivation for using crowdsourcing is to collect naturally alike monotonicity inference problems that include various expressions. One problem here is that it is un-

Figure 1: Overview of our human-oriented dataset creation. E: entailment, NE: non-entailment.

clear how to instruct workers to create monotonicity inference problems without knowledge of natural language syntax and semantics. We must make tasks simple for workers to comprehend and provide sound judgements. Moreover, recent studies (Gururangan et al., 2018; Poliak et al., 2018; Tsuchiya, 2018) point out that previous crowdsourced datasets, such as SNLI (Bowman et al., 2015a) and MultiNLI (Williams et al., 2018), include hidden biases. As these previous datasets are motivated by approximated entailments, workers are asked to freely write hypotheses given a premise, which does not strictly restrict them to creating logically complex inferences.

Taking these concerns into consideration, we designed two-step tasks to be performed via crowdsourcing for creating a monotonicity test set; (i) a hypothesis creation task and (ii) a validation task. The task (i) is to create a hypothesis by making some polarized part of an original sentence more specific. Instead of writing a complete sentence from scratch, workers are asked to rewrite only a relatively short sentence. By restricting workers to rewrite only a polarized part, we can effectively collect monotonicity inference examples. The task (ii) is to annotate an entailment label for the premise-hypothesis pair generated in (i). Figure 1 summarizes the overview of our human-oriented dataset creation. We used the crowdsourcing platform Figure Eight for both tasks.

3.1.1 Premise collection

As a resource, we use declarative sentences with more than five tokens from the Parallel Meaning Bank (PMB) (Abzianidze et al., 2017). The PMB contains syntactically correct sentences an-

notated with its syntactic category in Combinatory Categorial Grammar (CCG; Steedman, 2000) format, which is suitable for our purpose. To get a whole CCG derivation tree, we parse each sentence by the state-of-the-art CCG parser, depccg (Yoshikawa et al., 2017). Then, we add a polarity to every constituent of the CCG tree by the polarity computation system ccg2mono (Hu and Moss, 2018) and make the polarized part a blank field.

We ran a trial rephrasing task on 500 examples and detected 17 expressions that were too general and thus difficult to rephrase them in a natural way (e.g., *every one*, *no time*). We removed examples involving such expressions. To collect more downward inference examples, we select examples involving determiners in Table 1 and downward operators in Table 2. As a result, we selected 1,485 examples involving expressions having arguments with upward monotonicity and 1,982 examples involving expressions having arguments with downward monotonicity.

3.1.2 Hypothesis creation

We present crowdworkers with a sentence whose polarized part is underlined, and ask them to replace the underlined part with more specific phrases in three different ways. In the instructions, we showed examples rephrased in various ways: by adding modifiers, by adding conjunction phrases, and by replacing a word with its hyponyms.

Workers were paid US$0.05 for each set of substitutions, and each set was assigned to three workers. To remove low-quality examples, we set the minimum time it should take to complete each set to 200 seconds. The entry in our task was restricted to workers from native speaking English countries. 128 workers contributed to the task, and we created 15,339 hypotheses (7,179 upward examples and 8,160 downward examples).

3.1.3 Validation

The gold label of each premise-hypothesis pair created in the previous task is automatically determined by monotonicity calculus. That is, a downward inference pair is labeled as *entailment*, while an upward inference pair is labeled as *non-entailment*.

However, workers sometimes provided some ungrammatical or unnatural sentences such as the case where a rephrased phrase does not satisfy the

selectional restrictions (e.g., original: *Tom doesn't live in Boston*, rephrased: *Tom doesn't live in yes*), making it difficult to judge their entailment relations. Thus, we performed an annotation task to ensure accurate labeling of gold labels. We asked workers about the entailment relation of each premise-hypothesis pair as well as how natural it is.

Worker comprehension of an entailment relation directly affects the quality of inference problems. To avoid worker misunderstandings, we showed workers the following definitions of labels and five examples for each label:

1. *entailment*: the case where the hypothesis is true under any situation that the premise describes.

2. *non-entailment*: the case where the hypothesis is not always true under a situation that the premise describes.

3. *unnatural*: the case where either the premise and/or the hypothesis is ungrammatical or does not make sense.

Workers were paid US$0.04 for each question, and each question was assigned to three workers. To collect high-quality annotation results, we imposed ten test questions on each worker, and removed workers who gave more than three wrong answers. We also set the minimum time it should take to complete each question to 200 seconds. 1,237 workers contributed to this task, and we annotated gold labels of 15,339 premise-hypothesis pairs.

Table 3 shows the numbers of cases where answers matched gold labels automatically determined by monotonicity calculus. This table shows that there exist inference pairs whose labels are difficult even for humans to determine; there are 3,354 premise-hypothesis pairs whose gold labels as annotated by polarity computations match with those answered by all workers. We selected these naturalistic monotonicity inference pairs for the candidates of the final test set.

To make the distribution of gold labels symmetric, we checked these pairs to determine if we can swap the premise and the hypothesis, reverse their gold labels, and create another monotonicity inference pair. In some cases, shown below, the gold label cannot be reversed if we swap the premise and the hypothesis.

	Upward /cases(%)	Downward /cases(%)	Total /cases(%)
3 labels match	1,069 (7.0)	2,285 (14.9)	3,354 (21.9)
2 labels match	1,814 (11.8)	2,301 (15.0)	4,115 (26.8)
1 labels match	2,295 (15.0)	1,915 (12.5)	4,210 (27.5)
no match	1,998 (27.8)	1,652 (10.8)	3,650 (37.8)

Table 3: Numbers of cases where answers matched automatically determined gold labels.

(a) Replacement with synonyms In (5), *child* and *kid* are not hyponyms but synonyms, and the premise P and the hypothesis H are paraphrases.

(5) P: *Tom is no longer a child*
    H: *Tom is no longer a kid*

These cases are not strict downward inference problems, in the sense that a phrase is not replaced by its hyponym/hypernym.

(b) Non-constituents Consider the example (6).

(6) P: *The moon has no atmosphere*
    H: *The moon has no atmosphere, and the gravity force is too low*

The hypothesis H was created by asking workers to make *atmosphere* in the premise P more specific. However, the additional phrase *and the gravity force is too low* does not form constituents with *atmosphere*. Thus, such examples are not strict downward monotone inferences.

In such cases as (a) and (b), we do not swap the premise and the hypothesis. In the end, we collected 4,068 examples from crowdsourced datasets.

3.2 Linguistics-oriented dataset

We also collect monotonicity inference problems from previous manually curated datasets and linguistics publications. The motivation is that previous linguistics publications related to monotonicity reasoning are expected to contain well-designed inference problems, which might be challenging problems for NLI models.

We collected 1,184 examples from 11 linguistics publications (Barwise and Cooper, 1981; Hoeksema, 1986; Heim and Kratzer, 1998; Bonevac et al., 1999; Fyodorov et al., 2003; Geurts, 2003; Geurts and van der Slik, 2005; Zamansky et al., 2006; Szabolcsi et al., 2008; Winter, 2016; Denic et al., 2019). Regarding previous manually-curated datasets, we collected 93 examples for monotonicity reasoning from the GLUE diagnostic dataset, and 37 single-premise problems from FraCaS.

Genre	Tags	Premise	Hypothesis	Gold
Crowd	up	*There is a cat on the chair*	*There is a cat sleeping on the chair*	NE
	up: cond	*If you heard her speak English, you would take her for a native American*	*If you heard her speak English, you would take her for an American*	E
	up:rev conj	*Dogs and cats have all the good qualities of people without at the same time possessing their weaknesses*	*Dogs have all the good qualities of people without at the same time possessing their weaknesses*	E
	up:lex	*He approached the boy reading a magazine*	*He approached the boy reading a book*	E
	down:lex	*Tom hardly ever listens to music*	*Tom hardly ever listens to rock 'n' roll*	E
	down:conj	*You don't like love stories and sad endings*	*You don't like love stories*	NE
	down:cond	*If it is fine tomorrow, we'll go on a picnic*	*If it is fine tomorrow in the field, we'll go on a picnic*	E
	down	*I never had a girlfriend before*	*I never had a girlfriend taller than me before*	E
Paper	up:rev	*Every cook who is not a tall man ran*	*Every cook who is not a man ran*	E
	up:disj	*Every man sang*	*Every man sang or danced*	E
	up:lex rev	*None of the sopranos sang with fewer than three of the tenors*	*None of the sopranos sang with fewer than three of the male singers*	E
	non	*Exactly one man ran quickly*	*Exactly one man ran*	NE
	down	*At most three elephants are blue*	*At most three elephants are navy blue*	E

Table 4: Examples in the MED dataset. Crowd: problems collected through crowdsourcing, Paper: problems collected from linguistics publications, up: upward monotone, down: downward monotone, non: non-monotone, cond: condisionals, rev: reverse, conj: conjunction, disj: disjunction, lex: lexical knowledge, E: entailment, NE: non-entailment.

Type	Label	Crowd	Paper	Total
Upward (1,820)	Entailment	323	305	628
	Non-entailment	893	299	1,192
Downward (3,270)	Entailment	1,871	201	2,072
	Non-entailment	979	219	1,198
Non-monotone (292)	Entailment	0	15	15
	Non-entailment	2	275	277
Total		4,068	1,314	5,382

Table 5: Statistics for the MED dataset.

Both the GLUE diagnostic dataset and FraCaS categorize problems by their types of monotonicity reasoning, but we found that each dataset has different classification criteria.[2] Thus, following GLUE, we reclassified problems into three types of monotone reasoning (upward, downward, and non-monotone) by checking if they include (i) the target monotonicity operator in both the premise and the hypothesis and (ii) the phrase replacement in its argument position. In the GLUE diagnostic dataset, there are several problems whose gold labels are *contradiction*. We regard them as *non-entailment* in that the premise does not semantically entail the hypothesis.

3.3 Statistics

We merged the human-oriented dataset created via crowdsourcing and the linguistics-oriented dataset created from linguistics publications to create the current version of the monotonicity entailment dataset (MED). Table 4 shows some examples from the MED dataset. We can see that our dataset contains various phrase replacements (e.g., conjunction, relative clauses, and comparatives). Table 5 reports the statistics of the MED dataset, including 5,382 premise-hypothesis pairs (1,820 upward examples, 3,270 downward examples, and 292 non-monotone examples). Regarding non-monotone problems, gold labels are always *non-entailment*, whether a hypothesis is more specific or general than its premise, and thus almost all non-monotone problems are labeled as *non-entailment*.[3] The size of the word vocabulary in the MED dataset is 4,023, and overlap ratios of vocabulary with previous standard NLI datasets is 95% with MultiNLI and 90% with SNLI.

We assigned a set of annotation tags for linguistic phenomena to each example in the test set. These tags allow us to analyze how well models perform on each linguistic phenomenon related to monotonicity reasoning. We defined 6 tags (see Table 4 for examples):

1. *lexical knowledge* (2,073 examples): inference problems that require lexical relations (i.e., hypernyms, hyponyms, or synonyms)

2. *reverse* (240 examples): inference problems where a propositional object is embedded in a downward environment more than once

3. *conjunction* (283 examples): inference problems that include the phrase replacement by adding conjunction (*and*) to the hypothesis

---

[2]FraCaS categorizes each problem by whether its replacement broadens an argument (upward monotone) or narrows it (downward monotone).

[3]15 non-monotone problems which include the replacement with synonyms are labeled as *entailment*.

4. *disjunction* (254 examples): inference problems that include the phrase replacement by adding disjunction (*or*) to the hypothesis

5. *conditionals* (149 examples): inference problems that include conditionals (e.g., *if, when, unless*) in the hypothesis [4]

6. *negative polarity items* (NPIs) (338 examples): inference problems that include NPIs (e.g., *any, ever, at all, anything, anyone, anymore, anyhow, anywhere*) in the hypothesis

4 Results and Discussion

4.1 Baselines

To test the difficulty of our dataset, we checked the majority class label and the accuracies of five state-of-the-art NLI models adopting different approaches: BiMPM (Bilateral Multi-Perspective Matching Model; Wang et al., 2017), ESIM (Enhanced Sequential Inference Model; Chen et al., 2017), Decomposable Attention Model (Parikh et al., 2016), KIM (Knowledge-based Inference Model; Chen et al., 2018), and BERT (Bidirectional Encoder Representations from Transformers model; Devlin et al., 2019). Regarding BERT, we checked the performance of a model pretrained on Wikipedia and BookCorpus for language modeling and trained with SNLI and MultiNLI. For other models, we checked the performance trained with SNLI. In agreement with our dataset, we regarded the prediction label *contradiction* as *non-entailment*.

Table 6 shows that the accuracies of all models were better on upward inferences, in accordance with the reported results of the GLUE leaderboard. The overall accuracy of each model was low. In particular, all models underperformed the majority baseline on downward inferences, despite some models having rich lexical knowledge from a knowledge base (KIM) or pretraining (BERT). This indicates that downward inferences are difficult to perform even with the expansion of lexical knowledge. In addition, it is interesting to see that if a model performed better on upward inferences, it performed worse on downward inferences. We will investigate these results in detail below.

---

[4] *When*-clauses can have temporal and non-temporal interpretations (Moens and Steedman, 1988). We assign the conditional tag to those cases where *when* is interchangeable with *if*, thus excluding those cases where *when*-clauses have temporal episodic interpretation (e.g., *When she came back from the trip, she bought a gift*).

Model	Train	Upward	Downward	Non	All
Majority		65.5	63.3	99.3	50.4
BiMPM	SNLI	53.5	**57.6**	27.4	**54.6**
ESIM	SNLI	71.1	45.2	41.8	53.8
DeComp	SNLI	66.1	42.1	**64.4**	51.4
KIM	SNLI	78.8	30.3	53.1	48.0
BERT	SNLI	50.1	46.8	7.5	45.8
BERT	MNLI	**82.7**	22.8	52.7	44.7

Table 6: Accuracies (%) for different models and training datasets.

Training set	Upward	Downward	Non	All
MNLI	**82.7**	22.8	52.7	44.7
MNLI–Hyp	34.3	18.3	31.5	24.4
MNLI+HELP	76.0	**70.3**	**59.9**	**71.6**
MNLI+HELP–Hyp	61.3	30.5	34.9	41.1

Table 7: Evaluation results on types of monotonicity reasoning. –Hyp: Hypothesis-only model.

4.2 Data augmentation for analysis

To explore whether the performance of models on monotonicity reasoning depends on the training set or the model themselves, we conducted further analysis performed by data augmentation with the automatically generated monotonicity dataset HELP (Yanaka et al., 2019). HELP contains 36K monotonicity inference examples (7,784 upward examples, 21,192 downward examples, and 1,105 non-monotone examples). The size of the HELP word vocabulary is 15K, and the overlap ratio of vocabulary between HELP and MED is 15.2%.

We trained BERT on MultiNLI only and on MultiNLI augmented with HELP, and compared their performance. Following Poliak et al. (2018), we also checked the performance of a hypothesis-only model trained with each training set to test whether our test set contains undesired biases.

4.2.1 Effects of data augmentation

Table 7 shows that the performance of BERT with the hypothesis-only training set dropped around 10-40% as compared with the one with the premise-hypothesis training set, even if we use the data augmentation technique. This indicates that the MED test set does not allow models to predict from hypotheses alone. Data augmentation by HELP improved the overall accuracy to 71.6%, but there is still room for improvement. In addition, while adding HELP increased the accuracy on downward inferences, it slightly decreased accuracy on upward inferences. The size of down-
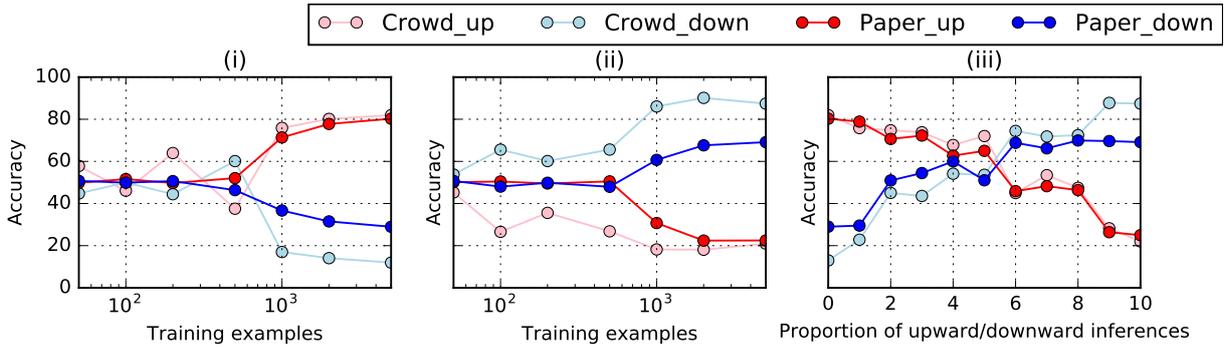
Figure 2: Accuracy throughout training BERT (i) with only upward examples and (ii) with only downward examples. We checked the accuracy at sizes [50, 100, 200, 500, 1000, 2000, 5000] for each direction. (iii) Performance on different ratios of upward/downward training sets. The total size of the training sets was 5,000 examples.

ward examples in HELP is much larger than that of upward examples. This might improve accuracy on downward inferences, but might decrease accuracy on upward inferences.

To investigate the relationship between accuracy on upward inferences and downward inferences, we checked the performance throughout training BERT with only upward and downward inference examples in HELP (Figure 2 (i), (ii)). These two figures show that, as the size of the upward training set increased, BERT performed better on upward inferences but worse on downward inferences, and vice versa.

Figure 2 (iii) shows performance on a different ratio of upward and downward inference training sets. When downward inference examples constitute more than half of the training set, accuracies on upward and downward inferences were reversed. As the ratio of downward inferences increased, BERT performed much worse on upward inferences. This indicates that a training set in one direction (upward or downward entailing) of monotonicity might be harmful to models when learning the opposite direction of monotonicity.

Previous work using HELP (Yanaka et al., 2019) reported that the BERT trained with MultiNLI and HELP containing both upward and downward inferences improved accuracy on both directions of monotonicity. MultiNLI rarely comes from downward inferences (see Section 4.3), and its size is large enough to be immune to the side-effects of downward inference examples in HELP. This indicates that MultiNLI might act as a *buffer* against side-effects of the monotonicity-driven data augmentation technique.

Genre		−HELP	+HELP	△
Crowd	Up	87.1	83.6	−3.5
	Down	21.2	70.3	+49.1
	Non	100.0	100.0	±0.0
	All	40.9	74.3	+33.4
Paper	Up	74.5	60.8	−13.7
	Down	33.8	69.5	+35.7
	Non	52.4	59.7	+7.3
	All	56.6	63.3	+6.7

Table 8: Evaluation results by genre. Paper: problems collected from linguistics publications, Crowd: problems via crowdsourcing.

4.2.2 Linguistics-oriented versus human-oriented

Table 8 shows the evaluation results by genre. This result shows that inference problems collected from linguistics publications are more challenging than crowdsourced inference problems, even if we add HELP to training sets. As shown in Figure 2, the change in performance on problems from linguistics publications is milder than that on problems from crowdsourcing. This result also indicates the difficulty of problems from linguistics publications. Regarding non-monotone problems collected via crowdsourcing, there are very few non-monotone problems, so accuracy is 100%. Adding non-monotone problems to our test set is left for future work.

4.2.3 Linguistic phenomena

Table 9 shows the evaluation results by type of linguistic phenomenon. While accuracy on problems involving NPIs and conditionals was improved on both upward and downward inferences, accuracy on problems involving conjunction and disjunction was improved on only one direction. In addition, it is interesting to see that the change in

Tag			−HELP	+HELP	△
	Lexical	(743)	81.0	70.8	−10.2
	non-Lexical	(1,077)	84.1	79.6	−4.5
	NPIs	(64)	20.3	35.9	+15.6
Up	Conditionals	(29)	51.7	62.1	+9.4
	Conjunction	(175)	94.3	88.0	−6.3
	Disjunction	(96)	4.2	32.3	+28.1
	Reverse	(240)	74.2	28.7	−45.5
	Lexical	(477)	46.1	64.6	+18.5
	non-Lexical	(2,793)	18.8	71.2	+52.4
	NPIs	(266)	44.0	60.2	+16.2
Down	Conditionals	(120)	15.8	20.0	+4.2
	Conjunction	(106)	24.5	40.6	+16.1
	Disjunction	(138)	80.4	40.6	−39.8
	Lexical	(182)	58.2	64.3	+6.1
	non-Lexical	(110)	43.6	52.7	+9.1
Non	NPIs	(8)	0.0	0.0	±0.0
	Disjunction	(20)	10.0	15.0	+5.0

Table 9: Evaluation results by linguistic phenomenon type. (non-)Lexical: problems that (do not) require lexical relations. Numbers in parentheses are numbers of problems.

accuracy on conjunction was opposite to that on disjunction. Downward inference examples involving disjunction are similar to upward inference ones; that is, inferences from a sentence to a shorter sentence are valid (e.g., *Not many campers have had a sunburn or caught a cold ⇒ Not many campers have caught a cold*). Thus, these results were also caused by addition of downward inference examples. Also, accuracy on problems annotated with *reverse* tags was apparently better without HELP because all examples are upward inferences embedded in a downward environment twice.

Table 9 also shows that accuracy on conditionals was better on upward inferences than that on downward inferences. This indicates that BERT might fail to capture the monotonicity property that conditionals create a downward entailing context in their scope while they create an upward entailing context out of their scope.

Regarding lexical knowledge, the data augmentation technique improved the performance much better on downward inferences which do not require lexical knowledge. However, among the 394 problems for which all models provided wrong answers, 244 problems are non-lexical inference problems. This indicates that some non-lexical inference problems are more difficult than lexical inference problems, though accuracy on non-lexical inference problems was better than that on lexical inference problems.

4.3 Discussion

One of our findings is that there is a type of downward inferences to which every model fails to provide correct answers. One such example is concerned with the contrast between *few* and *a few*. Among 394 problems for which all models provided wrong answers, 148 downward inference problems were problems involving the downward monotonicity operator *few* such as in the following example:

(7) P: *Few of the books had typical or marginal readers*
     H: *Few of the books had some typical readers*

We transformed these downward inference problems to upward inference problems in two ways: (i) by replacing the downward operator *few* with the upward operator *a few*, and (ii) by removing the downward operator *few*. We tested BERT using these transformed test sets. The results showed that BERT predicted the same answers for the transformed test sets. This suggests that BERT does not understand the difference between the downward operator *few* and the upward operator *a few*.

The results of crowdsourcing tasks in Section 3.1.3 showed that some downward inferences can naturally be performed in human reasoning. However, we also found that the MultiNLI training set (Williams et al., 2018), which is one of the dataset created from naturally-occurring texts, contains only 77 downward inference problems, including the following one.[5]

(8) P: *No racin' on the Range*
     H: *No horse racing is allowed on the Range*

One possible reason why there are few downward inferences is that certain pragmatic factors can block people to draw a downward inference. For instance, in the case of the inference problem in (9), unless the added disjunct in H, i.e., *a small cat with green eyes*, is salient in the context, it would be difficult to draw the conclusion H from the premise P.

(9) P: *I saw a dog*
     H: *I saw a dog or a small cat with green eyes*

Such pragmatic factors would be one of the reasons why it is difficult to obtain downward inferences in naturally occurring texts.

---

[5]The MultiNLI training set has 1,700 inference problems where the downward entailing operators *no* and *never* occur in both the premise and the hypothesis, but most of them are not an instance of downward inferences.

5 Conclusion

We introduced a large monotonicity entailment dataset, called MED. To illustrate the usefulness of MED, we tested state-of-the-art NLI models, and found that performance on the new test set was substantially worse for all state-of-the-art NLI models. In addition, the accuracy on downward inferences was inversely proportional to the one on upward inferences.

An experiment with the data augmentation technique showed that accuracy on upward and downward inferences depends on the proportion of upward and downward inferences in the training set. This indicates that current neural models might have limitations on their generalization ability in monotonicity reasoning. We hope that the MED will be valuable for future research on more advanced models that are capable of monotonicity reasoning in a proper way.

Acknowledgement

References

Lasha Abzianidze, Johannes Bjerva, Kilian Evang, Hessel Haagsma, Rik van Noord, Pierre Ludmann, Duc-Duy Nguyen, and Johan Bos. 2017. The Parallel Meaning Bank: Towards a multilingual corpus of translations annotated with compositional meaning representations. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL-2017)*, pages 242–247.

Jon Barwise and Robin Cooper. 1981. Generalized quantifiers and natural language. *Linguistics and Philosophy*, 4:159–219.

Johan van Benthem. 1983. Determiners and logic. *Linguistics and Philosophy*, 6(4):447–478.

Daniel Bonevac, Nicholas M. Asher, and Robert C. Koons. 1999. *Logic, Sets, and Functions*. Kendall Hunt Publishing.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015a. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP-2015)*, pages 632–642.

Samuel R. Bowman, Christopher Potts, and Christopher D. Manning. 2015b. Recursive neural networks can learn logical semantics. In *Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality*, pages 12–21.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Diana Inkpen, and Si Wei. 2018. Neural natural language inference models enhanced with external knowledge. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL-2018)*, pages 2406–2417.

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 (ACL-2017)*, pages 1657–1668.

Robin Cooper, Richard Crouch, Jan van Eijck, Chris Fox, Josef van Genabith, Jan Jaspers, Hans Kamp, Manfred Pinkal, Massimo Poesio, Stephen Pulman, et al. 1994. FraCaS–a framework for computational semantics. *Deliverable*, D6.

Ido Dagan, Dan Roth, Mark Sammons, and Fabio Massimo Zanzotto. 2013. *Recognizing Textual Entailment: Models and Applications*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Milica Denic, Vincent Homer, Daniel Rothschild, and Emmanuel Chemla. 2019. The influence of polarity items on inferential judgment. *Semantic Archive*, Archive/WY4OTMzO.

Jacob Devlin, Chang Ming-Wei, Lee Kenton, and Toutanova Kristina. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACLHLT-2019)*.

Richard Evans, David Saxton, David Amos, Pushmeet Kohli, and Edward Grefenstette. 2018. Can neural networks understand logical entailment? In *International Conference on Learning Representations (ICLR-2018)*.

Yaroslav Fyodorov, Yoad Winter, and Nissim Francez. 2003. Order-based inference in natural logic. *Logic Journal of the IGPL*, 11(4):385–416.

Bart Geurts. 2003. Reasoning with quantifiers. *Cognition*, 86:223–251.

Bart Geurts and Frans van der Slik. 2005. Monotonicity and processing load. *Journal of semantics*, 22(1):97–117.

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 (ACL-2018)*, pages 650–655.

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 (NAACLHLT-2018)*, pages 107–112.

Irene Heim and Angelika Kratzer. 1998. *Semantics in Generative Grammar*. Blackwell, Oxford.

Jack Hoeksema. 1986. Monotonicity phenomena in natural language. *Linguistic Analysis*, 16(1):25–40.

Hai Hu and Lawrence S. Moss. 2018. Polarity computations in flexible categorial grammar. In *Proceedings of the 7th Joint Conference on Lexical and Computational Semantics (*SEM-2018)*.

Thomas Icard and Lawrence Moss. 2014. Recent progress in monotonicity. *LILT*, 9(7):167–194.

Marc Moens and Mark Steedman. 1988. Temporal ontology and temporal reference. *Computational Linguistics*, 14(2):15–28.

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 (ACL-2018)*, pages 2340–2353.

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 (EMNLP-2016)*, pages 2249–2255.

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

Mark Steedman. 2000. *The Syntactic Process*. MIT Press.

Anna Szabolcsi, Lewis Bott, and Brian McElree. 2008. The effect of negative polarity items on inference verification. *Journal of semantics*, 25(4):411–450.

Masatoshi Tsuchiya. 2018. Performance impact caused by hidden bias of training data for recognizing textual entailment. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC-2018)*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the International Conference on Learning Representations (ICLR-2019)*.

Zhiguo Wang, Wael Hamza, and Radu Florian. 2017. Bilateral multi-perspective matching for natural language sentences. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI-2017)*, pages 4144–4150.

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 (NAACLHLT-2018)*, pages 1112–1122.

Yoad Winter. 2016. *Elements of Formal Semantics: An Introduction to the Mathematical Theory of Meaning in Natural Language*. Edinburgh University Press.

Hitomi Yanaka, Koji Mineshima, Daisuke Bekki, Kentaro Inui, Satoshi Sekine, Lasha Abzianidze, and Johan Bos. 2019. HELP: A dataset for identifying shortcomings of neural models in monotonicity reasoning. In *Proceedings of the 8th Joint Conference on Lexical and Computational Semantics (*SEM-2019)*.

Masashi Yoshikawa, Hiroshi Noji, and Yuji Matsumoto. 2017. A* CCG parsing with a supertag and dependency factored model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL-2017)*, pages 277–287.

Anna Zamansky, Nissim Francez, and Yoad Winter. 2006. A 'natural logic' inference system using the Lambek calculus. *Journal of Logic, Language and Information*, 15(3):273–295.