Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210124

Argumentation Reasoning with Graph Neural Networks for Reddit Conversation Analysis

Teresa ALSINET, Josep ARGELICH, Ramón BÉJAR [1], Daniel GIBERT, Jordi PLANES and Nil TORRENT
INSPIRES Research Center – University of Lleida
Jaume II, 69 – 25001 Lleida, SPAIN

Abstract The automated analysis of different trends in online debating forums is an interesting tool for sampling the agreement between citizens in different topics. In these online debating forums, users post different comments and answers to previous comments of other users. In previous work, we have defined computational models to measure different values in these online debating forums. A main ingredient in these models has been the identification of the set of *winning posts* trough an argumentation problem that characterizes this winning set trough a particular argumentation acceptance semantics. In the argumentation problem we first associate the online debate to analyze as a debate tree. Then, comments are divided in two groups, the ones that agree with the root comment of the debate, and the ones that disagree with it, and we extract a bipartite graph where the unique edges are the disagree edges between comments of the two different groups. Once we compute the set of winning posts, we compute the different measures we are interested to get from the debate, as functions defined over the bipartite graph and the set of winning posts. In this work, we propose to explore the use of graph neural networks to solve the problem of computing these measures, using as input the debate tree, instead of our previous argumentation reasoning system that works with the bipartite graph. We focus on the particular online debate forum Reddit, and on the computation of a measure of the polarization in the debate. Our results over a set of Reddit debates, show that graph neural networks can be used with them to compute the polarization measure with an acceptable error, even if the number of layers of the network is bounded by a constant.

Keywords. Reddit, social networks analysis, argumentation, graph neural networks

1. Introduction

Recently, there has been a growing interest in the use of Graph Neural Network (GNN) approaches to model and solve reasoning problems defined via graph inputs [10,12,17]. The most common approach used by a GNN is to map the feature vector of each node

[1]Correspondence to: R. Béjar. INSPIRES Research Center, University of Lleida. C/Jaume II, 69. Lleida, Spain. Tel.: +34 973703477; E-mail: ramon.bejar@udl.cat.

to an embedding representation that also uses (by aggregation) the feature vector of its neighbor nodes. By iterating this scheme k times, the final representation of each node tends to capture structural information within the node's $k-$hop neighborhood. This scheme can be used to learn any kind of function over graphs that outputs a labeling of its nodes, or that outputs a single value (for graph classification tasks).

In previous work, we have considered the use of argumentation based models to analyze different characteristics of social network debates. In the argumentation based approach, we first identify a valued argumentation problem with the debate to be solved, where debate posts are associated with arguments, under a particular acceptance semantics: a set of rules that define what arguments are accepted and what are rejected. The usual acceptance semantics tend to be NP-hard, like the *ideal semantics* [8] we have used in our previous works about measuring discussion polarization with argumentation based models [2,3].

In this work we initiate a line of investigation to study whether a GNN approach can be a good candidate to solve argumentation-based problems with less effort. Our focus is not on exactly replicating the set of accepted arguments of the discussion, as it has been already explored on recent work about solving some abstract argumentation problems with GNNs [6,13], but on being able to compute the final measure of interest defined from the set of accepted arguments. Our hypothesis is that even if the worst-case complexity of computing accepted arguments is in general NP-hard, it may be possible to compute, or approximate, the final measure with much less computational effort. In particular, in this work we focus on the computation of a measure of discussion polarization that is defined in function of the set of accepted arguments of a discussion, and whether these arguments agree or disagree with the root topic of the discussion. Our discussions come from the social network Reddit. A Reddit debate is first represented as a debate tree, where edges represent agreement or disagreement relationships between Reddit posts. Then, this debate tree is processed to get a bipartite debate graph where posts are divided in two groups: the ones that agree with the root comment of the debate, and the ones that disagree with it. The edges of the bipartite graph represent disagreement between comments of the two groups.

Our results show that we can devise a reasoning system to compute that polarization measure, defined initially from the set of accepted arguments and the bipartite debate graph, based only on the original debate tree (the graph previous to the bipartite graph) and that computes the polarization measure with acceptable error, without explicitly computing the set of accepted arguments of the associated argumentation problem.

The structure of the paper is as follows. In Section 2 we present the relevant definitions for our argumentation-based Reddit analysis system. In Section 3 we briefly survey previous results about GNNs. In Section 4 we present the GNN architecture we have used to model our reasoning system. Finally, in Section 5 we present the experimental results we have obtained with a dataset of Reddit debates.

2. Reddit Debate Analysis

In this section we give the definitions of the different components of the Reddit analysis system introduced in [3]. It is based on two main components: a Reddit debate retrieval system and an argumentation-based reasoning system. The retrieval system takes a root

comment and obtains the complete set of comments generated in the debate on that root comment.

Definition 1 *A* comment c *is a tuple* $c = (m, u, sc)$*, where* m *is the text of the comment,* u *is the user's identifier of the comment, and* $sc \in \mathbb{Z}$ *is the score of the comment.*

Let $c_1 = (m_1, u_1, sc_1)$ *and* $c_2 = (m_2, u_2, sc_2)$ *be two comments. We say that* c_1 answers c_2 *if* c_1 *is a reply to comment* c_2*.*

Let $r = (m_r, u_r, sc_r)$ *be a comment such that* m_r *contains a link to some news. A* Reddit debate *on the (root) comment* r *is a non-empty set* Γ *of Reddit comments such that* $r \in \Gamma$ *and every comment* $c \in \Gamma$*,* $c \neq r$*,* c *answers some comment in* Γ[1]*.*

Next, we obtain the tree representation of a Reddit debate where we incorporate edge labels that express the sentiment of the comments.

Definition 2 *Let* Γ *be a* Reddit debate *on a (root) comment* r*. The* Debate Tree (DebT) *for* Γ *is a tuple* $\mathcal{T} = \langle C, r, E, L \rangle$ *such that:*

- *for every comment in* Γ *there is a node in* C*,*
- *node* $r \in C$ *is the root node of* $\mathcal{T}$*,*
- *if* c_1 *answers* c_2 *then there is a directed edge* (c_1, c_2) *in* E*, and*
- L *is a labeling function* $L : E \to [-2, 2]$*, where the value assigned to an edge denotes the sentiment of the answer, from highly negative (-2) to highly positive (2).*

Only the nodes and edges obtained by applying this process belong to C *and* E*, respectively.*

As argued in [3], we consider in our model that subtrees with a neutral root do not contribute anything relevant with respect defending or rejecting the root comment of the debate. So, the next step is to prune out those subtrees with respect to a pruning threshold.

Definition 3 *Let* α *be a pruning threshold in the real interval* $[0, 2]$ *and let* $\mathcal{T} = \langle C, r, E, L \rangle$ *be a* DebT*. The* Pruned Debate Tree (PDebT) *for* $\mathcal{T}$ *with respect to* α *is a tuple* $\mathcal{T}_\alpha = \langle C_\alpha, r, E_\alpha, L \rangle$*, where both sets of pruned comments* $C_\alpha \subseteq C$ *and pruned edges* $E_\alpha \subseteq E$ *are defined as follows:*

- *the root node (comment)* $r \in C_\alpha$*,*
- r *is the root node of* $\mathcal{T}_\alpha$ *and*
- *if* $(c_1, c_2) \in E$ *with* $c_2 \in C_\alpha$*, then* $c_1 \in C_\alpha$ *and* $(c_1, c_2) \in E_\alpha$*, whenever* $|L(c_1, c_2)| \geq \alpha$*.*

Only the nodes and edges obtained by applying this process belong to C_α *and* E_α*, respectively.*

Note that for $\alpha = 0$ the pruning threshold has no effect, in the sense that the PDebT obtained corresponds to the original DebT and that for $\alpha = 2$ the PDebT obtained only contains strictly polarized both positive and negative answers. In any case, the PDebT $\mathcal{T}_\alpha$ is a subtree of $\mathcal{T}$ with r being the root node.

[1]Given the structure of a Reddit debate, except for the root comment, each comment answers exactly one previous comment, usually by another user or author.

Finally, we divide the set of comments into two sets: comments supporting the root comment and comments that disagree with it. Then, the attacks between the comments of both sets are defined as a subset of edges in E_α such that they are negative answers from a comment in one of the sets to a comment in the other set, obtaining a bipartite graph that represents both sides of the debate, and the disagreement between them. This bipartition can be computed with the algorithm that we presented in [2]. Moreover, we also label each node of the graph obtained with a weight that denotes the comments' social acceptance during the debate. Next we formalize the Weighted Bipartite Debate Graph structure.

Definition 4 *Let $\mathcal{T}_\alpha = \langle C_\alpha, r, E_\alpha, L\rangle$ be a* PDebT *for a Reddit debate Γ. A* Weighted Bipartite Debate Graph (WBDebG) *for $\mathcal{T}_\alpha$ is a tuple $G = \langle C_+ \cup C_-, E_-, W\rangle$ where*

- *C_+ and C_- is a bipartition of C_α. Thus, $C_+ \cup C_- = C_\alpha$ and $C_+ \cap C_- = \emptyset$, where C_+ denotes the set of comments that agree with the root comment c_r, and C_- denotes the set of comments that disagree with it.*
- *$E_- = \{(c_1, c_2) \in E_\alpha \mid L(c_1, c_2) < 0\}$ and corresponds with the set of disagreement edges between the comments in C_+ and C_-. Thus, if $(c_1, c_2) \in E_-$, then either $c_1 \in C_+$ and $c_2 \in C_-$ or, $c_1 \in C_-$ and $c_2 \in C_+$.*
- *W is a weighting scheme $W : C_\alpha \to \mathbb{N}$ of the weight of nodes (comments). The weighting scheme W evaluates the social acceptance of comments by mapping the score sc of a comment $(m, u, sc) \in C_\alpha$ to a value in $\mathbb{N}$.*

At this point we are ready to introduce the argumentation-based reasoning system used to obtain the set of comments, from the two opposite groups of a WBDebG, that are accepted in the sense that this set should represent a kind of consensus among all the comments of the debate. To this end, we use value-based abstract argumentation [5] to model the weighted argumentation problem associated with a WBDebG and ideal semantics [7] to compute its solution (the set of comments that can be accepted).

The *value-based abstract argumentation framework* (VAF) we define for a WBDebG $G = \langle C_+ \cup C_-, E_-, W\rangle$, interprets each comment in $C_+ \cup C_-$ as an argument and defines a *defeat* relation (or effective attack relation) between arguments as follows:

$$\textit{defeats} = \{(c_1, c_2) \in E_- \mid W(c_2)) \not\geq W(c_1)\};$$

i.e. argument c_1 *defeats* argument c_2 if and only if c_1 attacks or disagrees with c_2 and the social acceptance value of c_2 is not preferred over the social acceptance value of c_1, based on the weighting scheme W.

Then, a set of comments $S \subseteq C_+ \cup C_-$ is called *conflict-free* if for all $c_1, c_2 \in S, (c_1, c_2) \notin$ *defeats*, and a conflict-free set of comments $S \subseteq C_+ \cup C_-$ is defined as *maximally admissible* if for all $c_1 \notin S$, $S \cup \{c_1\}$ is not conflict-free and, for all $c_2 \in S$, if $(c_1, c_2) \in$ *defeats* then there exists $c_3 \in S$ such that $(c_3, c_1) \in$ *defeats*. Finally, the *solution* or *set of accepted comments* for a debate is the largest admissible conflict-free set of comments $S \subseteq C_+ \cup C_-$ in the intersection of all maximally admissible conflict-free sets.

We select this semantics to define the solution for a debate, because it represents a maximally admissible set of conflict-free comments, such that they defend against attacks outside the set with comments inside the set, and they are included in any admis-

sible set of comments. This set therefore represents a kind of *maximum consensus* between all the possible admissible sets of comments. For our particular case of an acyclic VAF, the picture is even simpler, as there is a unique maximally admissible set, and thus the solution for ideal semantics coincides with this set. Moreover, for the case of a VAF that is acyclic or bipartite (as in the case of a WBDebG), we can compute its solution in linear time, with respect to the number of comments, for instances of big size with the distributed algorithm we developed in [1]. However, in the worst case the status of each comment in the solution may depend on the status of the rest of the comments, so that is why we explore in this work a possible GNN-based architecture where nodes (comments) only consider information from nodes at distance bounded by a constant.

Given that the solution for the debate provides us with a consensus point of view, an interesting characteristic to analyze is its degree of polarization.

Definition 5 *Let* $G = \langle C_+ \cup C_-, E_-, W \rangle$ *be a* WBDebG *and let* $S \subseteq C_+ \cup C_-$ *be the solution for* G*. The* polarization degree *of solution* S *is a measure in the real interval* $[-1, 1]$ *defined as follows:*

$$\textit{polarization}(S) = \frac{\#(S \cap C_+) - \#(S \cap C_-)}{\#S}.$$

We use the polarization degree value as a measure of the bias of the solution S towards comments in C_+ and comments in C_-. The value that indicates total bipartisanship (0) is obtained when the number of comments of S in C_+ equals the number in C_-. The highest positive value is obtained when all the comments of the solution are found in C_+, and analogously for the lowest negative value. In order to classify debates in terms of the polarization degree, instead of this measure, we can also work with a more qualitative measure mapping from it, to a discrete set of values. For example, in this work we stratify Reddit debates in five levels, based on the polarization degree of the solution:

$$\textit{bias-level} : \textit{polarization}(S) \rightarrow \{-2, -1, 0, 1, 2\}.$$

3. Graph Neural Networks

In the last years, there has been an increasing interest in analyzing graphs with machine learning (ML) [9,16] because of the immense expressive power of graphs, i.e. graphs can be used to model the interaction between complex structures such as proteins, mRNA, particles in physics models, etcetera. Thus, a key factor to be considered when dealing with graphs using ML is the ability of the methods to deal with graphs of different sizes and shapes.

There have been various attempts in the literature using graph neural networks (GNNs), mainly by: (i) focusing on learning node embeddings by aggregating the nodes, and (ii) by mapping from the node neighbourhood domain (adjancency matrix) to spectral domain. From the first type, we highlight the Generalizing Aggregation GraphSage [10] used for node classification. This method focuses on learning node embeddings, and then a model aggregates the resulting embeddings to handle size-varying neighbourhoods. From the second type, we feature the Spectral Graph Convolution Model [12] used for the classification of nodes using their adjacency matrix. In addi-

tion, it uses Chebyshev filters (passband filters) and Lapacian regularization in the loss function.

A recent improvement over the first method are Graph Isomorphism Networks (GIN), presented in a study [17] of GNN expressivity w.r.t. Wesfeiler-Lehman (WL) test [15] of graph isomorphism, where they proposed a WL equivalent aggregator, i.e. it generalizes the WL test and thus, it achieves the maximum discriminative power among the GNNs in the literature.

4. GNN Modelling

We propose the use of Graph Isomorphism Networks (GIN) [17] in our GNN model to approximately compute the polarization degree of a Reddit debate. In particular, our GIN model receives as input a Pruned Debate Tree (PDebT) $\mathcal{T}_\alpha = \langle C_\alpha, r, E_\alpha, L \rangle$ with $|C_\alpha| = N$ nodes, obtained from a Reddit debate as explained in [2], and outputs a bias-level of the polarization degree.

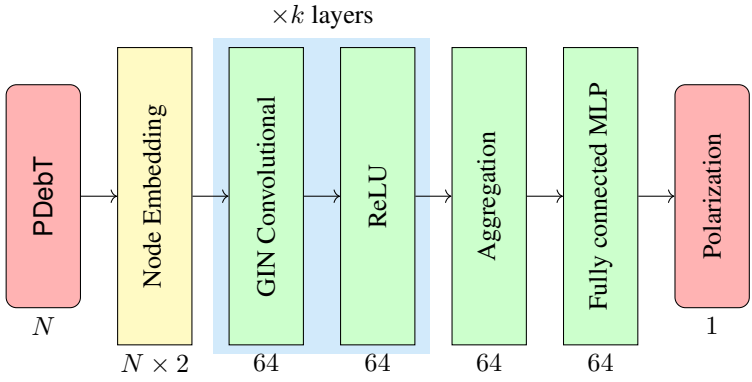

Figure 1. GNN architecture for computing the polarity degree of a Pruned Debate Tree.

The overall architecture is presented in Figure 1. It comprises the following layers:

Node embedding The input layer contains a two dimensional vector for each non-root comment $c_i = (m_i, u_i, sc_i)$ that contains the score of the comment sc_i and the sentiment from the label $L(c_i, c_j)$, where c_j is the unique comment such that $(c_i, c_j) \in E_\alpha$.

GIN Convolutional (k layers). Every layer combines the node embedding of the previous layer considering the node close neighbours. The aggregator in the layer l is the following:

$$\mathbf{x_i}^{(l)} = MLP\left((1+\epsilon)\cdot \mathbf{x_i}^{(l-1)} + \sum_{j \in \mathcal{N}(i)} \mathbf{x_j}^{(l-1)}\right)$$

where $\mathbf{x_i}^{(l)}$ is the embedding of node i in layer l, ϵ is a learnable parameter, and MLP is a multi-layer perceptron with nonlinearity, and $\mathcal{N}(i)$ is the set of neigh-

bours of node i. The first GIN layer has an input dimension of 2 and an output dimension of 64. The following layers have input and output dimensions of 64. Globally, this GIN block maps the two-dimensional vector of each node to a vector of 64 values that tries to capture the information from nodes k hops away from it. Also, we insert a Rectified Linear Unit (ReLU) layer after each GIN layer, to help encode non-linear outputs in the network.

Normalization We give also the option to include a normalization layer between consecutive GIN layers, because previous work suggests that it may speed up the learning process [4].

Aggregation The aggregation layer creates the final graph representation using the mean operator, aggregating all the node embeddings into one graph embedding, as a vector with the same dimension (64).

Fully connected MLP This block maps the final aggregated embedding representation of the graph into the polarization bias-level of the debate.

After every ReLU layer and at the end of the fully connected MLP, a dropout of 0.25 is applied to prevent overfitting [11]. We use the pytorch and pytorch geometric python libraries to implement this GNN model.

5. Experimental results

In this section we present the results obtained when learning a GNN model with the GIN architecture introduced in Section 4 to compute the polarization bias-level for a set of Reddit debates.

To train and test our models, we use a dataset with 40 Reddit debates, where 34 have been used for training and 6 have been used for testing. To download the set of comments for each Reddit debate we use the Python Reddit API Wrapper (PRAW) [2]. Then, in the PDebT $\mathcal{T}_\alpha$ obtained from each Reddit debate, the label for each edge (c_1, c_2) is computed with the sentiment analysis software of [14]. It uses the text of the comment c_1, where the value assigned denotes the sentiment of the answer, from highly negative (-2) to highly positive (2). The pruning parameter α is set to the value 0.15. We have tried three different values for the number of GIN layers $(2, 4, 6)$ and also experimented with either using a normalization layer after each GIN layer or not. The number of GIN layers is kept low, compared with the number of nodes of the graphs that ranges from 5 to 4472 nodes, to explore whether bounding the neighborhood size used by the GNN still allows a reasonable approximation of the right output value. As we prefer a GNN model where the output value is as closer as possible to the right polarization bias-level, we train our GNN models using as the loss function the mean square error.

The experimental results for the average loss for the training set and the average loss for the test set are shown in Table 1, where each experiment was repeated with two different number of epochs $(250, 500)$ and executed 10 times (generating each time a different training/test set). The results shown in the table for each experimental setting are the best ones (with respect to the test set loss) from the set of 10 executions. The results show that the training loss is slightly higher than in the test set, suggesting that our models seem to not overfit with the training set. The results obtained with different

[2] `https://github.com/praw-dev/praw`

number of GIN layers do not set seem to have a significant impact on the test set loss when considering 500 epochs for learning. Analogously, the use of normalization layers between GIN layers do not seem to have a significant impact, as with no normalization the results are slightly better.

		Training Loss		Test Loss	
Num GIN layers	Normalization	Epochs 250	Epochs 500	Epochs 250	Epochs 500
2	True	0.32	0.35	0.37	0.18
4	True	0.42	0.33	0.15	0.28
6	True	0.47	0.33	0.24	0.23
2	False	0.45	0.47	0.27	0.04
4	False	0.50	0.27	0.30	0.11
6	False	0.34	0.44	0.18	0.12

Table 1. Experimental results for polarization computation with our GNN model.

To check whether our GNN model generalizes well when the number of nodes of the input increases, we have repeated the previous experiment, but only with no normalization and number of epochs 500, fixing as the training set the graphs with the smallest size (from 5 to 414 nodes in our case) and as the test set the biggest ones (from 1029 to 4472 nodes). The results obtained show that we have a slight increase in the test set average loss: 0.26 for 2 GIN layers, 0.27 for 4 and 0.23 for 6. So, at least with this test set, training with the smallest ones does not seem to increase significantly the test set loss, although these results should be further confirmed with larger training and test sets.

6. Conclusions

In this paper, we have presented a GNN-based system to solve the problem of computing a polarization measure from a Reddit debate. Our GNN-based system is based on our previous work, where we used an argumentation approach to solve this problem. Although we do not use the GNN architecture to explicitly compute the solution of the argumentation problem, it is able to approximate the final polarization measure, that it is originally defined from that solution. This happens even if our GNN model aggregates information in each node considering always a neighborhood with bounded distance, given that the number of GIN layers is kept constant.

An interesting direction for future work is to consider the computation of other argumentation-based measures that consider as input author graphs, instead of debate trees. Author graphs come from the aggregation of comments from the same author in a single node, such that the resulting graph may contain cycles, and in that case the complexity of the argumentation-based reasoning algorithm is higher than the one for the acyclic graphs we have considered in this work. Also, we plan to work with a bigger Reddit dataset to get more significant results.

Acknowledgments This work was partially funded by Spanish Project PID2019-111544GB-C22, by the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement 723596 and Grant Agreement 768824, and by 2017 SGR 1537.

References

[1] T. Alsinet, J. Argelich, R. Béjar, and J. Cemeli. A distributed argumentation algorithm for mining consistent opinions in weighted twitter discussions. *Soft Comput.*, 23(7):2147–2166, 2019.

[2] T. Alsinet, J. Argelich, R. Béjar, and S. Martínez. An argumentation approach for agreement analysis in reddit debates. In *Artificial Intelligence Research and Development - Current Challenges, New Trends and Applications, CCIA 2018, 21st International Conference of the Catalan Association for Artificial Intelligence, Alt Empordà, Catalonia, Spain, 8-10th October 2018*, pages 217–226, 2018.

[3] T. Alsinet, J. Argelich, R. Béjar, and S. Martínez. Measuring user relevance in online debates through an argumentative model. *Pattern Recognit. Lett.*, 133:41–47, 2020.

[4] L. J. Ba, J. R. Kiros, and G. E. Hinton. Layer normalization. *CoRR*, abs/1607.06450, 2016.

[5] T. J. M. Bench-Capon. Persuasion in practical argument using value-based argumentation frameworks. *Journal of Logic and Computation*, 13(3):429–448, 2003.

[6] D. Craandijk and F. Bex. Deep learning for abstract argumentation semantics. In C. Bessiere, editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 1667–1673. International Joint Conferences on Artificial Intelligence Organization, 7 2020. Main track.

[7] P. M. Dung, P. Mancarella, and F. Toni. Computing ideal sceptical argumentation. *Artif. Intell.*, 171(10-15):642–674, 2007.

[8] P. E. Dunne. The computational complexity of ideal semantics. *Artif. Intell.*, 173(18):1559–1591, 2009.

[9] F. Errica, M. Podda, D. Bacciu, and A. Micheli. A fair comparison of graph neural networks for graph classification. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[10] W. L. Hamilton, Z. Ying, and J. Leskovec. Inductive representation learning on large graphs. In I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 1024–1034, 2017.

[11] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *CoRR*, abs/1207.0580, 2012.

[12] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[13] I. Kuhlmann and M. Thimm. Using graph convolutional networks for approximate reasoning with abstract argumentation frameworks: A feasibility study. In N. Ben Amor, B. Quost, and M. Theobald, editors, *Scalable Uncertainty Management*, pages 24–37. Springer International Publishing, 2019.

[14] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60, 2014.

[15] B. Weisfeiler and A. Leman. A reduction of a graph to a canonical form and an algebra arising during this reduction. *Nauchno-Technicheskaya Informatisiz*, 2(9):12–16, 1968.

[16] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu. A comprehensive survey on graph neural networks. *IEEE Trans. Neural Networks Learn. Syst.*, 32(1):4–24, 2021.

[17] K. Xu, W. Hu, J. Leskovec, and S. Jegelka. How powerful are graph neural networks? In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.