Towards Broad-coverage Meaning Representation: The Case of Comparison Structures

Omid Bakhshandeh
University of Rochester
omidb@cs.rochester.edu

James F. Allen
University of Rochester / IHMC
james@cs.rochester.edu

1 Introduction

Representing the underlying meaning of text has been a long-standing topic of interest in computational linguistics. Recently there has been a renewed interest in computational modeling of meaning for various tasks such as semantic parsing (Zelle and Mooney, 1996; Berant and Liang, 2014). Open-domain and broad-coverage semantic representation (Banarescu et al., 2013; Bos, 2008; Allen et al., 2008) is essential for many language understanding tasks such as reading comprehension tests and question answering.

One of the most common way for expressing evaluative sentiment towards different entities is to use comparison. Comparison can happen in very simple structures such as *'John is taller than Susan'*, or more complicated constructions such as *'The table is longer than the sofa is wide'*. So far the computational semantics of comparatives and how they affect the meaning of the surrounding text has not been studied effectively. That is, the difference between the existing semantic and syntactic representation of comparatives has not been distinctive enough for enabling deeper understanding of a sentence. For instance, the general logical form representation of the sentence *'John is taller than Susan'* using the Boxer system (Bos, 2008) is the following:

$$\begin{aligned} &named(x0, john, per) \\ &\quad \& \ named(x1, susan, nam) \\ &\qquad \& than(taller(x0), x1) \end{aligned} \tag{1}$$

Clearly, the above meaning representation does not fully capture the underlying semantics of the adjective 'tall' and what it means to be 'taller'. A human reader can easily infer that the 'height' attribute of John is greater than Susan's. Capturing the underlying meaning of comparison structures, as opposed to their surface wording, is crucial for accurate evaluation of qualities and quantities. Consider a more complex comparison example, *'The pizza was great, but it was still worse than the sandwich'*. The state-of-the-art sentiment analysis system (Manning et al., 2014) assigns an overall 'negative' sentiment value to this sentence, which clearly lacks the understanding of the comparison happening in the sentence.

My Mazda	drove	faster than his Hyundai
Self_mover	**Self_motion**	Manner

Figure 1: The frame-semantic parsing of the sentence *My Mazda drove faster than his Hyundai.*

As another example, consider the generic meaning representation of the sentence *'My Mazda drove faster than his Hyundai'*, according to frame semantic parsing using Semafor[1] tool (Das et al., 2014) as depicted in Figure 1. It is evident that this meaning representation does not fully capture how the semantics of the adjective *fast* relates to the *driving* event, and what it actually means for a car to drive *faster than* another car. More importantly, there is an ellipsis in this sentence, the resolution of which results in complete reading of *'My Mazda drove faster than his Hyundai drove fast'*, which is in no way captured in Figure 1[2].

[1] http://demo.ark.cs.cmu.edu/parse

[2] The same shortcomings are shared among other generic meaning representations such as LinGO English Resource

Proceedings of EMNLP 2016 Workshop on Uphill Battles in Language Processing: Scaling Early Achievements to Robust Methods, pages 27–31,
Austin, TX, November 5, 2016. ©2016 Association for Computational Linguistics

Although the syntax and semantics of comparison in language have been studied in linguistics for a long time (Bresnan, 1973; Cresswell, 1976; Von Stechow, 1984), so far, computational modeling of the semantics of comparison components of natural language has not been developed fundamentally. The lack of such a computational framework has left the deeper understanding of comparison structures still baffling to the currently existing NLP systems. In this paper we summarize our efforts on defining a joint framework for comprehensive semantic representation of the comparison and ellipsis constructions. We jointly model comparison and ellipsis as inter-connected predicate-argument structures, which enables automatic ellipsis resolution. In the upcoming sections we summarize our main contributions to this topic.

2 A Comprehensive Semantic Framework for Comparison and Ellipsis

We introduce a novel framework for modeling the semantics of comparison and ellipsis as interconnected predicate-argument structures. According to this framework, comparison and ellipsis operators are the predicates, where each predicate has a set of arguments called its *semantic frame*. For example, in the sentence *‘[Sam] is the tallest [student] [in the gym]’*, the morpheme *-est* is the comparison operator (hence, the comparison predicate) and the entities in the brackets are the arguments.

2.1 Comparison Structures

2.1.1 Predicates

We consider two main categories of comparison predicates (Bakhshandeh and Allen, 2015; Bakhshandeh et al., 2016), Ordering and Extreme, each of which can grade any of the four parts of speech including adjectives, adverbs, nouns, and verbs.

• **Ordering**: Shows the ordering of two or more entities on a scale, with the following subtypes:

– Comparatives expressed by the morphemes *more/-er* and *less*, with ‘$>$’, ‘$<$’ indicating that one degree is greater or lesser than the other.

(1) The steak is tastier than the potatoes.

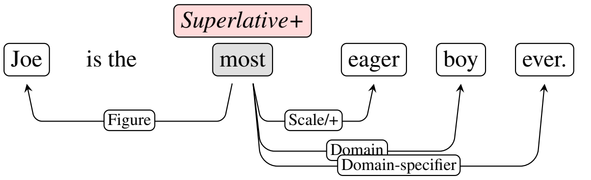

Figure 2: An example predicate-argument structure consisting of superlative predicate type and its corresponding semantic frame of arguments.

– Equatives expressed by *as* in constructions such as *as tall* or *as much*, with ‘$\geq$’ indicating that one degree equals or is greater than another.

(2) The Mazda drives as fast as the Nissan.

– Superlatives expressed by *most/-est* and *least*, indicates that an entity or event has the ‘highest’ or ‘lowest’ degree on a scale.

(3) That chef made the best soup.

The details of the Extreme type can be found in the earlier work (Bakhshandeh and Allen, 2015; Bakhshandeh et al., 2016).

2.1.2 Arguments

Each predicate takes a set of arguments that we refer to as the predicate’s ‘semantic frame’. Following are the main arguments included in our framework:

– Figure (Fig): The main role which is being compared.

– Ground: The main role Figure is compared to.

– Scale: The scale for the comparison, such as length, depth, speed. For a more detailed study on scales please refer to the work on learning adjective scales (Bakhshandeh and Allen, 2015).

Our framework also includes ‘Standard’, ‘Differential’, ‘Domain’, and ‘Domain Specifier’ argument types. Figure 2 shows an example meaning representations based on our framework.

2.2 Ellipsis Structures

As mentioned earlier in Section 1, resolving ellipsis in comparison structures is crucial for language understanding and failure to do so would deliver an incorrect meaning representation. In linguistics various subtypes of elliptical constructions are studied (Kennedy, 2003; Merchant, 2013; Yoshida et al., 2016). In our framework we mainly include six types which are seen in comparison struc-

Grammar (ERG) (Flickinger, 2011), Boxer (Bos, 2008), or AMR (Banarescu et al., 2013), among others.

tures (Bakhshandeh et al., 2016): ‘VP-deletion’, ‘Stripping’[3], ‘Pseudo-gapping’, ‘Gapping’, ‘Sluicing’, and ‘Subdeletion’. Ellipsis more often occurs in comparative and equative comparison constructions. A few examples of ellipsis in comparative constructions are as follows:

- **Comparatives**: Ellipsis site is the dependent clause headed by *than*. Three ellipsis possibilities for these clauses resuming (4) are shown below. The elided materials are written in subscript.

(4) **Mary drank more tea** ...

- VP-deletion (aka ‘Comparative Deletion’):
  ... than John did $_{\text{drink coffee}}$.
- Stripping (aka ‘Phrasal Comparative’):
  ... than John $_{\text{drank coffee}}$.
- Gapping:
  ... than John, $_{\text{drank } \textit{how-much}}$ coffee.
- Pseudogapping:
  ... than John did $_{\text{drank}}$ coffee.

Furthermore, we define three argument types for ellipsis, which help thoroughly construct the antecedent of the elided material by taking into account the existing words of the context sentence: Reference, Exclude, and How-much.

3 Data Collection Methodology

Given the new semantic representation, we aim at annotating corpora which then enables developing and testing models. The diversity and comprehensiveness of the comparison structures represented in our dataset is dependent on the genre of sentences comprising it. Earlier, we had experimented with annotating semantic structures on OntoNotes dataset (Bakhshandeh and Allen, 2015). Recently (Bakhshandeh et al., 2016), We have shifted our focus to actual product and restaurant reviews, which include many natural comparison instances. For this purpose we mainly use Google English Web Treebank[4] which comes with gold constituency parse trees. We augment this dataset with the Movie Reviews dataset (Pang and Lee, 2005), where we use Berkeley parser (Petrov et al., 2006) to obtain parse trees.

We trained linguists by asking them to read the semantic framework annotation manual as summarized in Section 2. The annotations were done via our interactive two-stage tree-based annotation tool. For this task, the annotations were done on top of constituency parse trees. This process yielded a total of 2,800 annotated sentences. Figure 3 visualizes the distribution of predicate types from the various resources. As this Figure shows, reviews are indeed a very rich resource for comparisons, having more comparison instances than any other resource of even a bigger size. There are a total of 5,564 comparison arguments in our dataset, with ‘scale’ and ‘figure’ being the majority types. The total number of ellipsis predicates is 240, with 197 Stripping, 31 VP-deletion and 12 Pseudo-gapping.

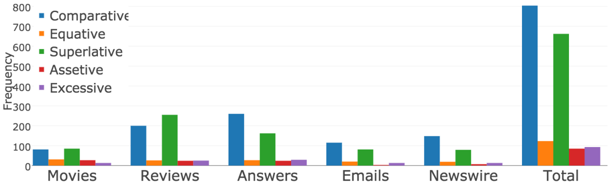

Figure 3: The number of various predicate types across different resources.

	ILP Model		
	P	**R**	**F1**
Average	0.72/0.78	0.91/0.97	0.76/0.80
	Baseline		
Average	0.62/0.64	0.87/0.97	0.66/0.69

Table 1: Predicate prediction Precision (P), Recall (R) and F1 scores on test set, averaged across all predicate types. Each cell contains scores according to Exact/Head measurement.

4 Predicting Semantic Structures

We model the prediction problem as a joint predicate-argument prediction of comparison and ellipsis structures. In a nutshell, we define a globally normalized model for the probability distribution of comparison and ellipsis labels over all parse tree nodes as follows:

$$p_C(c|v,T,\theta_C) \propto exp(\boldsymbol{f}_C(c,T)^T\boldsymbol{\theta_C}) \qquad (2)$$

$$p_{A_c}(a_c|c,e,v,T,\theta_{a_c}) \propto exp(\boldsymbol{f}_{A_C}(c,e,T)^T\boldsymbol{\theta_{a_c}}) \qquad (3)$$

where T is the underlying constituency tree, p_C is the probability of assigning predicate type c as the predicate type and p_{A_c} is the probability of assigning the argument type a_c as the argument type. In

[3] VP-deletion and stripping are the more frequent types.

[4] https://catalog.ldc.upenn.edu/LDC2012T13

	ILP Model (Exact/Head)			**ILP No Constraints** (Exact/Head)			**Baseline** (Exact/Head)		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
Average	0.37/0.61	0.54/0.87	0.43/0.71	0.01/0.01	0.86/1.00	0.10/0.10	0.20/0.42	0.36/0.73	0.25/0.52

Table 2: Results of argument prediction on test set, averaged across various argument types.

each of the above equations, f is the corresponding feature function. For predicates and the arguments the main features are lexical features and bigram features, among many others. θ_C, θ_E, θ_{a_c} is the parameters of the log-linear model. We calculate these parameters using Stochastic Gradient Descent algorithm.

For inference, we model the problem as a structured prediction task. Given the syntactic tree of a given sentence, for each node we first select the predicate type with the highest p_C. Then for each selected comparison predicate, we find the corresponding ellipsis predicate that has the highest p_E probability. We tackle the problem of argument assignment by Integer Linear Programming (ILP), where we pose domain-specific linguistic knowledge as constraints. Any specific comparison label calls for a unique set of constraints in the ILP formulation, which ensures the validity of predictions[5]. The details of this modeling can be found in earlier work (Bakhshandeh et al., 2016).

5 Experimental Results

We trained our ILP model on the train-dev part of the dataset (70%), and tested on the test set (30%). Evaluation is done against the reference gold annotation, with Exact and partial (Head) credits to annotating the constituency nodes. We mainly report on two models: our comprehensive ILP model (detailed in Section 4), and a rule-based baseline. In short, the baseline encodes the same linguistically motivated ILP constraints via rules and uses a few pattern extraction methods for finding comparison morphemes.

The average results on predicate prediction (across all types) is shown in Table 1. As the results show, overall, the scores are high for predicting the predicates, what is not shown here is ellipsis predicates being the most challenging. The baseline is competitive, which shows that the linguistic patterns can capture many of the predicate types. Our model performs the poorest on $Equatives$, achieving 71%/73% F1 score, which is a complex morpheme used in various linguistic constructions. Our analysis shows that the errors are often due to inaccuracies in automatically generated parse trees[6]. As you can see in Table 2, The task of predicting arguments is a more demanding task. The baseline performs very poorly at predicting the arguments. Our comprehensive ILP model consistently outperforms the *No Constraints* model, showing the effectiveness of our linguistically motivated ILP constraints.

6 Conclusion

In this work we summarized our work which focuses on an aspect of language with a very rich semantics: Comparison and Ellipsis. The current tools and methodologies in the research community are not able to go beyond surface-level shallow representations for comparison and ellipsis structures. We have developed widely usable comprehensive semantic theory of linguistic content of comparison structures. Our representation is broad-coverage and domain-independent, hence, can be incorporated as a part of any broad-coverage semantic parser (Banarescu et al., 2013; Allen et al., 2008; Bos, 2008) for augmenting their meaning representation.

Acknowledgment

We thank the anonymous reviewers for their comments. This work was supported in part by Grant W911NF-15-1-0542 with the US Defense Advanced Research Projects Agency (DARPA) and the Army Research Office (ARO).

References

James F. Allen, Mary Swift, and Will de Beaumont. 2008. Deep semantic analysis of text. In *Proceedings*

[5] For instance, the $Superlative$ predicate type never takes any $Ground$ arguments, or the argument $Standard$ is only applicable to the excessive predicate type.

[6] For example, challenging long review sentences with informal language.

of the 2008 Conference on Semantics in Text Processing, STEP '08, pages 343–354, Stroudsburg, PA, USA. Association for Computational Linguistics.

Omid Bakhshandeh and James Allen. 2015. Semantic framework for comparison structures in natural language. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 993–1002, Lisbon, Portugal, September. Association for Computational Linguistics.

Omid Bakhshandeh, Alexis Cornelia Wellwood, and James Allen. 2016. Learning to jointly predict ellipsis and comparison structures. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 62–74, Berlin, Germany, August. Association for Computational Linguistics.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract meaning representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186, Sofia, Bulgaria, August. Association for Computational Linguistics.

Jonathan Berant and Percy Liang. 2014. Semantic parsing via paraphrasing. In *Association for Computational Linguistics (ACL)*.

Johan Bos. 2008. Wide-coverage semantic analysis with boxer. In Johan Bos and Rodolfo Delmonte, editors, *Semantics in Text Processing. STEP 2008 Conference Proceedings*, Research in Computational Semantics, pages 277–286. College Publications.

Joan Bresnan. 1973. Syntax of the comparative clause construction in English. *Linguistic Inquiry*, 4(3):275–343.

Max Cresswell. 1976. The semantics of degree. *Barbara Hall Partee (ed.)*, pages 261–292.

Dipanjan Das, Desai Chen, AndrÃl' F. T. Martins, Nathan Schneider, and Noah A. Smith. 2014. Frame-semantic parsing. *Computational Linguistics*, 40:1:9–56.

Dan Flickinger. 2011. Accuracy vs. robustness in grammar engineering. In Emily M. Bender and Jennifer E. Arnold, editors, *Language from a Cognitive Perspective: Grammar, Usage and Processing*, number 201, pages 31–50. CSLI Publications, Stanford.

Christopher Kennedy. 2003. Ellipsis and syntactic representation. In Kerstin Schwabe and Susanne Winkler, editors, *The Interfaces: Deriving and Interpreting Omitted Structures*, number 61 in Linguistics Aktuell, pages 29–54. John Benjamins.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60.

Jason Merchant. 2013. Voice and ellipsis. *Linguistic Inquiry*, 44(1):77–108.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of ACL*, pages 115–124.

Slav Petrov, Leon Barrett, Romain Thibaux, and Dan Klein. 2006. Learning accurate, compact, and interpretable tree annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, ACL-44, pages 433–440, Stroudsburg, PA, USA. Association for Computational Linguistics.

Arnim Von Stechow. 1984. Comparing semantic theories of comparison. *Journal of Semantics*, 3(1):1–77.

Masaya Yoshida, Chizuru Nakao, and IvÃąn Ortega-Santos. 2016. *Ellipsis*. Routledge Handbook of Syntax, London, UK.

John M. Zelle and Raymond J. Mooney. 1996. Learning to parse database queries using inductive logic programming. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence - Volume 2*, AAAI'96, pages 1050–1055. AAAI Press.