Logical Inferences with Comparatives and Generalized Quantifiers

Izumi Haruta[1]
haruta.izumi@is.ocha.ac.jp

Koji Mineshima[2]
minesima@abelard.flet.keio.ac.jp

Daisuke Bekki[1]
bekki@is.ocha.ac.jp

[1]Ochanomizu University, Tokyo, Japan
[2]Keio University, Tokyo, Japan

Abstract

Comparative constructions pose a challenge in Natural Language Inference (NLI), which is the task of determining whether a text entails a hypothesis. Comparatives are structurally complex in that they interact with other linguistic phenomena such as quantifiers, numerals, and lexical antonyms. In formal semantics, there is a rich body of work on comparatives and gradable expressions using the notion of degree. However, a logical inference system for comparatives has not been sufficiently developed for use in the NLI task. In this paper, we present a compositional semantics that maps various comparative constructions in English to semantic representations via Combinatory Categorial Grammar (CCG) parsers and combine it with an inference system based on automated theorem proving. We evaluate our system on three NLI datasets that contain complex logical inferences with comparatives, generalized quantifiers, and numerals. We show that the system outperforms previous logic-based systems as well as recent deep learning-based models.

1 Introduction

Natural Language Inference (NLI), or Recognizing Textual Entailment (RTE), is the task of determining whether a text entails a hypothesis and has been actively studied as one of the crucial tasks in natural language understanding. In recent years, systems based on deep learning (DL) have been developed by crowdsourcing large datasets such as Stanford Natural Language Inference (SNLI) (Bowman et al., 2015) and Multi-Genre Natural Language Inference (MultiNLI) (Williams et al., 2018) and have achieved high accuracy. NLI datasets focusing on complex linguistic phenomena, such as negation, antonyms, and numerals, have also been developed (Naik et al., 2018).

However, it has been pointed out that these datasets contain various biases that can be exploited by DL models (Dasgupta et al., 2018; McCoy et al., 2019), including easily classifying numerical expressions in inference (Liu et al., 2019) and answering by only looking at a hypothesis (Gururangan et al., 2018). This suggests that the success of NLI models to date has been overestimated and that tasks remain unresolved.

To handle inferences involving various linguistic phenomena, there are also studies to probe the effects of additional training using artificially constructed data (Dasgupta et al., 2018; Richardson et al., 2020). However, in the case of structurally complex inferences involving comparisons and numerical expressions, there is a myriad of ways to combine possible inference patterns. For example, consider the following inference.

(1)
P_1: John is taller than 6 feet.
P_2: Bob is shorter than 5 feet.

H: Bob is not taller than John. (Yes)

To correctly derive H from P_1 and P_2, it is necessary to capture the predicate-argument structures of the sentences, antonyms (*tall*, *short*), numerical expressions, and negation. Note that if the hypothesis sentence H is changed to *John is not taller than Bob*, the correct answer is not an entailment (Yes) but rather a contradiction (No); even if numerical expressions are excluded, the number of combinations of sentence patterns that produces this kind of reasonable inference is enormous.

In another approach, unsupervised NLI systems based on various logics have been studied (Bos, 2008; MacCartney and Manning, 2008; Mineshima et al., 2015; Abzianidze, 2016). However, the accuracies of these systems on comparative constructions are relatively low (see Section 3). Although there have been detailed discussions in formal semantics taking into account the

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop, pages 263–270

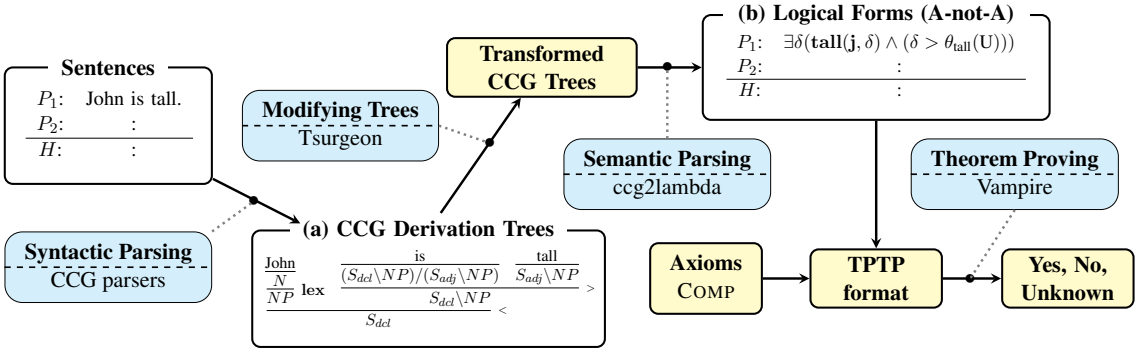


Figure 1: Overview of the proposed method. The premises and hypothesis are mapped to logical forms based on A-not-A analysis via CCG parsing and tree transformation; then a theorem prover judges *yes*, *no*, or *unknown* with the axioms for comparatives.

complexity associated with adjectives and comparative expressions (Cresswell, 1976; Kennedy, 1997; Heim, 2000; Lassiter, 2017), such theories have not yet been implemented in NLI systems. Also, some logic-based NLI systems handle comparatives (Chatzikyriakidis and Bernardy, 2019; Haruta et al., 2019), but these systems do not implement a parser and/or a prover.

The goal of this study is to fill this gap by implementing a formal compositional semantics based on the so-called A-not-A analysis (Seuren, 1973; Klein, 1980, 1982; Schwarzschild, 2008), which maps various comparative constructions in English to logical forms (LFs) via CCG (Steedman, 2000) derivation trees. Based on this, we present an inference system that computes complex logical inference over comparatives, generalized quantifiers, and numerals.[1] For evaluation, we use the FraCaS test set (Cooper et al., 1994), which contains various linguistically challenging inferences, and the Monotonicity Entailment Dataset (MED) (Yanaka et al., 2019), which contains inferences with generalized quantifiers. We also construct a new test set, the Comparative and Adjective Dataset (CAD), which extends FraCaS and collects both single-premise and multi-premise inferences with comparatives. The experiments show that our system outperforms previous logic-based systems as well as recent DL models.

2 System overview

Figure 1 shows the pipeline of the proposed system. First, the input sentences are a set of premises $P_1, \ldots, P_n$ and a hypothesis H. Next, the CCG derivation trees are obtained using CCG parsers. Derivation trees are modified to derive appropriate LFs based on A-not-A analysis. We use the semantic parsing system ccg2lambda (Martínez-Gómez et al., 2016) based on λ-calculus to obtain LFs, which are then converted to the Typed First-order Form (TFF) of the Thousands of Problems for Theorem Provers (TPTP) format (Sutcliffe, 2017), that is, a formal expression in first-order logic with equality and arithmetic operations. Finally, together with the axiom system COMP (Haruta et al., 2019) for comparatives and numerical expressions, a theorem prover checks whether $P_1 \wedge \cdots \wedge P_n \rightarrow H$ holds or not. The system output is *yes* (entailment), *no* (contradiction), or *unknown* (neutral).

2.1 Degree semantics: A-not-A analysis

In formal semantics, comparative and other gradable expressions are usually analyzed using the notion of *degree* (Cresswell, 1976).

(2) a. Ann is *taller* than Bob.
b. John is *5 feet tall*.
c. John is *tall*.

For example, the sentence (2a), in which the comparative form *taller* of the gradable adjective *tall* is used, compares the degree of height between two persons. (2b) is an expression that includes a specific height, which is the numerical expression *5 feet*. (2c) is a sentence using the positive form of the adjective, which can be regarded as representing a comparison with some implicit standard value. In degree-based semantics, such gradable adjectives are treated as two-place predicates that have entity and degree (Cresswell, 1976). For instance, (2b) is analyzed as $\mathbf{tall}(\mathbf{john}, \mathbf{5\ feet})$,

[1] GitHub repository with code and data: https://github.com/izumi-h/ccgcomp

Pattern	Example	Type	LF
(i)	1. John is tall.	Positive	$\exists\delta(\mathbf{tall}(\mathbf{john},\delta) \wedge (\delta > \theta_{\text{tall}}(\text{U})))$
	2. John is taller than Bob.	Increasing	$\exists\delta\,(\mathbf{tall}(\mathbf{john},\delta) \wedge \neg\,\mathbf{tall}(\mathbf{bob},\delta))$
	3. Ann has more children than Bob.	Numerical	$\exists\delta(\exists x(\mathbf{child}(x) \wedge \mathbf{have}(\mathbf{ann},x) \wedge \mathbf{many}(x,\delta))$ $\wedge\neg\exists x(\mathbf{child}(x) \wedge \mathbf{have}(\mathbf{bob},x) \wedge \mathbf{many}(x,\delta)))$
(ii)	1. John is as tall as Bob.	Equatives	$\forall\delta(\mathbf{tall}(\mathbf{bob},\delta) \rightarrow \mathbf{tall}(\mathbf{john},\delta))$
	2. Mary is 2 inches taller than Harry.	Differential	$\forall\delta(\mathbf{tall}(\mathbf{harry},\delta - 2'') \rightarrow \mathbf{tall}(\mathbf{mary},\delta))$
	3. John ate 3 more cookies than Bob.	Measure	$\forall\delta(\exists x(\mathbf{cookie}(x) \wedge \mathbf{eat}(\mathbf{bob},x) \wedge \mathbf{many}(x,\delta-3))$ $\rightarrow \exists x(\mathbf{cookie}(x) \wedge \mathbf{eat}(\mathbf{john},x) \wedge \mathbf{many}(x,\delta)))$

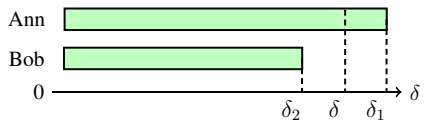
Table 1: Semantic representation of comparative constructions based on A-not-A analysis

where $\mathbf{tall}(x,\delta)$ is read as "x is *at least* as tall as degree δ" (Klein, 1991).

We use A-not-A analysis of comparatives, which analyzes (3a) as (3b).

(3) a. Ann is taller than Bob is.

b. $\exists\delta\,(\mathbf{tall}(\mathbf{ann},\delta) \wedge \neg\,\mathbf{tall}(\mathbf{bob},\delta))$



According to this analysis, (3a) is interpreted as saying that there exists a degree δ of height that Ann satisfies, but Bob does not. As shown in the figure in (3), this guarantees that Ann's height is greater than Bob's height. A-not-A analysis makes it possible to derive entailment relations between various comparative constructions in a simple way using first-order logic theorem provers.

Table 1 shows LFs for some example sentences using A-not-A analysis.[2] Here, LFs can be divided into two patterns. The examples in (i) in Figure 1 belong to the first type, where the degree of an individual **exceeds** a certain degree. For example, the sentence (i-2) means that the height of John is greater than the height of Bob. The sentence (i-3) means that the number of Ann's children exceeds the number of Bob's children. Under our analysis, this type of sentence is mapped to formulas of the form $\exists\delta(\cdots \wedge \cdots)$.

The second type includes the examples in (ii), which say that the degree of an individual is **greater than or equal to** a certain degree. For example, (ii-1) means that John's height is greater than or equal to Bob's height (Klein, 1982). The sentence (ii-3) means that the number of cookies John ate is 3 or more greater than the number of cookies that Bob ate; in other words, if Bob ate n cookies, then John ate at least $n+3$ cookies. Sentences of type (ii) are mapped to formulas of the form $\forall\delta(\cdots \rightarrow \cdots)$, as in Table 1.

[2]For the positive form, the comparison class (Klein, 1982) is relevant to determining the standard of degree (e.g., tallness). We use a default comparison class such as θ_{tall} in our implementation and leave the determination of comparison classes and relevant standards (cf. Pezzelle and Fernández, 2019) to future work.

2.2 Compositional semantics in CCG

In CCG, the mapping from syntax to semantics is defined by assigning syntactic categories to words (Steedman, 2000); the LF of a sentence is then compositionally derived using λ-calculus. However, there is a gap between the syntactic structures assumed in formal semantics and the output derivation trees of existing CCG parsers, i.e., statistical parsers trained on CCGBank (Hockenmaier and Steedman, 2007). For this reason, we modify the derivation trees provided by CCG parsers in post-processing. There are several types of modifications.

Syntactic features The first modification is to add syntactic features to CCG categories. For example, in the default CCG trees, a nominal adjective (*a tall boy*) has the category N/N, while a predicate adjective (*John is tall*) has the category $S_{adj}\backslash NP$. To provide a uniform degree semantics to both constructions, we rewrite N/N as N_{adj}/N for the category of nominal adjectives.

Multiword expressions Compound expressions for comparatives and quantifiers are combined as one word, such as *a few, a lot of*, and *at most*.

Empty categories We insert an empty category to systematically derive the LFs of the two patterns described in Table 1. The distinction between patterns (i) and (ii) can be controlled by an expression appearing in the adjunct position of an adjective phrase, for example, a degree modifier such as *very* or a numerical expression such as *2 cm*.

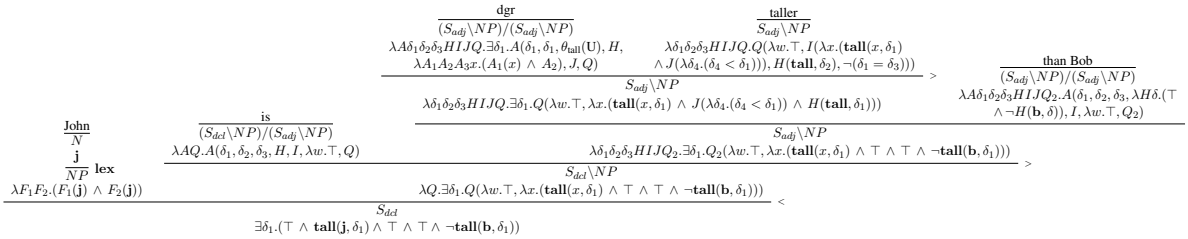


Figure 2: Derivation tree of *John is* (*dgr*) *taller than Bob*.

Example	LF
Mary has *many* dogs.	$\exists x(\mathbf{have}(\mathbf{mary}, x) \wedge \mathbf{dog}(x) \wedge \mathbf{many}(x, \theta_{\text{many}}(x)))$
Ann read *two* books.	$\exists x(\mathbf{read}(\mathbf{ann}, x) \wedge \mathbf{book}(x) \wedge \mathbf{many}(x, 2))$
Most apples are red.	$\exists \delta(\exists x(\mathbf{apple}(x) \wedge \mathbf{red}(x) \wedge \mathbf{many}(x, \delta)) \wedge \neg\exists x(\mathbf{apple}(x) \wedge \neg\mathbf{red}(x) \wedge \mathbf{many}(x, \delta)))$
No more than five boys ran.	$\neg\exists x\exists \delta(\mathbf{boy}(x) \wedge \mathbf{many}(x, \delta) \wedge (5 < \delta) \wedge \mathbf{run}(x))$

Table 2: LFs of generalized quantifiers based on our degree semantics

When such an adjunct expression does not appear, we insert an empty category *dgr* into the adjunct position, which is used to derive the desired LF compositionally. Figure 2 shows an example of a modified derivation tree containing an empty element *dgr* for increasing comparatives. Similarly, we use two other types of empty categories for equatives (e.g., *as tall as*) and the positive form.

2.3 Generalized quantifiers

The analysis of comparatives by the degree-based semantics described above can naturally be extended to generalized quantifiers. In the traditional analysis (Barwise and Cooper, 1981), generalized quantifiers such as *many, few, more than*, and *most* are analyzed as denoting a relation between sets. Alternatively, an analysis based on degree semantics has been developed, which represents expressions such as *many* and *few* as adjectives (Partee, 1988; Rett, 2018) and *most* as the superlative form of *many* (Hackl, 2000; Szabolcsi, 2010). We recast this alternative analysis in our degree-based semantics. Table 2 shows the LFs for some examples. We use the binary predicate $\mathbf{many}(x, n)$, which reads "x is composed of (at least) n entities". *Most A are B* is analyzed as meaning "More than half of *A* is *B*", following the standard truth-condition (Hackl, 2000).

3 Experiments

3.1 Experimental settings

For CCG parsing, we use two CCG parsers, namely, C&C (Clark and Curran, 2007) and depccg (Yoshikawa et al., 2017), to mitigate parsing errors. If two parsers output a different answer, we choose the system answer in the following way: if one answer is *yes* (resp. *no*) and the other is *unknown*, the system answer is *yes* (resp. *no*); if one answer is *yes* and the other is *no*, then the system answer is *unknown*. For POS tagging, we use the C&C POS tagger for C&C and spaCy[3] for depccg.

To implement compositional semantics, we use ccg2lambda[4]. We extend the semantic templates proposed in Mineshima et al. (2015) to handle linguistic phenomena based on degree-based semantics. The total number of lexical entries assigned to CCG categories is 106, and the number of entries directly assigned to particular words (e.g., *than* and *as* for comparatives and items for quantifiers) is 214. For tree transformation, we use Tsurgeon (Levy and Andrew, 2006). We use 74 entries (rewriting clauses) in the Tsurgeon script. For theorem proving, we use Vampire[5], which accepts TFF forms with arithmetic operations.

For evaluation, we use three datasets. First, FraCaS (Cooper et al., 1994) is a dataset comprising nine sections, each of which contains semantically challenging inferences related to various linguistic phenomena. In this study, we use three sections: Generalized Quantifiers (GQ; 73 problems), Adjectives (ADJ; 22 problems), and Comparatives (COM; 31 problems). The distribution of gold answer labels for the three sections is (*yes*/*no*/*unknown*) = (36/5/32), (9/6/7), (19/9/3), respectively.

Second, MED[6] is a dataset that contains in-

[3] https://github.com/explosion/spaCy
[4] https://github.com/mynlp/ccg2lambda
[5] https://github.com/vprover/vampire
[6] https://github.com/verypluming/MED

FraCaS-235 (COMPARATIVES) Gold answer: Yes	
Premise 1	ITEL won more orders than APCOM.
Premise 2	APCOM won ten orders.
Hypothesis	ITEL won at least eleven orders.
MED-1085 Gold answer: Unknown	
Premise 1	No more than fifty campers have caught a cold.
Hypothesis	No more than fifty campers have had a sunburn or caught a cold.
CAD-011 (COMPARATIVES) Gold answer: Yes	
Premise 1	Alex is not as tall as Chris is.
Hypothesis	Chris is taller than Alex is.
CAD-034 (ADJECTIVES) Gold answer: Yes	
Premise 1	Bob is 4 feet tall.
Premise 2	John is taller than Bob.
Hypothesis	John is more than 4 feet tall.

Table 3: Examples of entailment problems from the FraCaS, MED, and CAD test sets

FraCaS				
Section		GQ	ADJ	COM
#All		73	22	31
#Single		44	15	16
Majority		.48	.39	.61
Logic	MN	.77	.68	.48
	LP	.93	.73	-
	NL	.98*	.80*	.81*
	Ours	.92	.86	.77
	+rule	**.95**	**.95**	**.84**
DL	LSTM	.64*	.47*	.56*
	DA	.59	.45	.61
	BERT	.64	.45	.58

Table 4: Accuracy on the FraCaS test suite: '#All' shows the number of all problems and '#Single' the number of single-premise problems.

ferences with quantifiers (so-called monotonicity inferences). We use a subset (498 problems) of MED that does not require world knowledge and commonsense reasoning; these problems were collected from various linguistics papers. The distribution of the gold answer is (*yes*/*unknown*) = (215/283).

Because there are only 31 problems for comparatives in FraCaS, we created the CAD test set consisting of 105 problems, which focuses on comparatives and numerical constructions not covered by FraCaS. We collected a set of inferences (9 problems) from a linguistics paper (Klein, 1982) and created more problems by adding negation, using degree modifiers (e.g., *very*), changing numerical expressions, replacing positive and negative adjectives (e.g., *large* to *small*), and swapping the premise and hypothesis of an inference. Of the 105 problems 50 are single-premise problems, and 55 are multi-premise problems. The distribution of gold answer labels is (*yes*/*no*/*unknown*) = (50/17/38). All of the gold labels were checked by an expert in linguistics. Table 3 shows some example problems.

3.2 Results and discussion

FraCaS test suite Table 4 shows the experimental results on FraCaS. *Majority* is the accuracy of the majority baseline and *Ours* the accuracy of our system. Some errors were caused by failing to assign correct POS tags and lemmas to comparatives; for example, *cleverer* is wrongly assigned *NN* rather than *JJR* (FraCaS-217). To estimate the upper bound of the accuracy of our system by reducing error propagation, we added hand-coded rules to assign correct POS tags and lemmas (14 words). We also added two rules to join multi-word expressions to derive correct logical forms (*law lecturer* and *legal authority* for FraCaS-214, 215). In Table 4, *+rule* shows the improvement in accuracy realized by adding these rules.

We compare our system with previous logic-based NLI systems as well as three popular DL models. For logic-based systems, we use MN (Mineshima et al., 2015) and LP (Abzianidze, 2016) based on CCG parsers and theorem proving and NL (MacCartney and Manning, 2008) based on Natural Logic. NL is evaluated on single-premise problems only (indicated by *). Our system accepts both single-premise and multiple-premise problems and outperforms the previous logic-based systems on the adjectives and comparatives sections. Our system solves complex reasoning problems with multiple premises involving comparatives and numerical expressions, such as FraCaS-235 in Table 3, for which the previous systems were unable to give a correct answer.

For DL models, LSTM is the performance of a long short-term memory model trained on SNLI, which is reported in Bowman (2016) (only evaluated on single-premise problems). We also tested the Decomposable Attention (DA) model (Parikh et al., 2016), a simple attention-based model trained on SNLI. We used the implementation provided in AllenNLP (Gardner et al., 2018). Finally, BERT is the performance of a BERT model (Devlin et al., 2019). We used the bert-base-cased model fine-tuned with MultiNLI. We used the code available at the orig-

MED	
#All	498
Majority	.60
BERT+	.54
BERT	.56
Ours	**.84**

CAD	
#All	105
Majority	.48
DA	.51
BERT	.55
Ours	**.77**

Table 5: Accuracy on the MED and CAD datasets

inal GitHub repository.[7] Our system outperforms the three DL models by large margins.[8]

MED and CAD datasets Table 5 shows the results on MED and CAD. For MED, we compared our system with a BERT model fine-tuned with MultiNLI (**BERT**) and a BERT model with data augmentation (approximately 36K) in addition to MultiNLI (**BERT+**), both being tested in Yanaka et al. (2019). For CAD, we evaluated DA and BERT. The results show that our system achieved high accuracy on the logical inferences with adjectives, comparatives, and generalized quantifiers.

Table 6 shows examples that were solved by our system but not by DA and BERT. The DL models were particularly difficult to handle inferences related to antonyms (e.g., FraCaS-209) and numerical expressions (e.g., CAD-001). Indeed, the results on the DL models were predictable because these models were trained on datasets (SNLI and MultiNLI) that do not target the logical and numerical inferences we are concerned with in this study. However, it is fair to say that it is very challenging to generate effective training data to handle various complex inferences with comparatives, numerals, and generalized quantifiers.

There were some problems that our system could not solve. For FraCaS, the accuracy for the comparative section (COM) was relatively low (.84). This is because this section contains linguistically challenging phenomena such as clausal comparatives (FraCaS-239, 240, 241) and attributive comparatives (FraCaS-244, 245). For MED, the present system does not handle downward monotonic quantifiers (e.g., *less than*), non-monotonic quantifiers (e.g., *exactly*), and negative polarity items (e.g., *any*). Furthermore, the system needs to be extended to deal with linguistic phenomena such as comparative subdeletion and quantified comparatives that appear in CAD. To address these problems, further improvement of the CCG parsers will be needed.

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

[8] For DA and BERT, we evaluated multiple-premise problems by two methods: simply concatenating two or more premises (e.g., "S_1. S_2.") and by inserting *and* and commas between sentences (e.g., "S_1 *and* S_2."). Comparing the two methods, we used the better accuracy for each problem in MED and CAD in Table 4 and 5.

FraCaS-209 (ADJECTIVES) Gold answer: No	
Premise 1	Mickey is a small animal.
Premise 2	Dumbo is a large animal.
Hypothesis	Mickey is larger than Dumbo.
MED-1021 Gold answer: Unknown	
Premise 1	More than five campers have had a sunburn or caught a cold.
Hypothesis	More than five campers have caught a cold.
CAD-001 Gold answer: Yes	
Premise 1	John is 5 cm taller than Bob.
Premise 2	Bob is 170 cm tall.
Hypothesis	John is 175 cm tall.
CAD-103 Gold answer: Unknown	
Premise 1	Bob is not tall.
Premise 2	John is not tall.
Hypothesis	John is taller than Bob.

Table 6: Examples of problems solved by our system but not by the DL models. The answers of the DL models are: *yes* (DA and BERT) for FraCaS-209; *yes* (**BERT** and **BERT+**) for MED-1021; *no* (DA and BERT) for CAD-001; *yes* (DA) and *no* (BERT) for CAD-103.

4 Conclusion

In this study, we presented an end-to-end logic-based inference system for handling complex inferences with comparatives, quantifiers, and numerals. The entire system is transparently composed of several modules and can solve complex inferences for the right reason. In future work, we will extend our analysis to cover the more complex constructions mentioned in Section 3. We are also considering combining our system with an abduction mechanism that uses large knowledge bases (Yoshikawa et al., 2019) for handling commonsense reasoning with external knowledge.

Acknowledgments We are grateful to Hitomi Yanaka for sharing the detailed results on the MED dataset and Masashi Yoshikawa for continuous support. We also thank the three anonymous reviewers for their helpful comments and feedback. This work was supported by JSPS KAKENHI Grant Number JP18H03284.

References

Lasha Abzianidze. 2016. Natural solution to FraCaS entailment problems. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics (*SEM)*, pages 64–74.

John Barwise and Robin Cooper. 1981. Generalized quantifiers and natural language. *Linguistics and Philosophy*, 4(2):159–219.

Johan Bos. 2008. Wide-coverage semantic analysis with Boxer. In *Proceedings of the 2008 Conference on Semantics in Text Processing (STEP)*, pages 277–286.

Samuel R. Bowman. 2016. *Modeling Natural Language Semantics in Learned Representations*. Ph.D. thesis, Stanford University.

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 (EMNLP)*, pages 632–642.

Stergios Chatzikyriakidis and Jean-Philippe Bernardy. 2019. A wide-coverage symbolic natural language inference system. In *Proceedings of the 22nd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 298–303.

Stephen Clark and James R Curran. 2007. Wide-coverage efficient statistical parsing with CCG and log-linear models. *Computational Linguistics*, 33(4):493–552.

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.

Max J Cresswell. 1976. The semantics of degree. In Barbara Partee, editor, *Montague Grammar*, pages 261–292. Academic Press.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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 (NAACL-HLT)*, pages 4171–4186.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. Allennlp: A deep semantic natural language processing platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6.

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 (NAACL-HLT)*, pages 107–112.

Martin Hackl. 2000. *Comparative Quantifiers*. Ph.D. thesis, Massachusetts Institute of Technology.

Izumi Haruta, Koji Mineshima, and Daisuke Bekki. 2019. A CCG-based compositional semantics and inference system for comparatives. In *Proceedings of 33rd Pacific Asia Conference on Language, Information and Computation (PACLIC 33)*, pages 67–76.

Irene Heim. 2000. Degree operators and scope. In *Proceedings of the 10th Semantics and Linguistic Theory (SALT 10)*, pages 40–64.

Julia Hockenmaier and Mark Steedman. 2007. CCGbank: A corpus of CCG derivations and dependency structures extracted from the Penn Treebank. *Computational Linguistics*, 33(3):355–396.

Christopher Kennedy. 1997. *Projecting the Adjective: The Syntax and Semantics of Gradability and Comparison*. Ph.D. thesis, University of California, Santa Cruz.

Ewan Klein. 1980. A semantics for positive and comparative adjectives. *Linguistics and Philosophy*, 4(1):1–45.

Ewan Klein. 1982. The interpretation of adjectival comparatives. *Journal of Linguistics*, 18(1):113–136.

Ewan Klein. 1991. Comparatives. In Arnim von Stechow and Dieter Wunderlich, editors, *Semantics: An International Handbook of Contemporary Research*, pages 673–691. de Gruyter, Berlin.

Daniel Lassiter. 2017. *Graded Modality: Qualitative and Quantitative Perspectives*. Oxford University Press.

Roger Levy and Galen Andrew. 2006. Tregex and Tsurgeon: tools for querying and manipulating tree data structures. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, pages 2231–2234.

Nelson F. Liu, Roy Schwartz, and Noah A. Smith. 2019. Inoculation by fine-tuning: A method for analyzing challenge datasets. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 2171–2179.

Bill MacCartney and Christopher D. Manning. 2008. Modeling semantic containment and exclusion in natural language inference. In *Proceedings of the 22nd International Conference on Computational Linguistics (COLING)*, pages 521–528.

Pascual Martínez-Gómez, Koji Mineshima, Yusuke Miyao, and Daisuke Bekki. 2016. ccg2lambda: A compositional semantics system. In *Proceedings of ACL-2016 System Demonstrations*, pages 85–90.

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 (ACL)*, pages 3428–3448.

Koji Mineshima, Pascual Martínez-Gómez, Yusuke Miyao, and Daisuke Bekki. 2015. Higher-order logical inference with compositional semantics. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2055–2061.

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 (COLING)*, 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)*, pages 2249–2255.

Barbara H. Partee. 1988. Many quantifiers. In *Proceedings of the 5th Eastern States Conference on Linguistics (ESCOL)*, pages 383–402.

Sandro Pezzelle and Raquel Fernández. 2019. Is the red square big? MALeViC: Modeling adjectives leveraging visual contexts. 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 2865–2876.

Jessica Rett. 2018. The semantics of *many, much, few, and little*. *Language and Linguistics Compass*, 12(1).

Kyle Richardson, Hai Hu, Lawrence S Moss, and Ashish Sabharwal. 2020. Probing natural language inference models through semantic fragments. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Roger Schwarzschild. 2008. The semantics of comparatives and other degree constructions. *Language and Linguistics Compass*, 2(2):308–331.

Pieter A. M. Seuren. 1973. The comparative. In F. Kiefer and N. Ruwet, editors, *Generative Grammar in Europe*, pages 528–564. Riedel, Dordrecht.

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

Geoff Sutcliffe. 2017. The TPTP Problem Library and Associated Infrastructure. *Journal of Automated Reasoning*, 59(4):483–502.

Anna Szabolcsi. 2010. *Quantification*. Cambridge University Press.

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 (NAACL-HLT)*, pages 1112–1122.

Hitomi Yanaka, Koji Mineshima, Daisuke Bekki, Kentaro Inui, Satoshi Sekine, Lasha Abzianidze, and Johan Bos. 2019. Can neural networks understand monotonicity reasoning? In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 31–40.

Masashi Yoshikawa, Koji Mineshima, Hiroshi Noji, and Daisuke Bekki. 2019. Combining axiom injection and knowledge base completion for efficient natural language inference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 7410–7417.

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)*, pages 277–287.