Søren Kierkegaard at SemEval-2023 Task 4: Label-aware text classification using Natural Language Inference

Ignacio Talavera Cepeda and **Amalie Brogaard Pauli** and **Ira Assent**
Department of Computer Science, Aarhus University, Denmark
ignacio@post.au.dk, ampa@cs.au.dk, ira@cs.au.dk

Abstract

In this paper, we describe our approach to Task 4 in SemEval 2023. Our pipeline tries to solve the problem of multi-label text classification of human values in English-written arguments. We propose a label-aware system where we reframe the multi-label task into a binary task resembling an NLI task. We propose to include the semantic description of the human values by comparing each description to each argument and ask whether there is entailment or not.

1 Introduction

Shared tasks are competitions organized by different academic institutions that put the scope on specific research problems that are challenging for the community, typically because of low data availability. Natural Language Processing (NLP) has many shared tasks annually (Palmer et al., 2021; May et al., 2019; Rodríguez-Sánchez et al., 2022). In these tasks, a problem that currently concerns the community is stated, and datasets are provided to the participants, which then have to try to solve that problem to get the highest score possible, and to contribute to establishing and comparing state-of-the-art approaches. SemEval 2023 is the 17th edition of the workshop, and it features 12 different NLP shared tasks, divided in several categories that include *semantic structure, discourse and argumentation, medical application* and *social attitudes*.

This paper addressed Task 4 which is in the field of discourse and argumentation mining. The task is a text classification task where participants have to classify whether or not an argument belongs to certain human value categories (Kiesel et al., 2023). The categories are compiled from a Social Science study by Kiesel et. al. (Kiesel et al., 2022). The participants are able to submit runs that detect either a subset of the seven most common categories or all of them. Each argument is divided into a premise text, a conclusion text and a binary stance of the premise to the conclusion, which can be either "in favour of" or "against". Kiesel et al. (2023) provide a dataset with the challenges of identifying a subset of fundamental values behind written arguments. The task is a multi-label text classification.

We propose to address the task by reframing the task from a multi-label task to a binary task which includes the textual semantic information of the labels. We reformulated the task to resemble a Natural Language Inference (NLI) task. The structure of NLI tasks is given a tuple with *premise-hypothesis* to decide if there is entailment, neutrality or contradiction between the premise and hypothesis (MacCartney and Manning, 2008). We propose a similar structure to our task at hand; we generate pairs of each argument with a textual description of each human-value. We then set the training label to whether or not the pairs entail each other or not. The motivation is 1) this generates more training examples for a model to learn from and 2) this makes the system label-aware, and 3) it makes the opportunity to leverage on large-scale pre-trained NLI models.

2 Background

The task The task is a multi-label text classification task on identifying values behind written arguments in English. The aim is to predict any subset of human-values a written argument is building on. There are in total 20 categories of values, which are generated from a social science study (Kiesel et al., 2022). The provided dataset consists of approximately 9000 arguments based on several sources. The dataset has a structure of premise, conclusion and stance, which together create the whole textual input. An input example from the dataset is:

- **Conclusion**: *We should limit judicial activism.*
- **Stance**: *against.*

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 1871–1877
July 13-14, 2023 ©2023 Association for Computational Linguistics

- **Premise**: *Each case has different circumstances and certain things need to be taken into consideration for each one. Judicial activism allows the court to consider the circumstances of each crime.*

In the dataset, the ’conclusion’ and ’stance’ appear multiple times as templates, whereas the ’premise’ is a human-written text to support the conclusion of a given stance. Hence, it is in the ’premise’, we want to look for the values underlying the argument.

The annotation of the dataset happen in an open-source process (Mirzakhmedova et al., 2023), where the annotators did not use the 20 value categories directly. Instead, they were asked to identify 54 more "direct/simpler values" (first-level values) which later are translated into the 20 value categories (second-level values). If any of the 54 first-level values are present in the argument then the corresponding second value is marked as present in the argument as well. This is supposed to be an easier/faster task for annotators to reply yes/no to whether each simpler value is present or not in an argument. Hence, we also hypothesise it might be easier for a model to use how the dataset originally where annotated by including this information in the system.

The training dataset contains a class imbalance which can be seen in Figure 1.

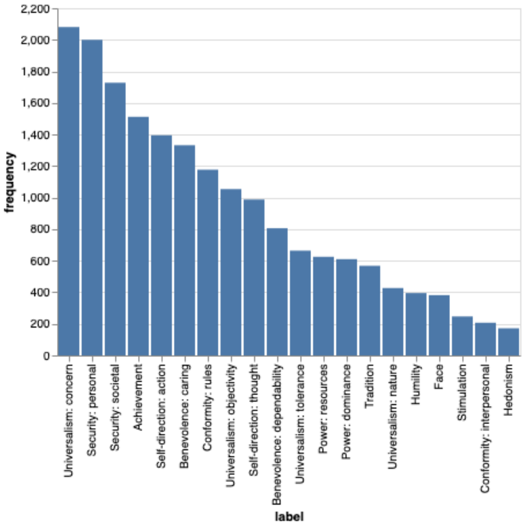


Figure 1: Label distribution in the training split.

Label-aware modelling In our system, we wanted to include the semantic information from the labels. Hence, we look to label-aware models. In classic text classification, labels are treated as a one-hot encoded vector or indices, but the language model does not process any information present in the label name, even though there is usually useful information encoded in the label names, and sometimes even a description about the label can provide information to the model. However, a recent stream of work has indeed worked on including the semantic information from the labels in the models, eg. (Mueller et al., 2022; Zhu et al., 2023).

Natural Language Inference We follow a modelling approach resembling a Natural Language Inference task (NLI). In an NLI, the task is to predict given two sentences if the sentences entail, contradict or are neutral regarding each other (MacCartney and Manning, 2008). On this task, there exists a large Standford Natural Language Inference (SNLI) corpus which is a collection of 570.000 human-written English sentence pairs (Bowman et al., 2015). We want to conduct experiments utilizing this by using a pre-trained NLI model as a backbone model for our training.

3 Our approach

3.1 Reframming

We propose to reframe the task from a multi-label task to a binary task resembling an NLI structure; We ask whether or not the value is entailing the premise of the argument or not. We, therefore, reformulate each data instance in the dataset to pairs of two sentences, and a binary label of zero or one. The **first sentence** in each pair is the premise of the argument from the dataset. We chose to only work with the premise since from our analysis of the dataset, we believe this is the user-generated input which is presenting the underlying values from a given conclusion and stance. The **second sentence** in our reformulated dataset, consists of a textual description of a value category. The descriptions are handcrafted for each of the 20 value classes using the information from the level 1 values with the intention to help the model understand what the level 2 values are (see section 2). We set the label for the training to indicate whether or not the value is in fact present in the argument or not. This restructuring gives a training set enlarged by a factor 20 corresponding to the number of classes. An example of this reframing of the training data is presented in Table 1.

Premise	Hypothesis	Entailment
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about being creative, curious or having freedom of thought.	Contradiction
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about choosing own goals, being independent, having freedom of action or having privacy.	Contradiction
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about having an exciting, varied life or be daring.	Contradiction
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about having pleasure.	Contradiction
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about being ambitious, having success, being capable or being intellectual.	Contradiction
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about having influence or having the right to command.	Contradiction
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about having wealth.	Contradiction
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about having social recognition or having good reputation.	Contradiction
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about having a sense of belonging, having good healt, being neat and tidy and having a comfortable life.	Entailment
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about having a safe country or a safe society.	Contradiction
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about respecting traditions or holding religious faith.	Contradiction
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about being compliant, being self-disciplined or be behaving properly.	Contradiction
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about being polite and be honoring elders.	Contradiction
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about being humble and having life accepted as it is.	Contradiction
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about being helpful, being honest, being forgiving or having the own family secured.	Contradiction
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about being responsible or having loyalty towards friends.	Contradiction
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about having equality, being just or having peace.	Entailment
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about protecting the environment or about nature.	Contradiction
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about being broadminded or accepting others.	Contradiction
payday loans create a more impoverished society with their crazy payback rates.	This sentence is about being logical and objective.	Contradiction

Table 1: Example of the 20 entailment data instances produced from a multi-label instance.

3.2 Model pipeline

After reframing the dataset into a binary task where each argument is compared to a textual written description of each value class, we are ready to start the training. In the training, we fine-tune pretrain Transformer-based models (Vaswani et al., 2017) which take two sentences as inputs. More concretely, we experiment with models based on the RoBERTa architecture (Liu et al., 2019), and with RoBERta models which are already once fine-tuned on the SNLI corpus (Bowman et al., 2015). We fine-tune further on our tasks dataset. If the model predicts an entailment between the argument and the value category, we considered that the premise belongs to the category. At the end of the training or the inference phase, the instances are translated back to the original multi-label structure.

4 Experimental Setup

We use the HuggingFace implementation of training transformer-based models (Wolf et al., 2020). Instead of focusing on the base model of RoBERTa, we explore different Transformer models pretrained over NLI datasets, like SNLI, to fine-tune a model already good at an NLI task. However, we avoid really large models, to be mindful of computational resources and to be able to train in a manageable amount of time. We chose to experiment with the available model *pepa/roberta-base-snli*[1] which is a version of RoBERTa base, fine-tuned over the SNLI dataset.

[1] https://huggingface.co/pepa/roberta-base-snli

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
2023-01-25-15-47-28	.49	.53	.58	.19	.30	.58	.35	.50	.27	.75	.62	.59	.53	.58	.18	.54	.15	.73	.77	.38	.39

Table 2: Achieved F_1-score of team soren-kierkegaard per test dataset, from macro-precision and macro-recall (All) and for each of the 20 value categories. Approaches in grey 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.

4.1 Data splits

The main data splits provided by the competition consist of a total of 8865 instances, divided into a training split (publicly annotated, 61%), a validation split (publicly annotated, 21%) and a test split (labels were not published, 18%). In order to effectively have a test split that could be used during the training phases (as the one that the organization provided was unlabelled and it was the one that the predictions should be inferred over and sent as the submission) and the hyper-parameter optimization, we merge all the instances again and develop our own splits. The total number of instances available consists of both the training and the validation split combined (all the labelled data), a total of 7289. Our training dataset has 55% of the instances (4008), our validation dataset has 25% of the instances (1820), and the development dataset has 20% of the instances (1461).

4.2 Hyper-Parameter Optimization

Our experiment setup gives promising results during the initial testing, where it obtained a micro-averaged F1-score of around 0.5 over all classes on our split in the development set. We therefor continue with the entailment approach using *pepa/roberta-base-snli*, and perform a Hyper-Parameter Optimization (HPO) (Yu and Zhu, 2020) process in three phases. Firstly, we focus on large ranges of hyper-parameters, and then we narrowed the random search values to lower intervals, focusing on the results obtained, and added the dropout value as a new attribute of the HPO to further explore the parameter space. The two best models (according to the micro-averaged F1 score) are fully trained in the third phase, including the translation of the predictions to the one-hot multi-label vector. The two first steps of this process are summarized in Table 4. The first HPO process consisted of 90 runs, and the second consisted of 50. The values for the hyperparameters were chosen following a normal distribution between the values in the intervals, and those values were manually narrowed from the first to the second HPO.

At the end of the second HPO, the two runs with the highest accuracy scores are the ones seen in Table 5. Those two best runs are retrained, as shown in Figure 2. We can see that the training behavior is very similar, and the hyperparameters that define these runs are also similar, but run 1 outperforms run 2 by a small margin. Therefore, the hyperparameters of run 1 are chosen for the final model.

The best run of our HPO process obtain the results over the development dataset that can be seen in Table 3. There have been a significant improvement from the original testing using the entailment approach.

5 Results

We submitted our system to the official leaderboard. It achieves an overall F1 score of 0.49, being the 13th best-performing model over the main test dataset. The results per category, and also the com-

	Precision	Recall	F1-Score
Micro-averaged	0.70	0.46	0.55
Macro-averaged	0.63	0.37	0.45

Table 3: Results of the best run of the HPO process over the development set

parison between our model and some of the best models and baselines can be seen in Table 2.

The model performs better in the categories that have more instances, as there is a class imbalance in the dataset (see Figure 1). Therefore, we conclude that the critical factor in the model performance is the number of instances per category, especially due to the high number of classes and their multi-label aspect.

6 Conclusion

We have submitted predictions to all value categories in Task 4 SemEval 2023. We have proposed reforming the multi-label task into a binary task and making the system label-aware. We have cast the problem to resemble an NLI task and thereby be able to transfer abilities from already pre-trained backbone models for this task. We have performed hyper-parameter tuning to find the best set for our system. Our submission rank above the standard BERT baseline and is also in the best half of all submission in terms of overall macro f1. However, we notice some limitations in our system and set of experiments: First 1) we notice a class imbalance in the training data which we do not mitigate well. We expect different sampling strategies might help. Second, 2) we do not report ablation studies. It could be relevant to examine how much e.g the reframing improves the performance and what performance boost comes from using a pre-trained NLI model as a backbone.

References

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

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, and 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.

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*.

Bill MacCartney and Christopher D. Manning. 2008. Modeling semantic containment and exclusion in natural language inference. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 521–528, Manchester, UK. Coling 2008 Organizing Committee.

Jonathan May, Ekaterina Shutova, Aurelie Herbelot, Xiaodan Zhu, Marianna Apidianaki, and Saif M. Mohammad, editors. 2019. *Proceedings of the 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics, Minneapolis, Minnesota, USA.

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.

Aaron Mueller, Jason Krone, Salvatore Romeo, Saab Mansour, Elman Mansimov, Yi Zhang, and Dan Roth. 2022. Label semantic aware pre-training for few-shot text classification.

Alexis Palmer, Nathan Schneider, Natalie Schluter, Guy Emerson, Aurelie Herbelot, and Xiaodan Zhu, editors. 2021. *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*. Association for Computational Linguistics, Online.

Francisco Rodríguez-Sánchez, Jorge Carrillo-de Albornoz, Laura Plaza, Adrián Mendieta-Aragón, Guillermo Marco-Remón, Maryna Makeienko, María Plaza, Julio Gonzalo, Damiano Spina, and Paolo Rosso. 2022. Overview of exist 2022: sexism identification in social networks. *Procesamiento de Lenguaje Natural*, 69:229–240.

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.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Tong Yu and Hong Zhu. 2020. Hyper-parameter optimization: A review of algorithms and applications.

Xiaofei Zhu, Zhanwang Peng, Jiafeng Guo, and Stefan Dietze. 2023. Generating effective label description for label-aware sentiment classification. *Expert Syst. Appl.*, 213(PC).

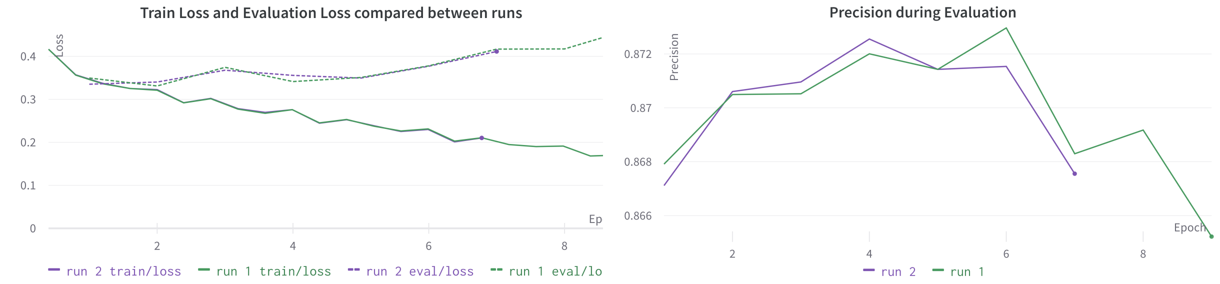
A Appendix A: Hyper-Parameters

Parameter Search Space	Learning Rate	Weight Decay	Batch Size	Warmup Steps	Dropout	Best Validation Accuracy (Binary)
First HPO	[1e-5 - 5e-5]	[0 - 0.03]	[8, 16, 32, 64]	[0 - 500]	-	0.859
Second HPO	4e-5	[0.01 - 0.03]	32	[0 - 300]	[0 - 0.3]	0.8718

Table 4: Search space and best results of the first two steps of the HPO process. Note that we are using validation accuracy because the validation was made with the data in the entailment format, so it is either entailment or contradiction. There is no difference between using accuracy or F1.

Parameters	Run 1	Run 2
Learning Rate	4e-5	4e-5
Weight Decay	0.01	0.02
Warmup Steps	100	100
Batch Size	32	32
Dropout	0.2	0.2
Best F1	0.8712 at epoch 3	0.8718 at epoch 6

Table 5: Best two runs after the second step of the HPO. Note that the epoch number is zero-based.



(a) Training and evaluation loss between runs

(b) Evaluation accuracy compared between runs

Figure 2: Training loss, evaluation loss and evaluation accuracy compared between two best runs after the second HPO process