BAMBAS at SemEval-2024 Task 4: How far can we get without looking at hierarchies?

Arthur B. Vasconcelos[1], **Luiz Matos**[1], **Eduardo Corrêa Gonçalves**[2], **Eduardo Bezerra**[3], **Aline Paes**[1] **and Alexandre Plastino**[1]
[1] Institute of Computing, Universidade Federal Fluminense, Niterói, RJ, Brazil
[2] National School of Statistical Sciences (ENCE/IBGE), RJ, Brazil
[3] Federal Center for Technological Education of Rio de Janeiro (CEFET/RJ), RJ, Brazil
{*athurbittencourt,lfmatosmelo*}*@id.uff.br*,
eduardo.correa@ibge.gov.br, *ebezerra@cefet-rj.br*, {*alinepaes,plastino*}*@ic.uff.br*

Abstract

This paper describes the BAMBAS team's participation in SemEval-2024 Task 4 Subtask 1, which focused on the multilabel classification of persuasion techniques in the textual content of Internet memes. We explored a lightweight approach that does not consider the hierarchy of labels. First, we get the text embeddings leveraging the multilingual tweets-based language model, Bernice. Next, we use those embeddings to train a separate binary classifier for each label, adopting independent oversampling strategies in each model in a binary-relevance style. We tested our approach over the English dataset, exceeding the baseline by 21 percentage points, while ranking in 23th in terms of hierarchical F1 and 11st in terms of hierarchical recall.

1 Introduction

In the multilabel classification problem (MLC), each instance may belong to zero, one, or multiple class labels. The goal is to learn a system to infer the correct labels of previously unseen instances (Gonçalves et al., 2018; Mylonas et al., 2023). MLC has several real-world applications, ranging from text categorization (Shimura et al., 2018) to protein and gene function prediction (Cerri et al., 2012). This work addresses a critical novel application of MLC: detecting persuasion techniques in memes, considering only their textual content, a subtask of SemEval-2024 task4[1].

The Merriam-Webster dictionary[2] defines *meme* as "an amusing or interesting item (such as a captioned picture or video) or genre of items that is spread widely online, especially through social media". Nonetheless, and unfortunately, in recent years, memes have been used not only to amuse people but also as a tool for disseminating disinformation in political campaigns (Renee, 2018; DeCook, 2018). Malicious actors embed sophisticated propaganda and persuasion techniques within these memes, employing psychological and rhetorical strategies. This manipulation extends to the memes' textual and visual components (Dimitrov et al., 2021).

Like other computational propaganda (Da San Martino et al., 2020), memes significantly influence public opinion. Their effectiveness stems from their widespread reach, potentially impacting millions of internet users globally. Additionally, memes are often not perceived as propaganda by these users, primarily because they do not mirror the appearance of conventional political advertisements (Nieubuurt, 2021).

As an effort to address this problem, SemEval-2024 Task 4 (Dimitrov et al., 2024) promoted a challenge in which competitors should develop algorithms to identify the use of persuasion techniques in memes, considering only their textual content (Subtask 1) or text and image together (Subtasks 2a and 2b). In this paper, we describe our approach to addressing Subtask 1. For this subtask, the shared-task organizers made available a collection of 8,500 texts in English extracted from real Internet memes (7,000 for training and the remaining divided into validation and dev sets). Each text may be assigned to a set of labels that indicate the persuasion techniques present in it[3]. There are a set of 20 possible labels organized in a hierarchy – thus, we have a hierarchical multilabel classification problem (Cerri et al., 2012). Some texts can have no label assigned, indicating they do not correspond to propaganda.

The shared task aimed to produce the best model according to the hierarchical-F1 metric. Test collec-

[1] https://propaganda.math.unipd.it/semeval2024task4/

[2] https://www.merriam-webster.com/wordplay/meme-word-origins-history

[3] Labels definitions are presented at https://propaganda.math.unipd.it/semeval2024task4/definitions.html

tions in four different languages were made available: English, Bulgarian, North Macedonian, and Arabic. Our team (BAMBAS) participated in the English challenge along with 31 other teams. We explored a lightweight approach based on three components. The first component is a language model from which we extract embedding features leveraging the [CLS] token. The second component is a binary relevance-based strategy to train 20 separate binary classifiers (one for each existing label) (Boutell et al., 2004). Our central inquiry focused on assessing the extent to which such a lightweight model that does not engage with the intricacies of hierarchical structures could be effective. The third core component handles the inherent imbalance of multilabel hierarchical problems by employing an independent oversampling strategy (Chawla et al., 2002; Menardi and Torelli, 2012) to reduce the imbalance between negative and positive examples present in each binary problem derived.

The hierarchical-F1 score of our submitted solution exceeded the baseline by 21 percentage points. In the hierarchical F1-based rank, we were the 23th out of 31 teams. However, when considering the hierarchical recall, we were ranked as 11st [4].

The rest of the paper is organized as follows. Section 2 briefly overviews MLC concepts relevant to this paper. Section 3 details our proposed system. In Sections 4 and 5, we present the experimental methodology and report the results, respectively. Finally, Section 6 brings the conclusion and future research directions.

2 Background

Over the last 20 years, MLC has been one of the most active research topics in machine learning (Mylonas et al., 2023). Among the several methods for multilabel learning in the literature (Bogatinovski et al., 2022; Prabhu et al., 2018), Binary Relevance (BR) (Boutell et al., 2004) stands out as one of the most prominent methods. This approach decomposes the multilabel problem into q binary problems, where q is the number of labels. Then, one binary classifier is independently trained for each label. The labels of new instances are predicted by combining the outputs of each classifier.

The BR method offers several key advantages. Firstly, its simplicity and intuitiveness make it highly accessible. Additionally, BR models can predict label sets not present in the training set, owing to their composition as a series of independent binary classifiers. Most crucially, BR has consistently exhibited high prediction accuracy values across various domains. In a recent extensive experimental comparison (Bogatinovski et al., 2022) involving 26 methods across 42 datasets, models utilizing BR outperformed all models trained with other different transformation strategies.

Nonetheless, the BR method suffers from three major drawbacks. First, it ignores the possible correlations among labels (Zhang et al., 2018). Second, BR has high training and prediction times for problems in which the number of labels is huge (tens of thousands to millions) (Prabhu et al., 2018). Third, its predictive performance is affected by class imbalance, which occurs when the number of examples relevant to each label is much inferior to the number of irrelevant ones (Mylonas et al., 2023; Zhang et al., 2018).

We consider that the first two drawbacks are not crucial for addressing SemEval-2024 Task 4 Subtask 1, as the number of labels in the problem is not large ($q = 20$) and there is no strong correlation between any pair of labels in the training set. More specifically, we found that the highest Pearson correlation value is 0.13 – between labels "Glittering generalities (Virtue)" and "Flag-waving". On the other hand, we consider that the issue of class imbalance needs to be taken into account as the imbalance ratio (ratio of negative to positive examples) is 47.38 on average in the training set, and the maximum value reaches 332.33 for the label "Obfuscation, Intentional vagueness, Confusion". Our approach is detailed in the next section.

3 System overview

The shared task proposed in SemEval-2024 Task 4 comprises an output of one or more labels – in case the meme is a propaganda – disposed in a hierarchical taxonomy of persuasion techniques. The root of such hierarchy is naturally labeled *persuasion*, while the second level has three possible branches: *ethos*, *pathos*, *logos*. While *ethos* and *logos* conduct to labels in a third level, *pathos* branch connects directly to the persuasion techniques – the leaves of the tree. This way, the final output can be one or more paths from the root to some leaf.

Handling such a hierarchical structure directly is quite challenging in machine learning. The algorithms should accurately predict multiple outputs

[4]Our code and experiments are available at https://github.com/MeLLL-UFF/bambas

while respecting the labels' hierarchical relationships. However, errors can propagate down the hierarchy. Moreover, some paths have very few instances, adding another layer of complexity to the problem: data sparsity and imbalance.

Therefore, our primary solution to the problem was to investigate how far an algorithm that disregards the hierarchy could go. Additionally, we also decided not to handle the multiple labels directly. However, employ the binary-relevance approach and consider a component to handle imbalance by adding synthetic instances with SMOTE (Chawla et al., 2002) and RandomOverSampler (Leevy et al., 2018), for each binary problem.

Algorithm 1 depicts the training procedure and Algorithm 2 the inference. Our method hinges on three core components. The first one creates the features from the meme textual content, leveraging a pre-trained language model (line 3 in Algorithm 1). The second component addresses class imbalance by creating synthetic instances (line 10 in Algorithm 1). The third component trains independent binary classifiers (line 12 in Algorithm 1), employing the binary-relevance strategy. During the inference phase, each label classifier undergoes evaluation, and the instance is assigned all the positive classifications predicted by each classifier.

Algorithm 1 Top-level Training Algorithm of BAMBAS team participation in SemEval-2024 Task4

1: $feats \leftarrow \emptyset, pos \leftarrow \emptyset, neg \leftarrow \emptyset, c_{labels} \leftarrow \emptyset$
2: **for** $meme \in dataset$ **do**
3: $\quad emb \leftarrow ptlm(meme.text)$
4: $\quad feats.append(\text{CLS token from } emb)$
5: $\quad$ **for** $label \in meme.labels$ **do**
6: $\quad\quad pos[label] \leftarrow pos[label] \cup meme.index$
7: $\quad$ **for** $label \notin meme.labels$ **do**
8: $\quad\quad neg[label] \leftarrow neg[label] \cup meme.index$
9: **for** $label \in labels$ **do**
10: $\quad aug\_pos[label], aug\_neg[label] \leftarrow oversampler(feats, pos[label], neg[label], rate)$
11: **for** $label \in labels$ **do**
12: $\quad c_{label} \leftarrow train\_classifier(feats, aug\_pos[label], aug\_neg[label])$
13: $\quad c_{labels} \leftarrow c_{labels} \cup c_{label}$
14: **return** c_{labels}

Algorithm 2 Top-level Inference Algorithm of BAMBAS team

1: $emb \leftarrow ptlm(meme\_text)$
2: $plabels \leftarrow \emptyset$
3: **for** $label \in labels$ **do**
4: $\quad plabel \leftarrow c_{label}(emb)$
5: $\quad$ **if** $plabel = True$ **then**
6: $\quad\quad plabels \leftarrow plabels \cup label$
7: **return** $plabels$

3.1 Extracting embedding from a pre-trained language model

In line with our straightforward premise, we have implemented a feature-based strategy that utilizes embeddings from pre-trained language models (PTLMs). The textual content of each meme is processed through the PTLM, allowing our system to capture the numeric feature vector from the [CLS] token. This method effectively harnesses the power of PTLMs to distill complex language information into a manageable form for further training our classifiers.

Our selection choice for PTLMs includes a writing free-style multilingual model, namely, XLM-RoBERTa (Conneau et al., 2020) and two informal writing style models, one monolingual (BERTweet (Nguyen et al., 2020)) and one multilingual (Bernice (DeLucia et al., 2022)).

XLM-RoBERTa is a multilingual adaptation of the RoBERTa model, pre-trained on a 2.5TB of data across 100 languages. RoBERTa itself is a transformers model trained on large raw text corpora. The key training method used is Masked Language Modeling (MLM), where 15% of the words in a sentence are masked, and the model predicts these masked words, learning a bidirectional representation of the sentence. BERTweet is a monolingual model trained from 850M Tweets. It has the same architecture as BERT-base but was trained using the RoBERTa pre-training procedure. Bernice is a multilingual RoBERTa language model trained from 2.5 billion tweets.

3.2 Training classifiers for each class

In our approach, we implemented the binary relevance strategy to train a suite of independent classifiers, each tailored to manage a binary prediction, of whether a meme belongs to a specific label. Our model comprises independent binary classifiers, each aligned to a distinct persuasion technique. Un-

der this strategy, a classifier corresponding to a label k is trained using a targeted approach: instances labeled with k are treated as positive examples, while all other instances are considered negative. This selective process ensures that each classifier becomes specialized in precisely identifying its respective label.

For instance, consider a meme m_j tagged with three labels (k_1, k_2, k_3). This meme serves as a positive training example for the classifiers c_{k_1}, c_{k_2}, and c_{k_3}, contributing to their ability to recognize these specific labels. Conversely, another meme m_z tagged with label (k_1) not only acts as a positive example for training the classifier c_{k_1} but also serves as a negative instance for c_{k_2} and c_{k_3}. This dual role of memes in the training process, as both positive and negative examples depending on their label associations, underscores each classifier's nuanced and specialized training within our binary relevance framework.

During the inference phase, each meme is processed through all the classifiers in our system. If a particular classifier predicts the meme as a positive instance, the corresponding label is assigned to the meme. By the time this processing is finished, the input meme accumulates a set of labels, each representing a positive prediction from the respective binary classifiers. This method ensures that the meme is comprehensively evaluated for all potential labels.

3.3 Creating synthetic instances

Considering the inherently imbalanced nature typical of multilabel hierarchical tasks (Mylonas et al., 2023; Zhang et al., 2018), we address this challenge by oversampling the dataset with synthetic instances. This approach is designed to equalize the number of examples for each binary classifier, thereby mitigating the imbalance issue. Our system generates synthetic examples independently for each binary classifier.

We leveraged two strategies: a simple random oversampler and the widely-used SMOTE (Synthetic Minority Oversampling Technique) (Chawla et al., 2002). SMOTE operates by identifying examples that are closely situated in the feature space. It then generates a line connecting these examples and creates a new and synthetic sample at a point along this line.

More precisely, for each classifier c_{k_i}, the process starts by selecting a random example from the minority class. Next, it identifies n nearest neighbors for this example. From these neighbors, one is randomly chosen. Subsequently, a synthetic example is crafted at a randomly determined point between the chosen neighbor and the original example in the feature space.

4 Experimental setup

Our solution was implemented using HuggingFace (Wolf et al., 2020)[5] and scikit-learn (Pedregosa et al., 2011)[6] libraries. The experiments were conducted on an NVidia DGX-1, using a single Tesla P100 GPU with 16GB of VRAM. We conducted a step-by-step analysis to reach the final modeling decisions. The intermediate results, necessary to decide the components of our final solution, are reported with the validation set. The final solution was trained with the training set and we report the results of the English dev and test set. We proceed this way because the dev set could only be measured with the submission page before the release of its gold labels.

All results are reported with the competition's evaluation metric, a hierarchical variant of the F1-score (Kiritchenko et al., 2006). The metric considers the classification taxonomy, rewarding a full score for exact leaves prediction, and rewarding a partial score for ancestor predictions. The closer the predicted ancestor is to the correct labels, the higher the partial score. Additionally, we report the hierarchical variants of precision and recall.

The first analysis consists of defining the PTLM to extract the embeddings. We did not employ oversampling during this phase and applied a binary-relevance model using logistic regression. The PTLM and logistic regression hyperparameters were left as default. The meme textual content is presented to the PTLM without any pre-processing. Next, we explore 6 other classifiers besides logistic regression: decision tree, extra tree, extra trees, KNN, random forest, and ridge classifier. The last analysis focused on selecting the best oversampling strategy. We experimented with SMOTE and a random oversampling strategy, both implemented in the imbalanced-learn library[7]. All the results so far included 20 binary classifiers, each associated with a persuasion technique in the leaves of the tree. Then, we investigate a final possibility of including some internal nodes related to the classes that

[5] https://huggingface.co/
[6] https://scikit-learn.org/stable/
[7] https://imbalanced-learn.org/stable/

were worst classified. The best model from those analyses was submitted to the competition.

5 Results

Table 1 depicts the results achieved by each pre-trained language model mentioned in Section 3.1 considering logistic regression and no oversampling strategy. Bernice achieves the best overall hierarchical-F1 (H-F1) results. We hypothesize that it was trained with a large set of informal texts from tweets, presenting a writing style close to those found in memes. Then, we select Bernice for the next analyses and to submit our final solution.

PTLM	H-F1	H-Prec.	H-Rec.
Bernice	**0.4996**	0.6246	**0.4163**
BERTweet	0.4334	0.7202	0.3100
XLM-RoBERTa	0.2928	**0.7410**	0.1825

Table 1: Validation results for choosing the PTLM

The next analysis concerns the method used as the base classifier of the binary relevance strategy. Table 2 depicts the results of the binary relevance when executed with each classifier. Logistic regression conducted to the best H-F1 score. Because of that, we proceed to the final analysis with it.

Classifier	H-F1	H-Prec.	H-Rec.
Log. Regression	**0.4996**	0.6246	**0.4163**
Decision Tree	0.3993	0.3856	0.4141
Extra Tree	0.3885	0.3826	0.3946
Extra Trees	0.1024	0.6831	0.0554
KNN	0.4252	0.5824	0.3348
Random Forest	0.1561	**0.8091**	0.0864
Ridge	0.4027	0.7388	0.2768

Table 2: Validation results of distinct Classifiers

Next, we explore our third core component, the oversampling technique. Table 3 shows the results of running the random oversampler and SMOTE with 50/50 rate for oversampling, and also a hybrid version which combines the best oversampler for each binary classifier, using different oversampling rates: 0.1 to 1.0 with step of 0.1.

Finally, we included additional classifiers to some internal nodes of the hierarchy. Such an extension includes only the internal nodes corresponding to the least accurately classified leaf nodes. These nodes are "Ad Hominem", "Distraction" and "Logos". Table 4 shows the validation set results without and with the addition of those internal nodes. Recall improved with the combined strategy, while precision remained nearly identical.

Strategy	H-F1	H-Prec.	H-Rec.
No Oversampling	0.4996	**0.6246**	0.4163
50/50 SMOTE	0.5456	0.4510	0.6904
50/50 Random	0.5383	0.4395	**0.6944**
Combination	**0.5487**	0.4783	0.6435

Table 3: Validation Results of Oversampling Strategies

Classifier	H-F1	H-Prec.	H-Rec.
W/O int. nodes	0.5487	**0.4783**	0.6435
+ int. nodes	**0.5548**	0.4782	**0.6607**

Table 4: Validation results with some internal nodes

Given the preceding results, we selected that approach for the final submission. Table 5 shows the final results achieved by the solution we submitted to SemEval-2024 Task4 platform. In the first line, we highlight the results achieved on the dev set while the second line shows (in bold) the test set result.

Set	H-F1	H-Prec.	H-Rec.
dev	0.5759	0.5046	0.6707
test	**0.5767**	**0.5012**	**0.6788**

Table 5: Final results for the official submission on both dev and test sets

6 Conclusion

This paper addresses the SemEval-2024 competition with a lightweight solution to investigate how a model that neglects the hierarchy would behave in a hierarchical task. Our solution uses a tweets-based PTLM as a feature extractor, generates synthetic data to account for imbalance, and employs a binary relevance strategy to handle multiple labels. Our next step is to investigate training a structured output classifier that predicts the paths in the hierarchy. Moreover, given that oversampling strategies enhanced the performance of most of the classes, we plan to design other strategies explicitly tailored to the style of memes.

Acknowledgments

This research was financed by CNPq (National Council for Scientific and Technological Development), under grants 311275/2020-6

(Aline Paes) and 315750/2021-9 (Alexandre Plastino) and FAPERJ - *Fundação Carlos Chagas Filho de Amparo à Pesquisa do Estado do Rio de Janeiro*, processes SEI-260003/000614/2023, SEI-260003/002930/2024 (Aline Paes) and E-26/201.139/2022 (Alexandre Plastino).

References

Jasmin Bogatinovski, Ljupčo Todorovski, Sašo Džeroski, and Dragi Kocev. 2022. Comprehensive comparative study of multi-label classification methods. *Expert Systems with Applications*, 203:117215.

Matthew R. Boutell, Jiebo Luo, Xipeng Shen, and Christopher M. Brown. 2004. Learning multi-label scene classification. *Pattern Recognition*, 37(9):1757–1771.

Ricardo Cerri, Rodrigo C. Barros, and Andre C. P. L. F. de Carvalho. 2012. A genetic algorithm for hierarchical multi-label classification. In *Proceedings of the 27th Annual ACM Symposium on Applied Computing*, SAC '12, page 250–255, New York, NY, USA. Association for Computing Machinery.

Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. 2002. Smote: synthetic minority over-sampling technique. *J. Artif. Int. Res.*, 16(1):321–357.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Giovanni Da San Martino, Stefano Cresci, Alberto Barrón-Cedeño, Seunghak Yu, Roberto Di Pietro, and Preslav Nakov. 2020. A survey on computational propaganda detection. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, IJCAI'20, pages 4826–4832.

Julia R. DeCook. 2018. Memes and symbolic violence: #proudboys and the use of memes for propaganda and the construction of collective identity. *Learning, Media and Technology*, 43(4):485–504.

Alexandra DeLucia, Shijie Wu, Aaron Mueller, Carlos Aguirre, Philip Resnik, and Mark Dredze. 2022. Bernice: A multilingual pre-trained encoder for Twitter. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6191–6205, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Dimitar Dimitrov, Firoj Alam, Maram Hasanain, Abul Hasnat, Fabrizio Silvestri, Preslav Nakov, and Giovanni Da San Martino. 2024. Semeval-2024 task 4: Multilingual detection of persuasion techniques in memes. In *Proceedings of the 18th International Workshop on Semantic Evaluation*, SemEval 2024, Mexico City, Mexico.

Dimitar Dimitrov, Bishr Bin Ali, Shaden Shaar, Firoj Alam, Fabrizio Silvestri, Hamed Firooz, Preslav Nakov, and Giovanni Da San Martino. 2021. Detecting propaganda techniques in memes. 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 6603–6617, Online. Association for Computational Linguistics.

Eduardo Corrêa Gonçalves, Alex A. Freitas, and Alexandre Plastino. 2018. A survey of genetic algorithms for multi-label classification. In *2018 IEEE Congress on Evolutionary Computation (CEC)*, pages 1–8.

Svetlana Kiritchenko, Stan Matwin, Richard Nock, and A Fazel Famili. 2006. Learning and evaluation in the presence of class hierarchies: Application to text categorization. In *Advances in Artificial Intelligence: 19th Conference of the Canadian Society for Computational Studies of Intelligence, Canadian AI 2006, Québec City, Québec, Canada, June 7-9, 2006. Proceedings 19*, pages 395–406. Springer.

Joffrey Leevy, Taghi Khoshgoftaar, Richard Bauder, and Naeem Seliya. 2018. A survey on addressing high-class imbalance in big data. *Journal of Big Data*, 5(42).

Giovanna Menardi and Nicola Torelli. 2012. Training and assessing classification rules with imbalanced data. *Data Mining and Knowledge Discovery*, 28:92–122.

Nikolaos Mylonas, Ioannis Mollas, Bin Liu, Yannis Manolopoulos, and Grigorios Tsoumakas. 2023. On the persistence of multilabel learning, its recent trends, and its open issues. *IEEE Intelligent Systems*, 38(2):28–31.

Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. 2020. BERTweet: A pre-trained language model for English tweets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 9–14, Online. Association for Computational Linguistics.

Joshua Troy Nieubuurt. 2021. Internet memes: leaflet propaganda of the digital age. *Frontiers in Communication*, 5:547065.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Yashoteja Prabhu, Anil Kag, Shrutendra Harsola, Rahul Agrawal, and Manik Varma. 2018. Parabel: Partitioned label trees for extreme classification with application to dynamic search advertising. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, page 993–1002, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Diresta Renee. 2018. Computational propaganda: If you make it trend, you make it true. *The Yale Review*, 106(4):12–29.

Kazuya Shimura, Jiyi Li, and Fumiyo Fukumoto. 2018. HFT-CNN: Learning hierarchical category structure for multi-label short text categorization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 811–816, Brussels, Belgium. Association for Computational Linguistics.

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.

Min-Ling Zhang, Yukun Li, Xu-Ying Liu, and Xin Geng. 2018. Binary relevance for multi-label learning: an overview. *Frontiers of Computer Science*, 12:191 – 202.

A Validation set classification results per-label

Due to the imbalanced nature of the explored problem, we further investigate the classification results on a per-label basis. Table 6 describes the results for each label in the validation set. We include the internal nodes in the hierarchy alongside all leaves. To calculate scores for the internal nodes, predictions of any of their children are considered as correct node predictions. The best-performing label was "Appeal to Authority", which achieved the highest F1 score. The internal nodes "Logos" and "Ad Hominem" follows in second and third place, respectively. Also, most of the worst performing labels have scarce examples on the datasets, like "Vagueness, Confusion" and "Straw Man".

B After competition deadline results: multilabel classifiers

The solution presented in the paper relaxes the multiple labels per example setting and trains independent binary classifiers for each class. We additionally explored an alternative setup that leverages a multi-layer perceptron (MLP) classifier with a multilabel classification layer. We trained two classifiers in this way: the first follows the previous feature-based approach to train a multilabel feedforward (FF) MLP; the second adds a multilabel classification layer on top of the PTLM and fine-tunes all its weights. As before, the PTLM is Bernice.

The feature-based approach classifier includes a single 768-dimension hidden layer with scikit-learn default parameters. The fine-tuning approach runs for five epochs, with a learning rate of $3.9e-5$ and weight decay of $1e-3$, all selected with the validation set. Both approaches did not involve oversampling, and the classifiers were trained with the union of the train and validation sets and evaluated on the dev set during training.

Tables 7 and 8 depict the results for the dev and test sets. The results show the superior performance of the fine-tuning approach, with test set H-F1 score higher than our official competition's submission. Also, the standalone FF classifier achieved F1 above average, indicating that a dedicated oversampling strategy for the multilabel approach is a promising research avenue to explore further in the future.

Label	F1	Prec.	Rec.
Appeal to Authority	**0.7194**	0.6578	**0.7936**
Logos (internal node)	0.6965	**0.7307**	0.6653
Ad Hominem (internal node)	0.6751	0.6986	0.6530
Smears	0.5460	0.4971	0.6056
Loaded Language	0.5202	0.4782	0.5703
Name calling/Labeling	0.5119	0.4776	0.5517
Flag-Waving	0.4615	0.3870	0.5714
Black-and-White/Dictatorship	0.4000	0.3472	0.4716
Repetition	0.3859	0.3235	0.4782
Slogans	0.3650	0.2873	0.5000
Bandwagon	0.3000	0.2307	0.4285
Glittering Generalities (Virtue)	0.2807	0.2051	0.4444
Thought-Terminating cliché	0.2635	0.1868	0.4473
Exaggeration/Minimisation	0.2597	0.2000	0.0000
Appeal to Fear/Prejudice	0.2474	0.1714	0.4444
Distraction (internal node)	0.2439	0.3846	0.1785
Doubt	0.2222	0.1578	0.3750
Causal Oversimplification	0.1666	0.1282	0.2380
Whataboutism	0.0338	0.0263	0.0476
Presenting Irrelevant Data	0.0000	0.0000	0.0000
Reductio ad Hitlerum	0.0000	0.0000	0.0000
Vagueness, Confusion	0.0000	0.0000	0.0000
Straw Man	0.0000	0.0000	0.0000

Table 6: Validation set results for each task label, sorted by descending F1

Classifier	H-F1	H-Prec.	H-Rec.
Bernice$_{emb}$ $\rightarrow$ FF	0.5063	0.7257	0.3887
Bernice$_{class}$	**0.5724**	**0.7431**	**0.4655**

Table 7: Dev set results for the multilabel classifiers

Classifier	H-F1	H-Prec.	H-Rec.
Bernice$_{emb}$ $\rightarrow$ FF	0.5044	0.7177	0.3889
Bernice$_{class}$	**0.5840**	**0.7594**	**0.4744**

Table 8: English test set results for the multilabel classifiers