Legal argument mining on italian datasets
NLP 3-cfu Project Work

Daniele Santini and **Muhammad Saleem Ghulam**
Master's Degree in Artificial Intelligence, University of Bologna
{ daniele.santini2, muhammad.ghulam }@studio.unibo.it

Abstract

We try to transpose the techniques currently tested for legal argument mining on english datasets [5] to a corresponding italian dataset [4] and test new models and techniques to overcome the issues inherent with the italian dataset (smaller size, higher imbalance).

1 Introduction

The study of argumentation in legal contexts is a research topic that combines Artificial Intelligence and Law to determine how different claims and opinions are proposed, debated and evaluated, considering relations and inter-dependencies.[5]

Machine Learning techniques for Natural Language Processing are increasingly being explored to solve the tasks in this field, which include:

- Argument Detection (AD): classification task that given a sentence, classifies it as premise, conclusion, or neither;
- Argument Classification (AC): binary classification task that takes a sentence that is known to be argumentative and classifies it as premise or conclusion;
- Type Classification (TC): a multi-label classification problem where a sentence that is known to be a premise is classified as legal and/or factual;
- Scheme Classification (SC): multi-label classification task that given a sentence known to be a legal premise, classifies it according to its argumentation scheme [11]

In this work, we address the tasks AC and DC on the Italian VAT dataset [4]. Both tasks are affected by data imbalance, which is particularly severe in the SC task, as shown in Figure 2, where some classes have very few examples compared to the majority ones. To mitigate this issue, we applied different strategies, including focal loss [9] and data augmentation techniques.

2 Background

The field of legal argument mining has seen remarkable work on english language datasets. Multiple high-quality datasets have been realized, most notably Demosthenes [5]. However, to our knowledge, only one Italian language dataset has been released for this domain, specifically Italian VAT [4]. This dataset includes annotations for all of the tasks cited above. The size of this dataset, however, is limited and some scheme classes do not have sufficient support for training a machine learning model, and moreover the classes are not equally distributed.

This issue prompted our research for data augmentation techniques and models capable of handling this imbalance and successfully using this dataset for Argument Classification and Scheme Classification tasks.

Class imbalance is a widely studied issue in the field of Artificial Intelligence, the literature includes many solutions for different cases and with different outcomes[6]. These include:

- random oversampling of classes with smaller support;
- hidden space augmentation with techniques like GE3 [13], REPRINT [14] or ECRT [3];
- advanced losses like focal loss [9], self-adjusting dice loss [7] and label distribution-aware margin loss [2];
- dynamic curriculum learning [12]

Similarly to datasets, work on machine learning models for legal argumentation mining has also focused mainly on english language. Much fewer studies have been done on italian language specific

models (notably [4]). Some work has also been done to fine tune models specifically for the legal domain in italian language (notably *Italian-Legal-BERT* [8]).

3 System description

For both tasks, we used as baseline the SVM algorithm with TF–IDF embedding, which previously obtained the best results on these tasks in [5] and [4]. We then used *Italian-Legal-BERT* [8], a transformer pre-trained model on legal texts, and fine tuned it for each task with a specific classification head: a binary output for argument classification and a multi-binary output head for scheme classification.

The scheme classification task presents significant challenges due to its highly imbalanced and multi-label nature. To mitigate these issues, we employed strategies such as replacing the standard cross-entropy loss with weighted focal loss and applying data augmentation techniques to enhance the representation of rare classes. We also evaluated the performance of the model on the argument classification task using focal loss, since there is also an imbalance between the two classes.

To address class imbalance in both tasks, we used the *weighted focal loss*. This is a variant of cross-entropy loss that focuses more on samples that are difficult for the model, while assigning less weight to those predicted correctly. By incorporating class weights, the rare classes contribute more to the loss during training, preventing the model from simply learning to predict the majority classes. This approach helps the model achieve better performance on rare classes. The weights applied to the loss function were computed as the inverse frequency of each class by dividing the total number of samples by the count of samples for that class. Then, we applied a logarithmic transformation and finally, we divided the weights by the smallest one, in order to enlarge the weights of rare classes.[9]

We applied two data augmentation approaches to the training set. During augmentation, if a sample from a rare class also belonged to a majority class, that majority class was augmented as well to maintain consistency.

The first approach consists of translating the original sentences into English using *Google Translator*. Then, with the *NLPaug* library[10], we applied the contextual augmenter based on BERT-large, which replaces words by leveraging the contextual embeddings provided by the model, in order to generate the desired number of augmented samples. The number of augmented samples generated for each original instance depends on the class, the smaller the support of a class in the training set, the more new samples we generate for it. In the next section, we report the number for each augmented class. To complete the augmentation process, the augmented sentences were translated back into Italian.

The second augmentation strategy is based on the generation of new samples using a Large Language Model, specifically *Meta LLaMA 3.2-3b-Instruct*. We provide the model with both a system prompt and a user prompt. The system prompt contains the instructions for the LLM on the expected behavior, output format and what to avoid, while the user prompt contains the original text to be augmented.

Data splitting was handled differently for the two tasks. For argument classification, we used a standard random split into training, validation, and test sets. For scheme classification, instead, we applied a stratified strategy: first, we separated documents linked to rare classes, then we split the remaining documents with a standard approach, and finally split the separated documents and added to the sets. In this way, the split is no longer completely random, but it ensures that rare classes are represented in all sets, which in our case significantly improved the results.

The models were trained on the training set, tuned on the validation set, and evaluated on the test set. For the Argument Classification task we measured the macro-F1 score. Instead, for scheme classification, given the imbalance between classes and our attempt to mitigate the issue, we used macro-F1 score to measure performance across all classes, treating them equally, and also micro-F1 score was used to check overall performance of the model.

4 Data

The Italian VAT dataset, which we used for this work, includes for each argumentation annotations about the role of sentences (Premise or Conclusion, useful for Argument Classification tasks) and for premise sentences includes also annotations about its argumentation scheme (useful for Scheme Classification tasks). This dataset uses the following argumentation schemes:[5][1]

- Rule: Argument from an established rule

- Prec: Argument from precedent
- Itpr: Argument from interpretation
- Princ: Argumentation from principle
- Class: Argument from verbal Classification
- Aut: Authoritative argument
- Syst: Argument from systematic interpretation
- Tele: Teleological argument
- Lit: Argument from literal interpretation
- Psy: Argumentation from intention of the legislator

Some of these schemes had such low support in our dataset that their use for training would have been prohibitive, even using data augmentation (see Figure 1). For this reason, as part of data preprocessing we used an arbitrary cut-off threshold of 15 samples and ignored all schemes with fewer samples.

As part of data pre-processing we also dropped phrases which were not as tagged neither as premises nor as conclusions or with a NaN content in the text or argumentation scheme. We also replaced multiple whitespaces with a single space.

Then we encoded the dataset for each of the two tasks separately. For Argument Classification, which is a binary classification task, we used a label encoder. For Scheme Classification, which is a multi-class classification task, we used a multi-label binarizer, creating one binary field for each scheme. Not only this is necessary for multi-class classification, it also allows us to encode the cases where a premise is labeled with multiple schemes.

After data preprocessing we also applied the aforementioned data augmentation techniques in selected runs.

Finally, data was split for training, validation and test sets as described above.

5 Experimental setup and results

All transformer-based experiments used Italian-Legal-BERT and its corresponding tokenizer, both loaded from the Hugging Face library. For augmentation, we considered only the classes *Tele*, *Syst*, *Aut*, *Class* and *Princ*, generating 6, 5, 4, 3 and 1 new samples per original instance, respectively. For LLM based augmentation, we used *Llama 3.2-3b-Instruct* variant with the *transformers* library, loading the model directly on Kaggle.

First, we selected three random seeds (*27*, *42* and *777*) and ran each model with each one of them, fixing the random seed for reproducibility and to ensure that the results were stable and not influenced by random initialization. All the runs were executed on *Kaggle*.

For the argument classification task, after performing a random split of the dataset into training, validation, and test sets, we first evaluated the SVM algorithm with a linear kernel using TF–IDF embeddings as a baseline. We then trained *Italian-Legal-BERT* for 8 epochs using the Adam optimizer, comparing training with binary cross-entropy (BCE) and focal loss. During training, we monitored not only the validation loss but also the macro-F1 score on the validation set to better capture performance on minority classes.

For the scheme classification task, which is highly imbalanced and multi-label, we applied stratified splitting strategy. As a baseline, we fitted a SVM model with TF–IDF embeddings. We then fine-tuned *Italian-Legal-BERT* for 8 epochs with the Adam optimizer, first with BCE loss, then with focal loss, again monitoring macro-F1 on the validation set. To further tackle class imbalance, we then trained the model on back-translation augmented data combined with BCE loss (TNAUG+BCE), and finally on LLM augmented data with BCE loss.

After the training process, we evaluated the trained model on the test set by computing a complete classification report using the *classification_report* function from *scikit-learn*. This report provides not only the overall F1-score, but also precision and recall for each individual class, alongside their support. In this way, it is possible to understand not only the global performance of the model, but also how it behaves across different classes, especially the rare ones.

The results (average F1 micro and F1 macro scores) of the executions can be found in Table 1, instead the complete results with the different seeds can be found in figure 2.

6 Discussion

As shown in the Table 1, in the AC task, the BERT model outperforms the baseline. Both BCE and focal loss work well, with focal loss performing

Task	Baseline		BCE		FocalLoss		TNAUG+BCE		LLM+BCE	
	micro	macro	micro	macro	micro	macro	micro	macro	micro	macro
AC	/	0.87	/	0.89	/	**0.9**	/	/	/	/
SC	0.64	0.33	0.72	0.35	0.71	0.44	0.73	0.45	**0.72**	**0.49**

Table 1: Average score over the 3 seeds for AC and SC

slightly better.

For the SC task, we decided to report both micro-F1 and macro-F1 to better understand the model's performance. The first observation is that results are highly sensitive to the choice of seed, meaning that different data splits can lead to significantly different outcomes. The focal loss variant outperforms both the baseline and BCE loss, highlighting its effectiveness on underrepresented classes. Additionally, both augmentation techniques improve macro-F1, confirming their benefit for handling rare classes.

Regarding the quality of the generated data, in the translation based approach the new instances were usually very similar to the originals. In contrast, LLM based generations were more diverse (we used a temperature of 0.5). In some cases, the samples preserved the context information and were reformulated successfully, while in others we observed issues such as the multiple repetition of a single fragment from the original text or very short outputs consisting of only a few words. We tried to mitigate these problems by specifying in the system prompt both what the LLM should produce and what it should avoid, but the model often introduced undesired artifacts, such as multiple consecutive spaces or unmatched parentheses, despite explicit instructions. Of course, the LLM we used was a small variant and was not specifically fine-tuned for legal text. However, our goal was to explore whether lightweight models could still produce useful synthetic data.

7 Conclusion

In AC task, the results are quite satisfactory, though they could be improved with data augmentation techniques. Instead the SC task turned out to be particularly challenging due to the strong imbalance, its multi-label nature and the very limited support for minority classes, which makes augmenting underrepresented classes more challenging, and also because often these samples also belong to one of the majority classes. The use of techniques such as focal loss helped to improve performance, and generating synthetic data with LLMs proved to be a promising direction, even though the model we used was not fine-tuned on legal text.

Future work could focus on using bigger LLMs, that are more likely to produce better and diverse text. Another option could be to leverage LLMs fine-tuned on legal text, or even coupling LLM with a RAG (Retrieval Augmented Generation) based system that integrates a legal knowledge base to ensure more accurate and domain specific augmentation. Other possibilities could be to implement a threshold optimization process for each class, which may lead to better performance.

8 Links to external resources

- Project code: github.com
- The phrasing is corrected using ChatGPT (openai.com)

References

[1] ADELE-project. Annotation guidelines. pages 43–46.

[2] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. 2019. Learning imbalanced datasets with label-distribution-aware margin loss.

[3] Junya Chen, Zidi Xiu, Benjamin Goldstein, Ricardo Henao, Lawrence Carin, and Chenyang Tao. 2021. Supercharging imbalanced data learning with energy-based contrastive representation transfer. In *Advances in Neural Information Processing Systems*, volume 34, pages 21229–21243. Curran Associates, Inc.

[4] Federico Galli, Giulia Grundler, Alessia Fidelangeli, Andrea Galassi, Francesca Lagioia, Elena Palmieri, Federico Ruggeri, Giovanni Sartor, and Paolo Torroni. 2022. Predicting outcomes of italian vat decisions1. In *Legal Knowledge and Information Systems*, volume 362 of *Frontiers in Artificial Intelligence and Applications*, pages 188–193.

[5] Giulia Grundler, Piera Santin, Andrea Galassi, Federico Galli, Francesco Godano, Francesca Lagioia, Elena Palmieri, Federico Ruggeri, Giovanni Sartor, and Paolo Torroni. 2022. Detecting arguments in CJEU decisions on fiscal state aid. In *Proceedings of the 9th Workshop on Argument Mining*, pages 143–157, Online and in Gyeongju, Republic of Korea. International Conference on Computational Linguistics.

[6] Sophie Henning, William Beluch, Alexander Fraser, and Annemarie Friedrich. 2023. A survey of methods for addressing class imbalance in deep-learning based natural language processing. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 523–540, Dubrovnik, Croatia. Association for Computational Linguistics.

[7] Xiaoya Li, Xiaofei Sun, Yuxian Meng, Junjun Liang, Fei Wu, and Jiwei Li. 2020. Dice loss for data-imbalanced nlp tasks.

[8] Daniele Licari and Giovanni Comandè. 2024. Italian-legal-bert models for improving natural language processing tasks in the italian legal domain. *Computer Law & Security Review*, 52:105908.

[9] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2018. Focal loss for dense object detection.

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

[11] Douglas Walton, Christopher Reed, and Fabrizio Macagno. 2008. *Argumentation Schemes*. Cambridge University Press.

[12] Yiru Wang, Weihao Gan, Jie Yang, Wei Wu, and Junjie Yan. 2019. Dynamic curriculum learning for imbalanced data classification.

[13] Jason Wei. 2021. Good-enough example extrapolation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5923–5929, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

[14] Jiale Wei, Qiyuan Chen, Pai Peng, Benjamin Guedj, and Le Li. 2022. Reprint: a randomized extrapolation based on principal components for data augmentation. *ArXiv*, abs/2204.12024.

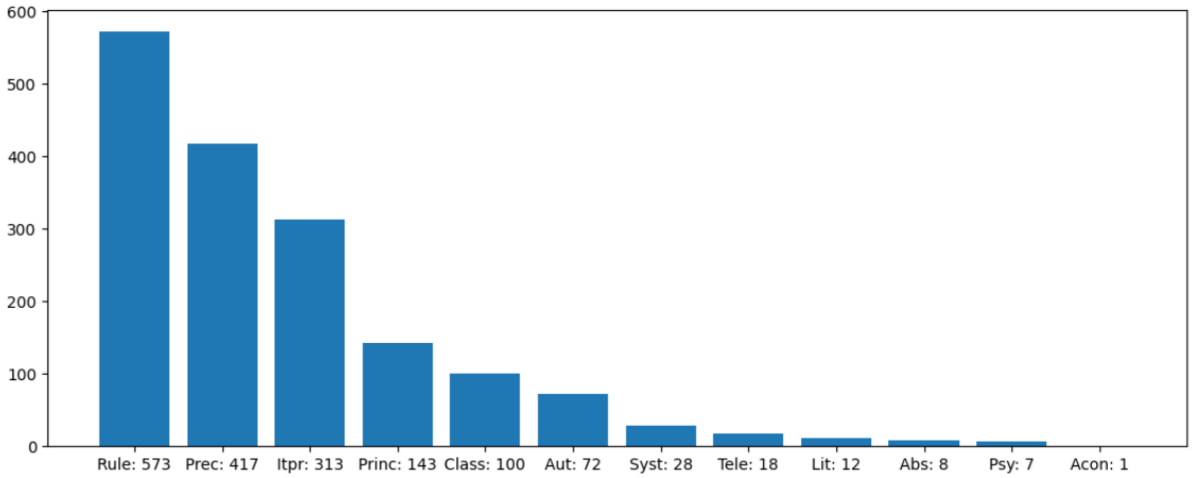

Figure 1: Scheme support in the dataset

Seed	Task	Baseline		BCE		FocalLoss		TNAUG+BCE		LLM+BCE	
		F1 micro	F1 macro	F1 micro	F1 macro	F1 micro	F1 Macro	F1 micro	F1 macro	F1 micro	F1 macro
27	AC	/	0,86	/	0,89	/	0,9	/	/	/	/
27	SC	0,7	0,45	0,75	0,31	0,72	0,53	0,76	0,59	0,77	0,6
42	AC	/	0,89	/	0,91	/	0,92	/	/	/	/
42	SC	0,65	0,32	0,73	0,4	0,72	0,42	0,74	0,43	0,71	0,44
777	AC	/	0,86	/	0,89	/	0,9	/	/	/	/
777	SC	0,58	0,24	0,7	0,35	0,71	0,37	0,71	0,34	0,7	0,44
Average	AC	/	0,87	/	0,89	/	0,9	/	/	/	/
Average	SC	0,64	0,33	0,72	0,35	0,71	0,44	0,73	0,45	0,72	0,49

Figure 2: Full results table