SUTNLP at SemEval-2023 Task 4: LG-Transformer for Human Value Detection

Hamed Hematian Hemati, Sayed Hesam Alavian, Hossein Sameti, Hamid Beigy
AI Group, Computer Engineering Department, Sharif University of Technology, Tehran, Iran
{ hematian.hemati.hamed, hesam.alavian70}@gmail.com, {sameti, beigy}@sharif.edu

Abstract

When we interact with other humans, human values guide us to consider the human element. As we shall see, value analysis in NLP has been applied to personality profiling but not to argument mining. As part of SemEval-2023 Shared Task 4, our system paper describes a multi-label classifier for identifying human values. Human value detection requires multi-label classification since each argument may contain multiple values. In this paper, we propose an architecture called **L**abel **G**raph **Transformer** (LG-Transformer). LG-Transformer is a two-stage pipeline consisting of a transformer jointly encoding argument and labels and a graph module encoding and obtaining further interactions between labels. Using adversarial training, we can boost performance even further. Our best method scored $50.00\,\%$ using F1 score on the test set, which is $7.8\,\%$ higher than the best baseline method. Our code is publicly available on Github.[1]

1 Introduction

Most, if not all, social sciences stress the importance of human values (Rokeach, 1973) and have integrated them into computational frameworks of argumentation (Bench-Capon, 2003). The analysis of values in NLP has been applied to personality profiling but hasn't been applied to argument mining (Maheshwari et al., 2017). Values are commonly accepted answers to why some option is desirable in the ethical sense and are thus essential both in real-world argumentation and theoretical argumentation frameworks (Kiesel et al., 2022). For Identifying Human Values behind Arguments, Touché23-ValueEval collected 9,324 arguments from 6 diverse sources, including religious texts, political discussions, free-text arguments, newspaper editorials, and online democracy platforms (Mirzakhmedova et al., 2023b).

[1] https://github.com/SUTNLP/LG_Transformer

The purpose of this system paper is to present SUTNLP's (David Gauthier on leaderboard) work on the SemEval-2023 Shared Task 4 which is focused on developing a classifier to classify human values (Kiesel et al., 2023). Detecting human values requires multi-label classification since each argument may contain several values. To tackle this problem, we propose a two-stage pipeline consisting of (1) a transformer (Vaswani et al., 2017) which jointly encodes the argument and labels and (2) a graph module (Wu et al., 2021) which further encodes and gets interactions between labels. Lastly, we employ adversarial training (Bai et al., 2021) to further enhance the model's performance.

The best performing method achieves F1 score of $50.00\,\%$ on the test set, outperforming the best baseline method by $7.8\,\%$. Using our method, we ranked 8th out of 40 competing teams. Results are submitted through TIRA (Fröbe et al., 2023).

2 Background

In human value detection task, a textual argument and multiple human value categories are given and the goal is to classify which categories, the argument draws on. This task uses a set of 20 value categories compiled from the social science and described in Identifying the Human Values behind Arguments paper (Kiesel et al., 2022). The value categories are shown in Figure 1.

The main focus of Human Value Detection has been on developing multi-label Text classification (Kiesel et al., 2022; Mirzakhmedova et al., 2023b). Text classification is a fundamental task in Natural Language Processing (NLP), and multi-label text classification (MLTC) is a key branch of it. MLTC has undergone a transformation from traditional machine learning to deep learning, and various models with excellent performance have emerged one after another (Chen et al., 2022).

Recently researches have shown that pre-trained language models like transformers approach have

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 340–346

Figure 1: The employed value taxonomy of 20 value categories (Mirzakhmedova et al., 2023b)

proven to be effective in multi-class text classification (Fallah et al., 2022). The BERT model (Devlin et al., 2018) has emerged as a popular state-of-the-art model in recent years. It is able to cope with NLP tasks such as multi-label text classification (Cai et al., 2020).

Transformer models have become the most effective neural network architecture for neural language modeling. A novel model architecture called DeBERTa (Decoding-enhanced BERT with disentangled attention) improves the BERT model using two novel techniques. The first is the disentangled attention mechanism, where each token is represented using two vectors that encode its content and position respectively. The attention weights among tokens are computed using disentangled matrices on their contents and relative positions respectively. Second, an enhanced mask decoder is used to incorporate absolute positions in the decoding layer to predict the masked tokens in model pre-training (He et al., 2020).

A sample of text can be assigned to more than one classification in Multi-Label Text Classification (MLTC). Since in MLTC tasks, there are dependencies or correlations among labels, in recent years several models have been proposed to captures the dependencies between labels. In a recent work, a graph attention network-based model is proposed to handle the attentive dependency structure among the labels. The graph attention network uses a feature matrix and a correlation matrix to model and explore the crucial dependencies between the labels and generate classifiers (Pal et al., 2020). Ma et al. (2021) proposes a label-specific dual graph neural network, which incorporates labels information to learn label-specific components from documents, and employs dual graph convolution network to model interactions among these components based on co-occurrence and dynamic reconstruction graph in a joint way. Zhang et al. (2021) jointly encodes text and labels to exploit labels' semantics, to model correlation between labels. Aside from classifying labels, they further define additional targets. Cai et al. (2020) also uses labels' semantics and they capture labels' semnatic in two stages. a label graph construction approach is proposed to capture the label correlations and a neoteric attention mechanism to establish the semantic connections between labels and words and to obtain the label-specific word representation.

In designing a machine learning method, generalization and robustness are both critical requirements. Adversarial training is a means to enhance robustness and generalization (Goodfellow et al., 2014). In natural language processing (NLP), pre-training large neural language models such as BERT have demonstrated impressive gain in generalization for a variety of tasks, with further improvement from adversarial fine-tuning (Liu et al., 2020). In recent years, adversarial training has gained popularity in the field of natural language processing (NLP). Miyato et al. (2016) applies adversarial training to improve the performance of semi-supervised text classification models. The method has been applied to a variety of NLP applications, including sentiment analysis, spam detection, and topic modeling (Iyyer et al., 2018; Wu et al., 2017; Zhu et al., 2021). Further, performance on several benchmarks in NLP have been improved by deploying adversarial training (Zhu et al., 2019).

3 System Overview

3.1 Model Architecture

3.1.1 Transformer

We adopt a pre-trained language model as the backbone of our model. It has been demonstrated that modeling the interaction between input text and labels would be effective. Following Zhang et al. (2021); Cai et al. (2020) to fully utilize the power of pre-trained transformers to use labels' semantics and achieve premise-aware label representations, we concatenate the premise with label names and tokenize the whole input and feed it to the transformer. The transformers outputs a hidden vector

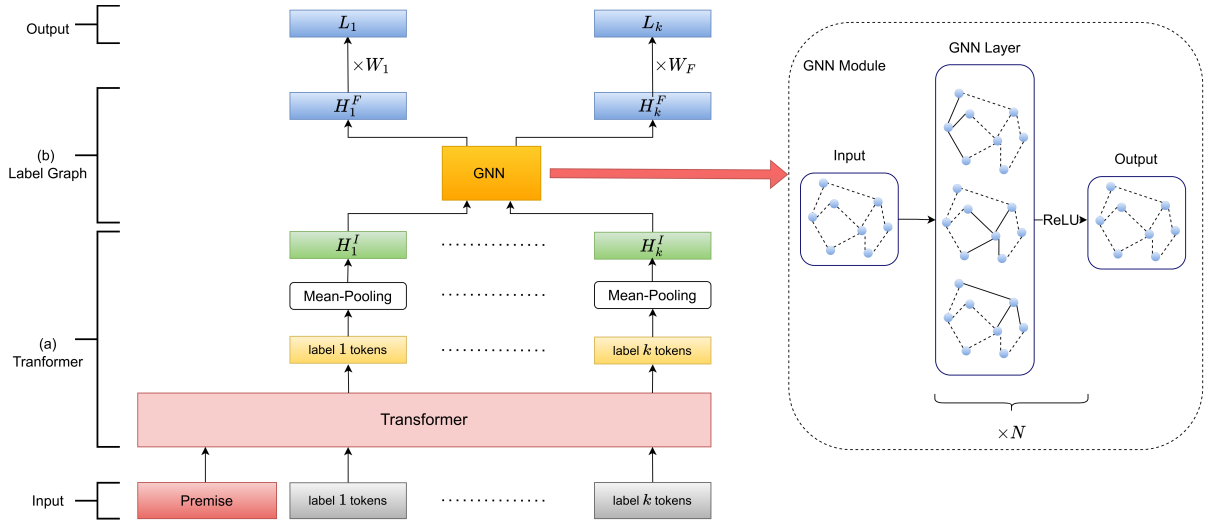

Figure 2: Architecture of the proposed model. (a) is the transformer part, (b) represents graph module, N is the hidden layers count in graph module, k is the number of labels which is 20 for the dataset. Output consists of the final logits. GNN represents Graph Neural Network.

for each token. To obtain a representation of each label, we average its tokens since each label could have been divided into multiple tokens. Suppose t_{l_s} is the first token of label l and t_{l_e} is the last token of label l. H_l^I is the intermediate representation of label l.

$$H_l = MeanPool(t_{l_s}, \cdots, t_{l_e}) \tag{1}$$

3.1.2 Label Graph

H_l^I contains information about the current label, as well as other labels and the input premise because using the transformer, the premise is jointly modeled with labels. In our experiments, we find out that it is also beneficial to further let labels interact and share their specific information to better model their correlation. Since there could be some correlation between labels (e.g. some labels might always come together) and this correlation is not leveraged in the transformer part, the label graph module is added to the architecture to do so. In order to do this and capture label correlation, we make a graph of labels. Each label represents a node in the graph and two labels are connected if there is at least one argument in the train data which contains both values (labels) at the same time. Then we apply an N-layer GCN (Kipf and Welling, 2016) to the constructed graph to further let labels share information among themselves. Input features of node l are H_l^I. After applying GCN to the graph, we obtain the final representation of each label. H_l^F is the final representation of label l.

3.1.3 Prediction

After obtaining H_l^F, a trainable weight matrix W_l is multiplied with H_l^F to obtain the final logit for label l. L_l is the final logit for label l which is used for training and evaluation.

$$L_l = H_l^F W_l \tag{2}$$

3.2 Loss Reweighting

Since each label has an imbalanced number of 1s and 0s, we further weight the final loss for each label according to the inverse ratio of its 1s and 0s counts in the training data.

3.3 Adversarial Training

In recent years, several studies in NLP have employed Adversarial Training (AT) to boost the generalization of their model (Zhu et al., 2021; Liu et al., 2020; Wang et al., 2019; Zhu et al., 2019; Cui et al., 2022; Lu et al., 2022). Following these studies, we employ adversarial training to further boost the generalization and ultimately the performance of the model. Adversarial training method tries to find the optimal parameters θ^* minimizing the maximum possible adversarial perturbation δ to the outputs of a random layer, inside a norm ball of ϵ which can be stated as follows:

$$\theta^* = \arg\min_{\theta} \mathbb{E}_{\mathcal{D}} \left\{ \max_{\delta:\|\delta\|\leq\epsilon} L(f_\theta(X+\delta), Y) \right\} \tag{3}$$

In Equation 3, θ denotes the network parameters, $\mathcal{D}$ is the dataset distribution and X is the random layer output which adversarial training is applied to. Y represents the label. f_θ is the network and L represents the loss function. K-projected gradient descent (K-PGD) (Madry et al., 2017) is adopted to train the network using adversarial training. K is a hyperparameter which is tuned.

4 Experimental Setup

4.1 Dataset and Evaluation

There are 9324 arguments in the SemEval-2023 Task 4 dataset on a variety of statements in different styles, including religious texts (Nahj al-Balagha), newspaper articles (The New York Times), political discussions (Group Discussion Ideas), free-text arguments (IBM-ArgQ-Rank-30kArgs), community discussions (Zhihu), and democratic discourse (Conference on the Future of Europe) (Mirzakhmedova et al., 2023a). According to the organizers, dataset is divided into (5220, 1896, 1576) which represents the number of (train, dev, test) arguments respectively. Each argument consists of one premise, one conclusion, and a stance attribute indicating whether the premise is in favor or against the conclusion (Mirzakhmedova et al., 2023a). There are 20 labels and each argument contains a 0 or 1 for each label (value) which 1 means that argument contains a value and 0 means it does not contain the value. To estimate the performance of the system, the organizers employ mean of F1 scores over all labels. Following the organizers of the task, we use the mean of F1 score over all labels as the main evaluation metric of our models. This metric is also used by organizers to sort the competitor teams' performances for the competition. In addition, we report the mean score of precision and recall over all labels. All the reported results are from the test data.

4.2 Parameter Settings

We use trial and error to optimize the hyperparameters and here we report the hyperparameters used to train our task for the best performing model on dev set. We train model for 30 epochs and after training, the best performing model checkpoint on dev set is used for prediction. Initial learning rate is set to $2e-5$ and is decreased to zero using polynomial decay scheduler. Batch size is set to 2 and weight decay is set to $1e-3$. N is set to 3 and we adopt a graph convolutional network with 3 layers, each with hidden size of 400. After each layer a dropout layer with a drop rate of 0.3 and a ReLU activation is used. cross-entropy is used for the loss. AdamW (Loshchilov and Hutter, 2017) algorithm is adopted as the optimization algorithm. K is set to 1 for PGD.

The implementation is fully based on the torch framework. We use huggingface library[2] to implement transformers and torch-geometric library[3] to implement graph neural network.

5 Results

5.1 Main Results

All experiments Results are shown in Table 1. Following Mirzakhmedova et al. (2023a), we report two baseline methods. First one is a 1-Baseline and second one is BERT Baseline. LG-BERT + AT indicates LG-Transformer method with BERT as backbone transformer and AT indicates adversarial training. As can be seen LG-BERT + AT boosts the performance significantly while LG-BERT + AT only contains 0.005 % more parameters than BERT. DeBERTa is a method in which text is passed to DeBERTa as input and use $[CLS]$ token for prediction of each label. DeBERTa + labels indicates a method in which labels are present as input but label graph is removed and to get logits for this method linear layers are applied to intermediate representations. And at last the method which is used for ranking in competition is LG-DeBERTa + AT which indicates LG-Transformer method with DeBERTa as backbone transformer and adversarial training. DeBERTa signifies DeBERTa-Base model which is shortened for simplicity.

Method	F1	Precision	Recall
1-Baseline	26.3	15.10	100.0
BERT Baseline	42.20	58.70	32.90
LG-BERT + AT	45.03	45.78	44.03
DeBERTa	48.98	51.05	47.07
DeBERTa + labels	46.30	47.45	45.21
LG-DeBERTa	49.34	48.13	50.61
LG-DeBERTa + AT	50.00	50.27	49.70

Table 1: Main Results

[2] https://huggingface.co/

[3] https://pytorch-geometric.readthedocs.io/en/latest/

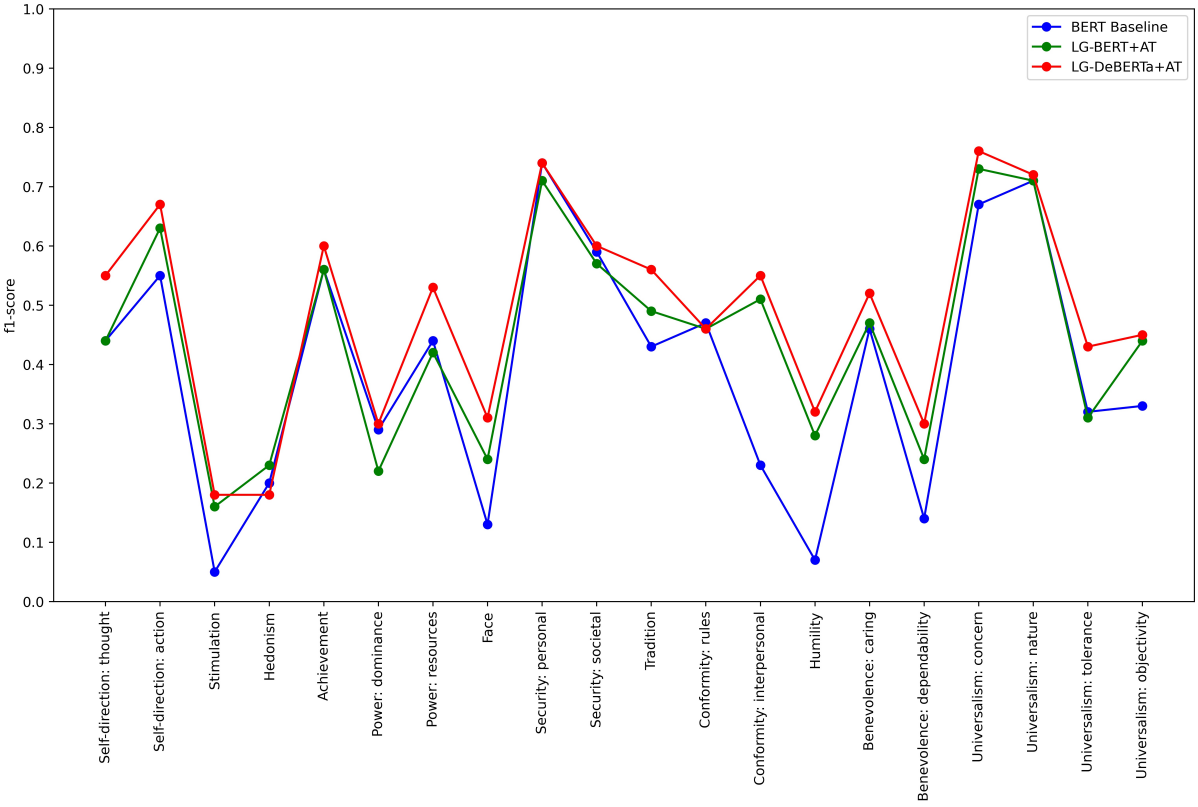

Figure 3: Comparison on our methods with BERT baseline over different categories

5.2 Effect of label graph

This section evaluates the label graph module's impact on the pipeline. We perform the evaluation using methods which use DeBERTa as their backbone transformer since it is obvious that methods with DeBERTa as their backbone transformer perform quite better than those which use BERT as their backbone transformer. In Table 1, we observe that LG-DeBERTA has $+0.36\,\%$ increase in F1 score compared to DeBERTa. In order to understand whether this gain is due to the label graph or not, an experiment is conducted using labels with text input. The fifth row of Table 1 shows the results of this method. As it is clear adding labels without using the label graph induces a huge drop ($-2.68\,\%$) in the performance of DeBERTa. This indicates that simply adding labels to the transformer input could result in some noise which degrades the model performance and further modeling between labels is required. Additionally since the label graph uses GCN layers, it might induce a regularization effect on the model and hence boost the performance. In Figure 3, a comparison of F1 scores of LG-Transformer methods with the BERT-Baseline over different labels is depicted. For most of the labels our method variants perform quite better than the baseline BERT. For some labels like "Hedonism", "Conformity:Rules", and "Universalism:nature" our best model performs near or worse than the BERT-Baseline.

6 Conclusion

The paper presents SUTNLP's submission to SemEval-2023 Task 4 "ValueEval: Identification of Human Values behind Arguments" competition. To solve this problem, we use transformers to jointly encode premise and labels and exploit the semantic correlation between label names and between label names and the premise. Further we propose a label graph to enhance labels interaction and further process label representations. Additionally we employ adversarial training to further boost the performance of the model. Through experiments, we show that using label graph is imperative for the task when using label semantics. Mean of F1 scores over labels is used to compare the performance of the models. By this criteria our proposed method proves to be effective and our best model outperforms the BERT baseline by $7.8\,\%$.

References

Tao Bai, Jinqi Luo, Jun Zhao, Bihan Wen, and Qian Wang. 2021. Recent advances in adversarial training for adversarial robustness. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 4312–4321. International Joint Conferences on Artificial Intelligence Organization. Survey Track.

Trevor J. M. Bench-Capon. 2003. Persuasion in Practical Argument Using Value-based Argumentation Frameworks. *Journal of Logic and Computation*, 13(3):429–448.

Linkun Cai, Yu Song, Tao Liu, and Kunli Zhang. 2020. A hybrid bert model that incorporates label semantics via adjustive attention for multi-label text classification. *Ieee Access*, 8:152183–152192.

Xiaolong Chen, Jieren Cheng, Jingxin Liu, Wenghang Xu, Shuai Hua, Zhu Tang, and Victor S Sheng. 2022. A survey of multi-label text classification based on deep learning. In *Artificial Intelligence and Security: 8th International Conference, ICAIS 2022, Qinghai, China, July 15–20, 2022, Proceedings, Part I*, pages 443–456. Springer.

Xuange Cui, Wei Xiong, and Songlin Wang. 2022. Zhichunroad at semeval-2022 task 2: Adversarial training and contrastive learning for multiword representations. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 197–203.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Haytame Fallah, Patrice Bellot, Emmanuel Bruno, and Elisabeth Murisasco. 2022. Adapting transformers for multi-label text classification. In *CIRCLE (Joint Conference of the Information Retrieval Communities in Europe) 2022*.

Maik Fröbe, Matti Wiegmann, Nikolay Kolyada, Bastian Grahm, Theresa Elstner, Frank Loebe, Matthias Hagen, Benno Stein, and Martin Potthast. 2023. Continuous Integration for Reproducible Shared Tasks with TIRA.io. In *Advances in Information Retrieval. 45th European Conference on IR Research (ECIR 2023)*, Lecture Notes in Computer Science, Berlin Heidelberg New York. Springer.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. 2018. Adversarial example generation with syntactically controlled paraphrase networks. *arXiv preprint arXiv:1804.06059*.

Johannes Kiesel, Milad Alshomary, Nicolas Handke, Xiaoni Cai, Henning Wachsmuth, and Benno Stein. 2022. Identifying the human values behind arguments. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4459–4471.

Johannes Kiesel, Milad Alshomary, Nailia Mirzakhmedova, Maximilian Heinrich, Nicolas Handke, Henning Wachsmuth, and Benno Stein. 2023. Semeval-2023 task 4: Valueeval: Identification of human values behind arguments. In *Proceedings of the 17th International Workshop on Semantic Evaluation*, Toronto, Canada. Association for Computational Linguistics.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Xiaodong Liu, Hao Cheng, Pengcheng He, Weizhu Chen, Yu Wang, Hoifung Poon, and Jianfeng Gao. 2020. Adversarial training for large neural language models. *arXiv preprint arXiv:2004.08994*.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Junyu Lu, Hao Zhang, Tongyue Zhang, Hongbo Wang, Haohao Zhu, Bo Xu, and Hongfei Lin. 2022. Guts at semeval-2022 task 4: Adversarial training and balancing methods for patronizing and condescending language detection. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 432–437.

Qianwen Ma, Chunyuan Yuan, Wei Zhou, and Songlin Hu. 2021. Label-specific dual graph neural network for multi-label text classification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3855–3864, Online. Association for Computational Linguistics.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2017. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

Tushar Maheshwari, Aishwarya N. Reganti, Samiksha Gupta, Anupam Jamatia, Upendra Kumar, Björn Gambäck, and Amitava Das. 2017. A societal sentiment analysis: Predicting the values and ethics of individuals by analysing social media content. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 731–741, Valencia, Spain. Association for Computational Linguistics.

Nailia Mirzakhmedova, Johannes Kiesel, Milad Alshomary, Maximilian Heinrich, Nicolas Handke, Xiaoni Cai, Barriere Valentin, Doratossadat Dastgheib, Omid Ghahroodi, Mohammad Ali Sadraei, Ehsaneddin Asgari, Lea Kawaletz, Henning Wachsmuth, and Benno Stein. 2023a. The Touché23-ValueEval Dataset for Identifying Human Values behind Arguments. *CoRR*, abs/2301.13771.

Nailia Mirzakhmedova, Johannes Kiesel, Milad Alshomary, Maximilian Heinrich, Nicolas Handke, Xiaoni Cai, Barriere Valentin, Doratossadat Dastgheib, Omid Ghahroodi, Mohammad Ali Sadraei, et al. 2023b. The touch\'e23-valueeval dataset for identifying human values behind arguments. *arXiv preprint arXiv:2301.13771*.

Takeru Miyato, Andrew M Dai, and Ian Goodfellow. 2016. Adversarial training methods for semi-supervised text classification. *arXiv preprint arXiv:1605.07725*.

Ankit Pal, Muru Selvakumar, and Malaikannan Sankarasubbu. 2020. Multi-label text classification using attention-based graph neural network. *arXiv preprint arXiv:2003.11644*.

Milton Rokeach. 1973. *The nature of human values.* Free press.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Dilin Wang, Chengyue Gong, and Qiang Liu. 2019. Improving neural language modeling via adversarial training. In *International Conference on Machine Learning*, pages 6555–6565. PMLR.

Yi Wu, David Bamman, and Stuart Russell. 2017. Adversarial training for relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1778–1783.

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S. Yu. 2021. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32(1):4–24.

Ximing Zhang, Qian-Wen Zhang, Zhao Yan, Ruifang Liu, and Yunbo Cao. 2021. Enhancing label correlation feedback in multi-label text classification via multi-task learning. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1190–1200, Online. Association for Computational Linguistics.

Chen Zhu, Yu Cheng, Zhe Gan, Siqi Sun, Tom Goldstein, and Jingjing Liu. 2019. Freelb: Enhanced adversarial training for natural language understanding. *arXiv preprint arXiv:1909.11764*.

Danqing Zhu, Wangli Lin, Yang Zhang, Qiwei Zhong, Guanxiong Zeng, Weilin Wu, and Jiayu Tang. 2021. At-bert: Adversarial training bert for acronym identification winning solution for sdu@ aaai-21. *arXiv preprint arXiv:2101.03700*.