Andronicus of Rhodes at SemEval-2023 Task 4: Transformer-Based Human Value Detection using Four Different Neural Network Architectures

Georgios Papadopoulos
University of Piraeus,
Dept. of Digital Systems,
Piraeus, 18435, Greece
georgepap@unipi.gr

Marko Kokol
Semantika Research, Semantika d.o.o.,
Zagrebška 40a, 2000 Maribor, Slovenia
marko.kokol@semantika.eu

Maria Dagioglou
Inst. of Informatics & Telecommunications,
National Centre for Scientific Research
(N.C.S.R.) "Demokritos"
Aghia Paraskevi, Attica, Greece
mdagiogl@iit.demokritos.gr

Georgios Petasis
Inst. of Informatics & Telecommunications,
National Centre for Scientific Research
(N.C.S.R.) "Demokritos"
Aghia Paraskevi, Attica, Greece
petasis@iit.demokritos.gr

Abstract

This paper presents our participation to the "Human Value Detection" shared task (Kiesel et al., 2023), as "Andronicus of Rhodes". We describe the approaches behind each entry in the official evaluation, along with the motivation behind each approach. Our best-performing approach has been based on BERT large, with 4 classification heads, implementing two different classification approaches (with different activation and loss functions), and two different partitioning of the training data, to handle class imbalance. Classification is performed through majority voting. The proposed approach outperforms the BERT baseline, ranking in the upper half of the competition.

1 Introduction

The "Human Value Detection" shared task (Kiesel et al., 2023) relates to the identification of values that can be associated with an argument. In the context of the shared task, a single argument is represented by a single premise, a conclusion, and information on what the stance of the premise is towards the conclusion. Each argument can be classified into one or more value categories, among the 20 value categories, based on Schwartz (1994). The shared task covers a single language, English, providing 5393 examples for training, 1896 examples for validation, while participating systems are evaluated on 1576 test examples (Mirzakhmedova et al., 2023).

With the ever-increasing ubiquity of artificial intelligence in real-world usage scenarios, the awareness of the so-called alignment problem (Christian, 2020) is increasing, as is the need to develop novel strategies to measure and verify the alignment of machine learning (ML)/artificial intelligence (AI) systems (Brown et al., 2021) with human values.

However, high-quality datasets and novel (moral) value recognition methods are needed to further develop this area. Some existing initiatives presenting datasets for human value detection are based: either on specific types of texts, such as the SemEval argument annotation task (Mirzakhmedova et al., 2023), or on broader over-arching studies of values and their analysis in different spatiotemporal contexts, such as the VAST Project approach (Castano et al., 2021). Nevertheless, since human value recognition in texts is a problem that can be challenging even for humans, let alone for machines (Haas, 2020), despite the existing datasets, additional methods need to be developed to detect values in various scenarios.

In order to approach the challenge, several different natural language processing (NLP) and ML/AI approaches have been previously proposed and tested (Kiesel et al., 2022; Yu et al., 2020; Cortiz, 2021; Brown et al., 2021). Several modern and well-performing techniques for text classification (of shorter text sequences) are based on artificial neural networks (ANN), commonly based on LSTM, GRU, CNNs, and RNNs (Yu et al., 2020), and frequently using the so-called transformer architectures (Cortiz, 2021; Devlin et al., 2018), and attention (Vaswani et al., 2017).

To that effect, we focused on building upon the previously described successful approaches. Specifically, we designed different ANN architectures building upon BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019) pre-trained models for the English language and fine-tuning them using the SemEval 2023 human values dataset (Mirza-

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 542–548

khmedova et al., 2023). Additionally, we utilized approaches that have proven successful in short text feature extraction and classification, such as siamese networks (Yan et al., 2018), and training the networks by treating the under-represented classes separately (Kokol et al., 2022) by providing a specialised classification head.

2 Background

As mentioned in the introduction, our approach was built upon BERT and RoBERTa models, with the models incorporated into a wider ANN with three different approaches (additional details are provided in the System Overview):

- A single classifier with 1 classification head outputting 20 scalars, each of which corresponds to a value category; the *sigmoidF1* loss function (Bénédict et al., 2021) was used to train this classifier.
- A single multi-label classifier for all 20 values and 4 decision heads; 2 for the majority and 2 for the minority classes, with 2 different loss functions per class. Final classification is performed through majority voting.
- A siamese network classifier for all 20 values and 4 decision heads; 2 for the majority and 2 for the minority classes, with 2 different loss functions per class.

Having as a starting point a simple multi-label classifier based on BERT, we explored the training part of the shared task dataset. As evident from the initial classification results on the validation dataset and Figures 1 and 2, class imbalance has a significant impact on performance. Thus, handling the class imbalance in the dataset was the main motive for the approaches that we have applied on the shared task.

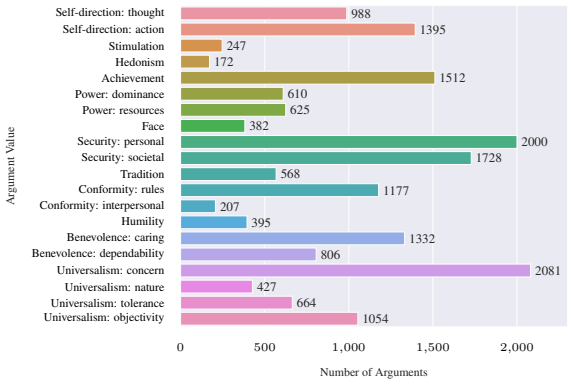

Figure 1: Number of arguments classified in each value category.

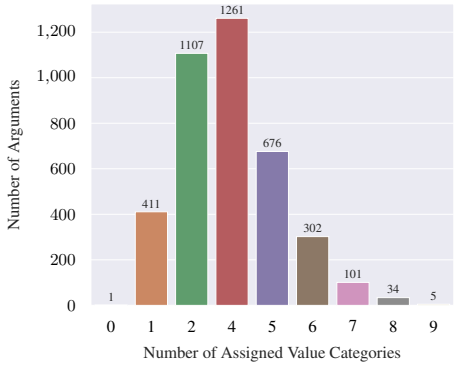

Figure 2: Number of value categories per argument.

Our first attempt was based on the observation that after a few training epochs on a simple multi-label BERT-based classifier with *BCE* (binary cross entropy) loss, the validation F1 score was either increasing or remaining stable, while the validation loss was also increasing. This motivated us to seek an objective function that better estimates the F1 metric; a suitable differentiable candidate is the *sigmoidF1* loss (Bénédict et al., 2021). This approach is further detailed in section 3.1, and constitutes our first entry to the shared task (entry "2023-01-28-03-02-04" in Table 2).

Despite the fact that we have not observed a significant correlation among the value categories, we opted to design a multi-label approach. Initially seeking to exploit multi-task learning (using stance prediction between premise and conclusion as a secondary task), we ended up using slightly different ways in performing the same multi-label classification problem: A classification head that involves a sigmoid layer (a typical approach for multi-label classification), and a classification head that involves a softmax layer (typically used in multi-class classification), following (Mahajan et al., 2018). The two classification heads are combined through voting, to produce the final classification, selecting only the value categories that both heads agree.

Returning to the problem of class imbalance, we tried to "sub-sample" the dataset, through the removal of associations with the majority classes from training examples that are associated with more than 3 value categories, including any majority class. The motivation behind this removal of assignments was the hypothesis that there may have been an annotation bias towards some values (like "Universalism: concern", "Security: personal", and "Security: societal"), either due to annotators' personal biases, or due to the presence of values in the selected arguments. Despite the fact that the

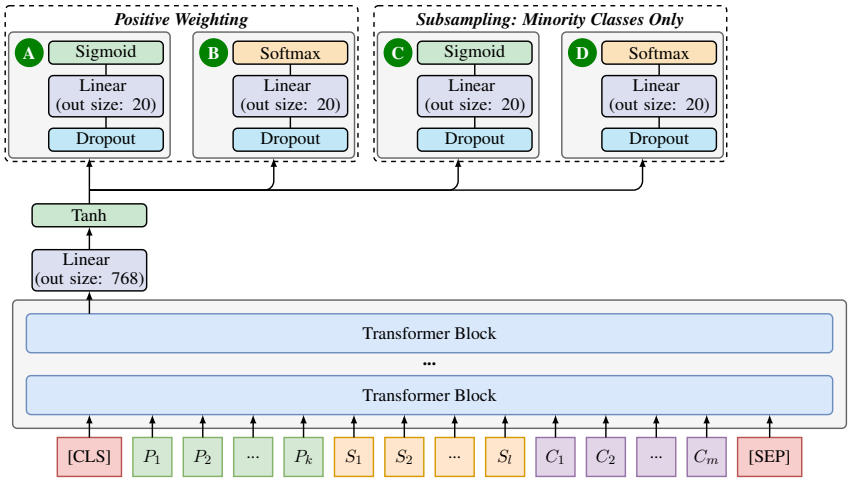

Figure 3: The high-level architecture of the single multi-label classification ANN.

three majority classes were significantly reduced (as shown in Table 1), this sub-sampling did not have any significant effect in the performance of the baseline BERT classifier (F1 has not significantly changed on the validation dataset). As a result, we opted to apply up-sampling and sub-sampling simultaneously: we used positive weighting in both classification heads, that try to balance positive and negative examples within value categories, and we added two more (identical) classification heads, which are trained only on argument examples that are assigned to minority classes[1]. Again the final classification is performed through majority voting among the 4 classification heads. The architecture of this approach is shown in Figure 3, and constitutes our second and third entries to the shared task (entries "2023-01-28-03-07-13" and "2023-01-28-10-02-00" in Table 2). The third entry is the best performing approach (as evaluated on the test set) of our team.

Finally, we designed an approach which combines the idea of the end-to-end ensemble and simultaneous upsampling/subsampling, with a siamese network architecture (Figure 4), exploiting different pre-trained models, BERT and RoBERTa. This approach constitutes our fourth entry (entry "2023-01-30-16-39-54"), which is not part of the official evaluation, as it was submitted after the submission deadline (but before the evaluation results were made public).

Value Category	Initial	Sub-sampling
Universalism: concern	2081	1171
Security: personal	2000	916
Security: societal	1728	544

Table 1: Reduction in the association of argument examples (training dataset) with majority classes, after removing associations from examples having more than 3 associated value categories.

3 System Overview

Overall, all approaches were based on the Huggingface library (Wolf et al., 2019), utilising a PyTorch backend (Paszke et al., 2019), and a custom dataloader that was created for the task, described in more detail in the Experimental Setup[2].

3.1 Single Multi-label Classifier with SigmoidF1 Loss

This multi-label classifier employs a very simple approach: the BERT English base uncased model (Devlin et al., 2018) representations of [CLS] token is fed into a classification head with a single dense layer and dropout (0.5 probability) which produces 20 scalars; followed by a sigmoid activation function. The output, denoted by $\hat{\boldsymbol{y}}$, is a two-dimensional matrix of shape $[batch\ size, 20]$. The labels, $\boldsymbol{y}$, are also given in a two-dimensional matrix, thus we compute the *sigmoidF1* loss as

[1]Minority classes have been defined as the value categories that have a frequency lower than 850 in the combined training and validation datasets.

[2]The source code is publicly available through GitHub: https://github.com/DimitrisPatiniotis/Human-Value-Detection.

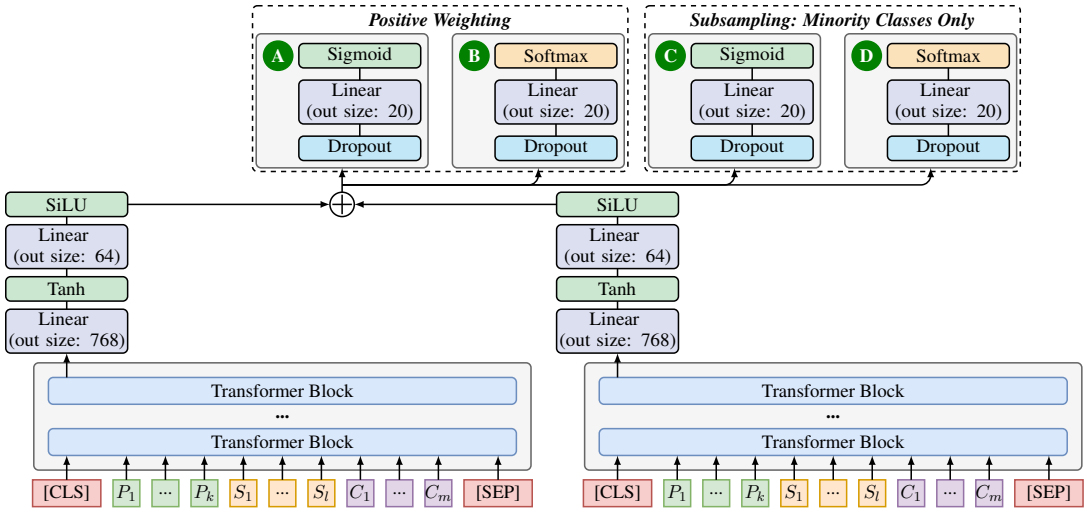

Figure 4: The high-level architecture of the siamese multi-label classification ANN.

follows:

$$\mathcal{L}_{\widetilde{F1}} = 1 - \frac{1}{C}\sum \widetilde{F1} \tag{1}$$

$$\widetilde{F1} = \frac{2\tilde{tp}}{2\tilde{tp} + \tilde{fn} + \tilde{fp}} \tag{2}$$

where $C = 20$, $\tilde{tp}$, $\tilde{fp}$, and $\tilde{fn}$ are rough surrogates of the true positives, false positives, and false negatives, respectively, calculated as:

$$\tilde{tp} = \frac{1}{N}\sum \hat{\boldsymbol{y}} \odot \boldsymbol{y} \tag{3}$$

$$\tilde{fp} = \frac{1}{N}\sum \hat{\boldsymbol{y}} \odot (\mathbb{1} - \boldsymbol{y}) \tag{4}$$

$$\tilde{fn} = \frac{1}{N}\sum (\mathbb{1} - \hat{\boldsymbol{y}}) \odot \boldsymbol{y} \tag{5}$$

Note that the sign $\odot$ represents the element-wise multiplication and $\mathbb{1}$ is a two-dimensional matrix of the same shape as $\hat{\boldsymbol{y}}$ and $\boldsymbol{y}$, filled with ones. We should mention that in the original *sigmoidF1* loss, the sum over the batch samples is calculated in Equations (3), (4), and (5), but we replaced it with the average over the batch samples since we discovered that this way yields higher performance. Similarly, we added the average over the classes in Equation (1), where neither the sum nor the average was utilized.

Furthermore, we have experimented with replacing the single dense layer head with a GRU head which takes as input the entire sequence of BERT-representations, i.e., the representations of all the words in the argument. In another trial, we replaced the single dense layer head with 20 concrete heads (each with its own dense layer), one for each value category. None of these yielded better results than the simple approach (as measured by the validation F1), thus we stuck with the single dense layer head. We have also performed experiments where we augmented the training dataset with the unlabeled test samples, applying the GAN-BERT method (Croce et al., 2020). Unfortunately, the results were inferior, probably due to the small number of unlabeled samples compared to the labeled ones.

3.2 Single Multi-label Classifier

The multi-label classifier utilises BERT (Devlin et al., 2018) as its backbone. Two versions were tested: one using the BERT base uncased English model (Devlin et al., 2018), and one using the BERT large uncased English model (Devlin et al., 2018) (Figure 3). Finally, the model feeds into 4 separate heads, all connected via a fully connected layer with dropout (with 0.1 probability), trained on 4 different tasks:

- Positively weighted classes, where under-represented classes are weighted to try equalizing the importance of classes, using two different activation functions: a) Sigmoid activation (Figure 3 (A)); b) Softmax activation (Figure 3 (B));
- Subsampling using minority classes only, using two different activation functions: a) Sigmoid activation (Figure 3 (C)); b) Softmax activation (Figure 3 (D));

The layer stacking is shown on the Figure 3, after the BERT (Devlin et al., 2018) Tanh activation layer, there is a dropout layer, followed by a Linear layer. The function for the dropout layer is:

$$\mathbf{h}_{drop} = \mathbf{r} \odot \mathbf{h} \tag{6}$$

where $\mathbf{h}$ is the input vector, $\mathbf{r}$ is a binary mask vec-

tor of the same shape as $\mathbf{h}$, and $\odot$ represents the element-wise multiplication operation. The mask vector $\mathbf{r}$ is generated by drawing each element independently from a Bernoulli distribution with parameter p, where p is the probability of keeping each element.

The function for the linear layer is the standard fully connected layer:

$$z = Wx + b \tag{7}$$

where x is the input vector, W is the weight matrix, b is the bias vector, and z is the output vector This is then fed into the Sigmoid activation function for heads (A) and (C) with the equation:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{8}$$

where z is the input to the sigmoid function. The sigmoid function outputs a value between 0 and 1, which can be interpreted as a probability or as an activation value for a neuron in the neural network.

The heads (B) and (D) use the Softmax activation function with the equation:

$$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}} \tag{9}$$

where z_i is the input to the softmax function for the i-th output neuron.

3.3 Siamese Network Classifier

The multi-value classifier utilised the BERT and BERT-based models as its backbone, combining the BERT base uncased English pre-trained model (Devlin et al., 2018), and the RoBERTa base uncased English language model (Liu et al., 2019), as shown in Figure 4.

Both models are then attached to a 64 neuron fully connected layers utilising the SiLU function (Elfwing et al., 2018), and combined using simple addition (summation). The PyTorch autograd (PyTorch, 2023) functionality is used during training for back-propagation.

The model feeds into the same 4 heads, as already mentioned in the description of the Single Multi-label Classifier, and shown on Figure 4.

The Siamese network utilises the output of BERT (Devlin et al., 2018) and feeds it into a linear layer (eq. 7), after which the SiLU function (Elfwing et al., 2018) is used, which is calculated by:

$$\text{SiLU}(z) = z \cdot \sigma(z) \tag{10}$$

where z is the input to the SiLU function and $\sigma(z) = \frac{1}{1+e^{-z}}$ is the sigmoid function. The output of the SiLU layer is then fed into the same heads as already described for the Single Multi-label Classifier.

4 Experimental Setup

The three different models also used different setups, and full details are available in GitHub[3], containing the full source code used for the task.

4.1 Single Multi-label Classifier with SigmoidF1 Loss

The experimental setup for this classifier included the:

- Example representation starts with the [CLS] token, followed by the conclusion, the special token [SEP], the premise, and another [SEP] at the end.
- A custom training loop was used, with a batch size of 8 and a learning rate of 0.00002.
- We exploited *early stopping* with patience of 5 epochs to avoid overfitting and reduce training time. After determining the epoch at which training should stop (based on the validation loss), we combined training and validation sets and trained the model again for the same number of epochs.

4.2 Single Multi-label Classifier

The experimental setup for the Single Multi-label classifier consisted of the following steps:

- Example representation starts with the special [CLS] token, followed by the premise, stance, conclusion, and finally, the special [SEP] token; as shown in Figure 3.
- The training dataset is augmented using the NLPAUG library (Ma, 2019); specifically, for each input, an additional input is generated using the Word2Vec augmentation function for the English language.
- Minority classes are determined, and a special subsample of only minority classes is created to be used with classification heads (C) and (D) (Figure 3).
- The Huggingface (Wolf et al., 2019) built-in trainer was used for training, using a batch size of 16 and a learning rate of 0.00005.

[3] https://github.com/DimitrisPatiniotis/Human-Value-Detection/

Test set / Approach	All	Self-direction: thought	Self-direction: action	Stimulation	Hedonism	Achievement	Power: dominance	Power: resources	Face	Security: personal	Security: societal	Tradition	Conformity: rules	Conformity: interpersonal	Humility	Benevolence: caring	Benevolence: dependability	Universalism: concern	Universalism: nature	Universalism: tolerance	Universalism: objectivity
Main																					
Best per category	.59	.61	.71	.39	.39	.66	.50	.57	.39	.80	.68	.65	.61	.69	.39	.60	.43	.78	.87	.46	.58
Best approach	.56	.57	.71	.32	.25	.66	.47	.53	.38	.76	.64	.63	.60	.65	.32	.57	.43	.73	.82	.46	.52
BERT	.42	.44	.55	.05	.20	.56	.29	.44	.13	.74	.59	.43	.47	.23	.07	.46	.14	.67	.71	.32	.33
1-Baseline	.26	.17	.40	.09	.03	.41	.13	.12	.12	.51	.40	.19	.31	.07	.09	.35	.19	.54	.17	.22	.46
2023-01-28-03-02-04	.47	.51	.62	.20	.23	.61	.37	.50	.24	.71	.58	.45	.57	.45	.13	.51	.25	.69	.75	.38	.49
2023-01-28-03-07-13	.47	.49	.58	.23	.13	.58	.36	.48	.21	.70	.59	.46	.53	.26	.18	.47	.26	.67	.74	.39	.47
2023-01-28-10-02-00	.48	.47	.65	.25	.29	.58	.35	.54	.30	.71	.60	.51	.54	.27	.14	.52	.31	.69	.76	.39	.48
2023-01-30-16-39-54*	.46	.51	.61	.20	.23	.55	.45	.47	.28	.71	.59	.50	.52	.34	.17	.54	.23	.71	.77	.36	.40

Table 2: Achieved F_1-score of team Andronicus-of-Rhodes per test dataset, from macro-precision and macro-recall (All) and for each of the 20 value categories. Approaches marked with * were not part of the official evaluation. Approaches in gray are shown for comparison: an ensemble using the best participant approach for each individual category, the best participant approach, and the organizer's BERT and 1-Baseline.

4.3 Siamese Network Classifier

The Siamese Network Classifier mostly used the same experimental setup as the Single Multi-label classifier, using the same pre-processing and training approach. However, after performing the initial training, the training and validation sets were combined, and 3% of the combined dataset was randomly selected as the validation set for final fine-tuning, before running the model on the test set.

5 Results

Our best approach ranked in the upper half of the competition, and achieved a result above the BERT baseline of the shared task. As seen in Table 2, our best model (based on BERT large with 4 classification heads) achieved an overall F1 score of 0.48, with the single multi-label models following with 0.47, and the siamese network reaching 0.46. We have not used any datasets beyond the official competition dataset for training[4]. There is a broad variability of per-category results: both within our models, as well as within all submitted models on the leaderboard.

6 Conclusion

Our approach ranks in the upper half of the competition. We have tested a few different approaches, all utilising transformer- and attention-based language models.

An increased performance over baseline was achieved by combining multi-task learning with heads that: a) implement the classification task differently, and b) are trained on different partitions of the data that simultaneously upsample/subsample the training examples.

Since different models showed different performance characteristics for some of the classes, an expansion of the idea would be the use of ensemble-based methods, potentially combined with Automated ML approaches, to further improve the results.

[4]We have used data augmentation and pre-trained models.

7 Acknowledgments

The research leading to these results has received funding from the European Union's Horizon 2020 research and innovation programme, in the context of VAST project, under grant agreement No 101004949. This paper reflects only the view of the authors and the European Commission is not responsible for any use that may be made of the information it contains.

References

Gabriel Bénédict, Vincent Koops, Daan Odijk, and Maarten de Rijke. 2021. sigmoidf1: A smooth F1 score surrogate loss for multilabel classification. *CoRR*, abs/2108.10566.

Daniel S. Brown, Jordan Schneider, Anca Dragan, and Scott Niekum. 2021. Value Alignment Verification. In *Proceedings of the 38th International Conference on Machine Learning*, pages 1105–1115. PMLR. ISSN: 2640-3498.

Silvana Castano, Alfio Ferrara, Stefano Montanelli, and Francesco Periti. 2021. From Digital to Computational Humanities: The VAST Project Vision. Publisher: Zenodo.

Brian Christian. 2020. *The Alignment Problem: Machine Learning and Human Values*. W. W. Norton & Company. Google-Books-ID: Lh_WDwAAQBAJ.

Diogo Cortiz. 2021. Exploring Transformers in Emotion Recognition: a comparison of BERT, DistillBERT, RoBERTa, XLNet and ELECTRA. ArXiv:2104.02041 [cs].

Danilo Croce, Giuseppe Castellucci, and Roberto Basili. 2020. GAN-BERT: Generative adversarial learning for robust text classification with a bunch of labeled examples. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2114–2119, Online. Association for Computational Linguistics.

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*.

Stefan Elfwing, Eiji Uchibe, and Kenji Doya. 2018. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Networks*, 107:3–11.

Julia Haas. 2020. Moral Gridworlds: A Theoretical Proposal for Modeling Artificial Moral Cognition. *Minds and Machines*, 30(2):219–246.

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, Dublin, Ireland. Association for Computational Linguistics.

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.

Peter Kokol, Marko Kokol, and Sašo Zagoranski. 2022. Machine learning on small size samples: A synthetic knowledge synthesis. *Science Progress*, 105(1):00368504211029777.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Edward Ma. 2019. Nlp augmentation. https://github.com/makcedward/nlpaug.

Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten. 2018. Exploring the limits of weakly supervised pretraining. In *Computer Vision – ECCV 2018*, pages 185–201, Cham. Springer International Publishing.

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. 2023. The Touché23-ValueEval Dataset for Identifying Human Values behind Arguments. *CoRR*, abs/2301.13771. Publisher: arXiv.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

PyTorch. 2023. Autograd mechanics¶.

Shalom H. Schwartz. 1994. Are There Universal Aspects in the Structure and Contents of Human Values? *Journal of Social Issues*, 50(4):19–45.

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

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Leiming Yan, Yuhui Zheng, and Jie Cao. 2018. Few-shot learning for short text classification. *Multimedia Tools and Applications*, 77(22):29799–29810.

Shujuan Yu, Danlei Liu, Wenfeng Zhu, Yun Zhang, and Shengmei Zhao. 2020. Attention-based LSTM, GRU and CNN for short text classification. *Journal of Intelligent & Fuzzy Systems*, 39(1):333–340. Publisher: IOS Press.