Argument Mining for Understanding Peer Reviews (Supplementary Material)

Xinyu Hua, Mitko Nikolov, Nikhil Badugu, Lu Wang
Khoury College of Computer Sciences
Northeastern University
Boston, MA 02115
{hua.x, nikolov.m, badugu.n}@husky.neu.edu
luwang@ccs.neu.edu

A Annotation Details

Data Selection. We select 400 reviews from the ICLR 2018 dataset for the annotation study. To ensure the subset is representative of the full dataset, samples are drawn based on two aspects: review length and rating score.

Table 1 shows the distribution of reviews with regard to their length in the full ICLR 2018 dataset and the subset we sampled for annotation (AMPERE). As can be seen, the distribution over five bins are consistent between AMPERE and full dataset. A similar trend is observed on rating distribution in Table 2.

A subset of the reviews also have revision history, which can be used as a proxy for opinion change and review quality in future work. To that end, we manually set the ratio of revised reviews vs. unrevised ones to 3:1 (c.f. 9:1 on the full ICLR2018 dataset), to ensure that enough revised reviews are being annotated. Notice that, in this study, we only consider the initial version of a review if any revision exists.

Length	(0,200]	(200,400]	(400,600]	(600,800]	(800,∞)
AMPERE	14.8%	35.5%	25.3%	10.0%	14.6%
ICLR2018	17.6%	39.3%	23.8%	11.4%	7.9%

Table 1: Review length distribution of the full ICLR 2018 dataset and AMPERE, which consists of 400 sampled reviews.

Rating	**1**	**2**	**3**	**4**	**5**
AMPERE	3.0%	32.5%	43.8%	19.3%	1.5%
ICLR2018	2.6%	32.5%	42.4%	20.6%	1.8%

Table 2: Review rating distribution of AMPERE and the full ICLR 2018 dataset.

Inter-annotator Agreement (IAA). To measure IAA, we first follow Stab and Gurevych (2017) to calculate the unitized Krippendorf's α_U (Krippendorff, 2004) for each review, and report the average for each type.

We further consider agreement on the proposition level. However, since the segmented proposition boundaries by two annotators do not always match, we only consider the exact matched segments for Cohen's κ. The agreement scores for each type are listed in Table 3.

	EVAL	REQ	FACT	REF	QUOT	NON-A	overall
α_U	0.51	0.64	0.60	0.63	0.41	0.18	0.61
κ	0.60	0.68	0.64	0.88	0.59	0.27	0.64

Table 3: Inter-annotator agreement for all categories.

Sample Annotations. We show examples of annotated propositions in Table 4.

B Experiments

B.1 Data Preprocessing

For preprocessing, we tokenize and split reviews into sentences with the Stanford CoreNLP toolkit (Manning et al., 2014). We manually substitute special tokens for mathematical equations, URLs, and citations or references. In total, 302 variables (<VAR>), 125 equations (<EQN>), 62 URL links (<URL>), and 97 citations (<CIT>) are identified in 400 reviews.

B.2 Training Details

For all models except CNN, we conduct 5-fold cross validation on training set to select hyperparameters.

CRF. We utilize the CRFSuite (Okazaki, 2007) implementation and tune coefficients C_1 and C_2 for ℓ_1 and ℓ_2 regularizer. For segmentation task the optimal setup is $C_1 = 0.0$ and $C_2 = 1.0$; for joint prediction, $C_1 = 1.0$ and $C_2 = 0.01$ is used.

Evaluation	The paper shows nice results on a number of small tasks.
	With its poor exposition of the technique, it is difficult to recommend this paper for publication.
	I like the general approach of explicitly putting desired equivariance in the convolutional networks.
	The paper covers a very interesting topic and presents some though-provoking ideas.
	I'm not sure this strong language can be justified here.
Request	I would really like to see how the method performs without this hack.
	can the authors motivate this aspect better?
	I suggest using [hidelinks] for hyperref.
	More explanation needed here.
	In addtion -> In addition
Fact	Existing works on multi-task neural networks typically use hand-tuned weights for weighing losses across different tasks
	This work proposes a dynamic weight update scheme that updates weights for different task losses during training time by making use of the loss ratios of different tasks
	In this paper, the authors trains a large number of MNIST classifier networks with differing attributes (batch-size, activation function, no. layers etc.)
	This paper is based on the theory of group equivariant CNNs (G-CNNs), proposed by Cohen and Welling ICML'16.
Reference	[1] Burnetas, A. N., & Katehakis, M. N. (1997). Optimal adaptive policies for Markov decision processes. Mathematics of Operations Research, 22(1) , 222-255
	VARIANCE-BASED GRADIENT COMPRESSION FOR EFFICIENT DISTRIBUTED DEEP LEARNING
	see MuseGAN (Dong et al), MidiNet (Yang et al), etc
	e.g. Weakly-supervised Disentangling with Recurrent Transformations for 3D View Synthesis, Yang et al.
Quote	The author wrote "where r is lower bound of feature norm"
	"In a probabilistic context-free grammar (PCFG), all production rules are independent"
	Quoting from its abstract: "Using commodity hardware, our implementation achieves $\sim$ 90% scaling efficiency when moving from 8 to 256 GPUs."
Non-Arg	Did I miss something here?
	Below, I give some examples
	are all the test images resized before hand?
	How was this chosen?

Table 4: Sample annotated propositions.

BiLSTM-CRF. We experiment with implementation by Reimers and Gurevych (2017) with an extra ELMo embedding. Based on the cross validation for both segmentation and joint learning, the optimal network architecture selected has two layers with 100 dimensional hidden states each, with dropout probabilities of 0.5 for both layers. The word embedding pre-trained by Komninos and Manandhar (2016) is chosen, as it outperforms GloVe embeddings (Pennington et al., 2014) trained either on Google News or Wikipedia.

SVM. We utilize SAGA (Defazio et al., 2014) implemented in the Lightning library (Blondel and Pedregosa, 2016) to learn a linear SVM optimized with Coordinate Descent (Wright, 2015). The coefficient for a group Lasso regularizer (Yuan and Lin, 2006) is set to 0.001 by cross validation.

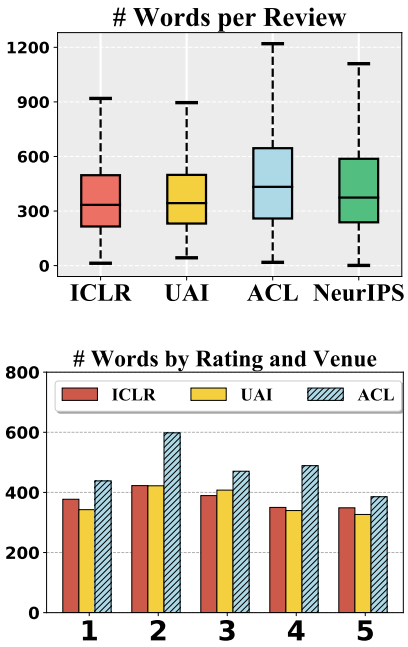

Figure 1: Word count in reviews by venue and rating. The word counts are significantly different between all venue pairs except UAI vs. ICLR and ACL vs. NeurIPS ($p < 10^{-6}$, unpaired t-test).

CNN. We implement the CNN-non-static variant as described in Kim (2014), with the following configuration: filter window sizes of {3,4,5}, with 128 feature maps each. Dropout probability is 0.5. 300 dimensional word embeddings are initiated with the pre-trained word2vec on 100 billion Google News (Mikolov et al., 2013).

C Further Analysis

Review Length by Venue and Rating. We compare review length of different venues in the top row of Figure 1. Unpaired t-test shows that ACL and NeurIPS have significantly longer reviews than UAI and ICLR ($p < 10^{-6}$), which is consistent with the trend for proposition counts, as described in Figure 2 in the paper.

We further group reviews by their ratings and display the average length per category in Figure 1. Again, we observe similar trends for the distribution of proposition count, where reviews with extreme ratings tend to be shorter.

Proposition Structure. We calculate the proposition type transition matrix as a proxy to uncover the local argumentative structure information. As is shown in Figure 2, propositions are more likely

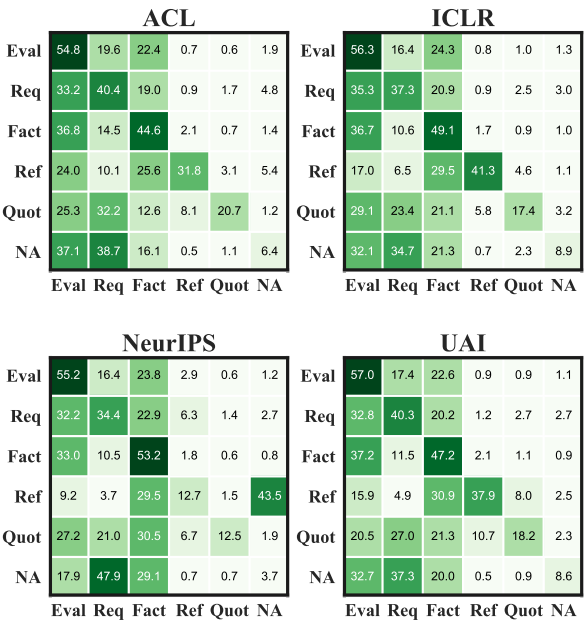

Figure 2: Proposition type transition matrix in different venues.

to be followed by propositions of the same type, while for NeurIPS the transition from reference to non-argument is much more prominent than other venues. A closer look at the dataset indicates that this might be because many formatted headers are mistakenly predicted as reference, e.g. "For detailed reviewing guidelines, see <URL>". They are usually followed by text such as "Comments to the author", which is predicted correctly as NON-ARG.

References

Mathieu Blondel and Fabian Pedregosa. 2016. Lightning: large-scale linear classification, regression and ranking in python.

Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. 2014. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in neural information processing systems*, pages 1646–1654.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751. Association for Computational Linguistics.

Alexandros Komninos and Suresh Manandhar. 2016. Dependency based embeddings for sentence classification tasks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1490–1500. Association for Computational Linguistics.

Klaus Krippendorff. 2004. Measuring the reliability of qualitative text analysis data. *Quality and Quantity*, 38:787–800.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Naoaki Okazaki. 2007. Crfsuite: a fast implementation of conditional random fields (crfs).

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2017. Reporting score distributions makes a difference: Performance study of lstm-networks for sequence tagging. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 338–348, Copenhagen, Denmark. Association for Computational Linguistics.

Christian Stab and Iryna Gurevych. 2017. Parsing argumentation structures in persuasive essays. *Computational Linguistics*, 43(3):619–659.

Stephen J Wright. 2015. Coordinate descent algorithms. *Mathematical Programming*, 151(1):3–34.

Ming Yuan and Yi Lin. 2006. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67.