Assignment 2

Mauro Dore, Giacomo Gaiani, Gian Mario Marongiu and **Riccardo Murgia**
Master's Degree in Artificial Intelligence, University of Bologna
{ mauro.dore, giacomo.gaiani, gianmario.marongiu, riccardo.murgia2 }@studio.unibo.it

Abstract

This report analyses the behavior of three different models and their performance on solving the human values classification problem (Kiesel et al., 2022). Moreover, two additional simple models are used as baselines for the study, a random and a majority classifier. The tests included a performance comparison of the models, an analysis of how the presence of various information about the input arguments contributed to the F1-score of the models and what are the most common prediction errors on the validation and test set of the best performing model.

1 Introduction

The human values detection problem is relatively recent. The most common proposed approaches that tackle this task rely on neural networks based on pre-trained Bert transformers. The intrinsic difficulty of this task, despite the approach, relates to the fact that values in arguments are often subtle, and their own definition may be vague (Kiesel et al., 2022). The approach described in this report is divided into the following main steps:

1. Pre-processing and tokenization: The datasets are organized into data frames, and the labels are mapped from level two categories up to level three. During this phase, data loaders are created by taking the row data, which is internally tokenized by the same function according to the different versions of Bert, which require different tokenization.

2. Creation of the models: two baselines are created for performance comparison: a random uniform classifier (RC) and a majority classifier (MC). The three main models are neural network classifiers based on BERT.

3. evaluation: the BERT-based models are fine-tuned and tested on the test set using three different seeds. This helps us make sure the results can be repeated and gives us a reliable average F1-score and standard deviation. Then, we compared the results from the baselines with those from the more advanced models.

2 System description

RC is a random uniform classifier that predicts the labels of a given input in a random way. MC is a majority classifier, which returns a prediction according to the most common class for each level-3 category. The BERT-based models take in input one or more parts of an argument and return in output a prediction for each of the four possible labels the argument may belong to. They are defined as follows:

1. Bert w/C: takes in input the conclusion of an argument. This conclusion passes through a Bert layer. The output of this layer is sent to a linear classifier.

2. Bert w/CP: takes in input conclusion and premise of an argument. These inputs pass through a Bert layer, whose outputs are concatenated and sent to a linear classifier.

3. Bert w/CPS: takes in input the conclusion, premise, and stance of an argument. As it was for Bert w/CP, the conclusion and premise are the inputs of a single Bert layer, whose output is concatenated with the Stance value (encoded as 0 or 1). This final tensor is eventually sent to a linear classifier.

All the models contain a dropout layer between the BERT layer and the linear layer. The linear layer is composed of four neurons (one for each category), and their output is fed to a Sigmoid function.

Model	F1_val	F1_test	std_deviation
RC	/	0,483	/
MC	/	0,372	/
Bert w/C	0,625	0,568	0,016
Bert w/CP	0,720	0,680	0,004
Bert w/CPS	0,723	0,681	0,006

Table 1: F1 macro score on evaluation and test set. The standard deviation is computed on the validation set averaging over 3 seeds.

3 Experimental setup and results

The performances of the models on the validation and test set are shown in (1).

The performance of every model is computed by calculating the average (macro) of the binary per-category F1-score. These models were trained using the *torch.Adam* optimizer with the implementation of a custom early stopping. The optimal hyperparameter combination was primarily discovered through empirical testing. This involved conducting a grid search to identify a favorable set of values. We fixed the learning rate to $1e-5$, and the batch size to $32, 16, 16$ for w/C, w/CP, and w/CPS respectively. The grid search tested three different dropout probabilities $(0.0, 0.3, 0.5)$ along with two BERT versions from Hugging-Face: *bert-base-uncased* and *roberta-base*.

4 Discussion

In our evaluation, the random classifier achieves an average F1 score of approximately 0.48. The Majority Classifier, despite its higher accuracy, reports a lower F1 score with respect to the random classifier, due to the label distribution among the classes. In particular, classes with a majority of 0s report an F1 of 0.0. On the other hand, it obtains very good performance in those classes where the most common label is 1, with values of F1 up to 0.83 for Conservation and 0.65 for Self Transcendence. Because of this, the average value of the majority classifier F1 is 0.37. For the BERT-based models, all three versions (BERT w/C, BERT w/CP, and BERT w/CPS) significantly surpass the baseline models in average F1 scores. The BERT w/C model demonstrates a well-rounded performance improvement across all categories compared to the baseline models. However, it is the least effective among the BERT models. This might be due to the dataset containing various arguments with identical conclusions but different premises and stances. Since BERT w/C is trained solely on conclusions, it overlooks the diversity in arguments with the same conclusion, resulting in lower performance. Eventually, the BERT w/CP and w/CPS models showed improvements in terms of F1, increasing their classification capabilities particularly in "Openness to Change" and "Self Enhancement. These models showed the best overall performance, with an almost insignificant difference in their F1-score. The improvement showed by the BERT models suggests they are adept at identifying significant patterns in the data, unlike the random classifier that relies on guesswork.

5 Conclusion

We can consider Bert w/CPS as the best-performing model, which slightly outperforms Bert w/CP with an average F1-score on the test set of $0,681$. Their slight difference shows how, contrarily to our initial hypothesis, the Stance input in Bert w/CPS contributed less to the model's effectiveness than we had anticipated. This aspect is probably due to the fact that the models are classifying high-level human values, which mostly depend on the couple <conclusion-premise>. In this context, Stance is almost redundant data as its value is highly derivable from the relation between conclusion and premise. Considering possible future improvements of this model, including human values of level 4 in the training phase would lead to a higher accuracy in the classification, as already shown in (Kiesel et al., 2022). Another potentially effective approach consists of extending the model with a sentiment classifier. In fact, given the model's application in classifying human values, which are often influenced by emotional contexts, sentiment analysis could lead to a higher accuracy in the classification.

6 Links to external resources

The full code can be found at: https://github.com/GianM0027/NLP_homework_2

References

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.