CHILab at HaSpeeDe3: Overview of the Taks A Textual

Irene Siragusa[1], Roberto Pirrone[1]

[1]Dipartimento di Ingegneria @ Università degli Studi di Palermo, Viale delle Scienze, Edificio 6 90129 - Palermo

Abstract

This technical report illustrates the system developed by the CHILab team for the competition HaSpeeDe3 as part of the EVALITA 2023 campaign. The key idea for HaSpeeDe3 task A - Political Hate Speech Detection - Textual, was to develop different systems arranged as suitable combinations of the Pre-Trained Language Model (PTLM) used for embedding extraction, neural architectures for further elaborations over the embeddings and a classifier. In particular, dense layers, LSTM, BiLSTM and Transformers were used. The best performing system across the ones investigated in this report was made by embeddings extracted via XLM-RoBERTa coupled with BiLSTM that reaches a macro-F1 score of 0.876.

Keywords

hate speech detection, BiLSTM, language model

1. Introduction

The continuous spread and usage of social media has become a problem when dealing with hate online. All social platforms use artificial intelligence techniques to detect and report or remove some dangerous contents in terms of hate or violence. The interest in this respect is also high in the scientific community, in fact different international campaigns for detecting hateful speeches have been proposed in recent years: OffensEval [1, 2], HatEval [3], HaHackathon [4]. Detection of hateful content in Italian has been addressed by the HaSpeeDe evaluation competitions [5, 6].

This paper introduces the architecture proposed by the CHILab team for the EVALITA 2023 campaign [7], and in particular as regards the Hate Speech Detection task (HaSpeeDe3 task A - Political Hate Speech Detection, textual) [8]. The general approach relies on encoding the text into suitable word embeddings that are processed via neural architectures like LSTM, BiLSTM or Transformers. Finally, the output classifier detects the presence of hateful content.

We conceived our pipelines as "minimalist" architectures. No generative models [9, 10] where considered in this respect to derive embeddings. Moreover, we decided not to use fine-tuning in our PTLMs to stress the use of light networks to be trained with low computing resources. Finally, we set up a unique approach for all the tasks we have participated in EVALITA 2023.

The paper is arranged as follows: Section 2 reports a description of our systems along with data pre-processing, while results are reported and discussed in Section 3. Concluding remarks are in Section 4.

2. Description of the system

The focus of HaSpeeDe3 was on political and religious hate, where strong polarized opinions can be found. The data set used in this edition for task A is the PolicyCorpus XL [11] that contains 7000 tweets annotated manually, and a presence of hate labels above 40%. The training data set was released for the campaign with a total of 5600 samples: for developing purposes, the given data set was randomly split in a training and validation set, using a 80-20 ratio, resulting in 4480 and 1120 samples respectively.

2.1. Pre-processing

The `[URL]` tag, mention references, and retweet notes were removed since they were not considered meaningful: in particular, mentions are referred to anonymized accounts thus they add no special information. This was done after an analysis on the most cited words and hashtags[1]. As reported in Table 1, the `[URL]` tag is the most frequent one between classes and adds no information just like the anonymized mentions in the form @unknown. Overall, no other relevant words appeared that suggest a strong separation between classes. The same considerations can be done for the hashtags as reported in Table 2.

Although there are some hashtags that are hateful (such as *salvinipagliaccio*, *speranzadimettiti* and *governodeipeggiori*), the most frequent ones are just either politicians' or parties' names, and politics related words, that do not express any polarized content. Moreover, since a strong and significant distinction between hateful and non-hateful hashtags can be done, their information has been used as a word inside the tweet, thus keeping the crucial information, while the hashtag symbol was removed.

EVALITA 2023: 8th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian, Sep 7 – 8, Parma, IT

irene.siragusa02@unipa.it (I. Siragusa); roberto.pirrone@unipa.it (R. Pirrone)

[1] for this analysis all the words were reported in their lower case form

Table 1
Word distribution statistics over the dataset divided per label.

All tweets	freq	NH tweets	freq	H tweets	freq
url	1585	url	1007	url	578
governo	455	unknown	297	governo	160
solo	376	governo	295	solo	133
unknown	364	solo	243	797998657209770	132
fare	276	fare	178	salvini	116
fatto	264	fatto	173	fa	112
fa	256	oggi	165	sempre	104
cosa	254	essere	159	cosa	99
salvini	247	cosa	155	fare	98
essere	247	italia	152	poi	96

Table 2
Hashtags distribution statistics over the dataset divided per label.

All tweets	freq	NH tweets	freq	H tweets	freq
salvini	886	salvini	557	salvini	329
m5s	630	m5s	495	salvinisciacallo	251
conte	419	conte	316	salvinipagliaccio	235
draghi	368	legge	284	governodeipeggiori	223
lega	341	governo	255	speranzadimettiti	179
governo	320	draghi	252	speranzavattene	175
legge	309	lega	218	m5s	135
renzi	306	renzi	212	salviniportasfiga	131
salvinisciacallo	255	pd	181	lega	123
pd	251	politica	170	draghi	116

Table 3
Emoji distribution statistics over the dataset divided per label.

All	freq	NH	freq	H	freq
😂	38	🇮🇹	21	💩	32
💩	32	🤔	15	😂	27
🤣	32	🤣	14	🤡	25
🇮🇹	26	🔵	13	🤣	18
🤔	26	😂	11	😡	15
🤡	26	🔴	10	🤔	11
😡	22	‼️	8	🤮	10
🤮	15	😅	8	🤬	10
👇	15	👇	8	🙄	8
🔴	13	😡	7	👇	7

Similar considerations were made for emojis: also in this case, a strong polarization in the use of emojis did not arise, particularly for the ones that are more associated with disgust and hate (Table 3). Since emojis are deeply used in social media communication, they were kept. No further elaboration were made over the tweets: words were not reported to their lower case form, thus allowing a more accurate extraction of embeddings for the case-sensitive PTLMs. As for emojis, uppercase texts has a specific meaning in social media communication in terms of prosodic and emotions interpretation [12, 13].

2.2. Network architectures

Different models were developed that share the same macro structure shown in figure 1. The key idea was to stress, as much as possible, existent neural architectures for sequence processing, that are LSTM [14], BiLSTM and Transformers [15]. Those architecture are used to further process the extracted embeddings.

After pre-processing, the input sentences were padded to $maxLength + 2$ tokens where $maxLength$ is the size of the longest sentence, and the remaining two tokens are respectively the [CLS] and the [SEP] one. Either a pre-trained language model or a static context-free embedding model were used for embedding generation. In the last case, *fastText* [16] was used that generates a 300 tokens embedding, while a 768 tokens embedding is obtained as usual by the different PTLMs. We used the following Encoder-based Language Models in the experiments: BERT base multilingual cased [17], BERT base italian uncased [18], XLM-RoBERTa [19] and AlBERTo [20] provided by the HuggingFace Transformers library[2]. The embeddings were extracted from the last layer of the PTLMs without fine-tuning. Fine-tuning in these configuration is an option that is not taken into account since the main idea is to stress the use of light

[2] https://huggingface.co/docs/transformers/index

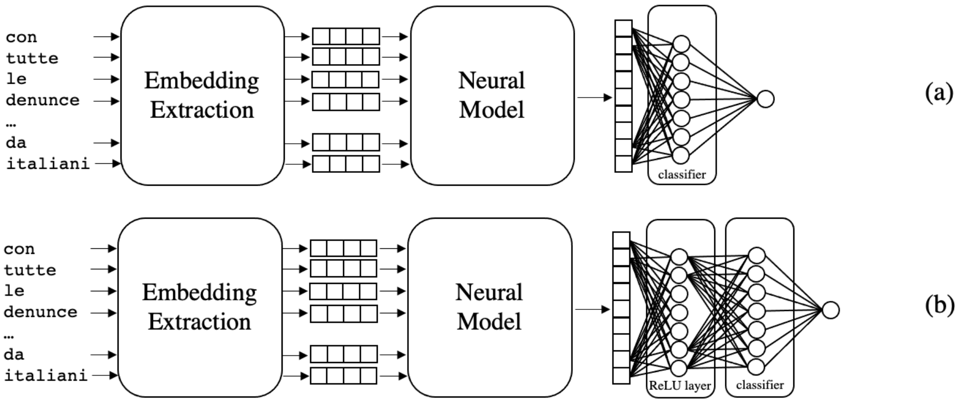

Figure 1: The first proposed architecture (a) has a module for embeddings extraction, a neural module for further processing on the extracted embeddings and a classifier. The second one (b) adds an additional ReLU dense layer.

networks to be trained with low computing resources.

The generated embedding is fed into a module for feature extraction that consists of a LSTM or a BiLSTM or a Transformer[3]. The output feature vector has the same size of the word embedding with the exception of the BiLSTM that generates a double-length output. Finally, the feature vector is passed to a classifier made by either 300 or 768 linear units, depending on the length of the embedding, and a sigmoidal output to achieve binary classification. Some experiments were run by inserting a ReLU dense layer before the aforementioned one with exactly the same size. Those architectures are referred as LSTM-Deep, BiLSTM-Deep and Trasformer-Deep (Figure 1.b).

The illustrated architectures were trained only on the given data set using a machine equipped with two Intel Xeon E5 CPUs 96GB RAM and an NVIDIA TITAN Xp GPU 12GB RAM. Hyperparameters were selected as follows: dropout values in $\{0.1, 0.2\}$, batch size 32, Adam optimizer [21] with learning rate 0.01, and a Binary Cross Entropy loss. Models were trained for a maximum of 1000 epochs with a patience value of 50.

Different feature extractors were implemented using 1, 2 or 3 LSTM/BiLSTM/Transformer layers, but the best results were obtained by the single layer feature extraction modules. In addition the developed models are relative small, where the trainable parameters range from 1M to 10M.

3. Results

The best F1-macro performances obtained on the test set from our models are reported in Table 4. The submitted modes were the best runs with respect to the validation set, namely AlBERTo/BiLSTM (run 1) and fastText/BiLSTM (run 2). After the release of the golden labels, it was possible to measure the actual performance of all the developed systems and this shows up that the XLM-RoBERTa/BiLSTM architecture gives the best results, ranking at the 7th place on the leaderboard, while the submitted runs are at last places as shown in Table 5.

Best results are obtained either when using a PTLM coupled with a LSTM/BiLSTM feature extractor and a single dense layer[4], while the Transformer based networks exploit better a context-free embedding by using a two layer classifier.

It is worth noticing that only the models that use *fastText* benefit from removing stopwords, while the PTLMs perform almost equally over LSTM and BiLSTM configurations as it was expected. In the training phase, AlBERTo outperformed the other PTLMs since it uses a more accurate tokenization compared to the others, and it takes advantage from its inner knowledge: AlBERTo was trained on a corpus of Italian tweets that share the same linguistic macro-structure of the PolicyCorpus. On the other hand, the best model is the one based on XLM-RoBERTa: this can be caused from its tokenizer that owns an inner representation for emojis, and consider them as unique tokens and not as `[UKN]`.

[3] The corresponding architectures are named according the specific neural module

[4] In table 4 some experiments and configurations are not reported, like the BiLSTM-Deep one, because they ran bad with respect to the submitted architectures.

Table 4
The table collects all the relevant results accross the developed architectures. Starred results are the ones submitted to the competition (run1 is BiLSTM model with AlBERTo and run2 is BiLSTM model with fastText), and the bold one is the highest score computed after the golden labels were released. Finally, the underlined results were obtained by removing stopwords from the data. XLM-RoBERTa, fastText and mBERT generate case-sentive embeddings.

	LSTM	LSTM-Deep	BiLSTM	Transformer-Deep
XLM RoBERTa	0.840	0.821	**0.876**	0.836
fastText	0.790	0.726	0.852*	0.861
AlBERTo	0.864	0.834	0.826*	0.852
mBERT	0.857	0.827	0.859	0.831
BERT-it	0.850	0.817	0.868	0.801

Table 5
The table collects the macro F1 results over the test set of the submitted models and the actual best developed model (the starred one). Result of the baseline model is also reported, along with the ranking and expected ranking position.

Run name	Macro F1	Rank
CHILab3*	0.876	7*
CHILab2	0.852	8
Baseline	0.846	9
CHILab1	0.826	10

3.1. Error analysis

Besides the aforementioned differences between the PTLMs used, another analysis was made on the mis-classified tweets by comparing the results of the best architectures (AlBERTo/LSTM, AlBERTo/LSTM-Deep, XLM-RoBERTa and fastText/Transformer-Deep) and the submitted models. Models agree in mis-classifing 32 tweets, and 25 of them are labeled as hateful.

None of these mis-classified tweets contain emoji, that is their presence or absence is not source of bias in those models. Moreover, the majority of those tweets contains hashtags or expressions referring to politicians and topics of interest in the political debate, that per se are not hateful. On the contrary, tweets containing the hashtag *speranzadimettiti*, considered hateful as in 2, can be found in non hateful tweets. In those tweets the author disapproves the governmental behaviour of a minister: in this case it cannot be considered as hateful since it express a negative opinion without insulting.

On the other side, hateful tweets usually contains profanities and vulgar expressions: hateful tweets that are not correctly classified by the developed models, lack of those expressions or put them in an unconventional way (self-obfuscation or embedded in other words) and this lead to their mis-classification.

4. Conclusion

This paper reported the architectures developed by the CHILab team for HaSpeeDe3 task A promoted at the EVALITA 2023 campaign. Our models show that a relatively small classical pipeline made by embedding extraction plus further neural elaboration can have good performance in hate speech detection without the need of fine-tuning PTLMs, and using few computational resources. The use of such “minimalist” architecture is intended to allow for future development of compact explainable models where explicit linguistic knowledge is injected in the network to improve its performance.

Acknowledgments

This work is supported by the PO FESR 2014-2020 grant n. 086201000543, “SCuSi - Smart Culture in Sicily”

References

[1] M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, R. Kumar, SemEval-2019 task 6: Identifying and categorizing offensive language in social media (OffensEval), in: Proceedings of the 13th International Workshop on Semantic Evaluation, Association for Computational Linguistics, Minneapolis, Minnesota, USA, 2019, pp. 75–86. URL: https://aclanthology.org/S19-2010. doi:10.18653/v1/S19-2010.

[2] M. Zampieri, P. Nakov, S. Rosenthal, P. Atanasova, G. Karadzhov, H. Mubarak, L. Derczynski, Z. Pitenis, c. Çöltekin, SemEval-2020 Task 12: Multilingual Offensive Language Identification in Social Media (OffensEval 2020), in: Proceedings of SemEval, 2020.

[3] V. Basile, C. Bosco, E. Fersini, D. Nozza, V. Patti, F. M. Rangel Pardo, P. Rosso, M. Sanguinetti, SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in Twitter, in: Proceedings of the 13th International Workshop on Semantic Evaluation, Association for Computational

Linguistics, Minneapolis, Minnesota, USA, 2019, pp. 54–63. URL: https://aclanthology.org/S19-2007. doi:10.18653/v1/S19-2007.

[4] J. A. Meaney, S. Wilson, L. Chiruzzo, A. Lopez, W. Magdy, SemEval 2021 task 7: HaHackathon, detecting and rating humor and offense, in: Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), Association for Computational Linguistics, Online, 2021, pp. 105–119. URL: https://aclanthology.org/2021.semeval-1.9. doi:10.18653/v1/2021.semeval-1.9.

[5] C. Bosco, D. Felice, F. Poletto, M. Sanguinetti, T. Maurizio, et al., Overview of the evalita 2018 hate speech detection task, in: Ceur workshop proceedings, volume 2263, CEUR, 2018, pp. 1–9.

[6] M. Sanguinetti, G. Comandini, E. D. Nuovo, S. Frenda, M. A. Stranisci, C. Bosco, T. Caselli, V. Patti, I. Russo, Haspeede 2 @ evalita2020: Overview of the evalita 2020 hate speech detection task, EVALITA Evaluation of NLP and Speech Tools for Italian - December 17th, 2020 (2020).

[7] M. Lai, S. Menini, M. Polignano, V. Russo, R. Sprugnoli, G. Venturi, Evalita 2023: Overview of the 8th evaluation campaign of natural language processing and speech tools for italian, in: Proceedings of the Eighth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2023), CEUR.org, Parma, Italy, 2023.

[8] M. Lai, F. Celli, A. Ramponi, S. Tonelli, C. Bosco, V. Patti, Haspeede3 at evalita 2023: Overview of the political and religious hate speech detection task, in: Proceedings of the Eighth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2023), CEUR.org, Parma, Italy, 2023.

[9] G. Mialon, R. Dessì, M. Lomeli, C. Nalmpantis, R. Pasunuru, R. Raileanu, B. Rozière, T. Schick, J. Dwivedi-Yu, A. Celikyilmaz, et al., Augmented language models: a survey, arXiv preprint arXiv:2302.07842 (2023).

[10] Y. Liu, T. Han, S. Ma, J. Zhang, Y. Yang, J. Tian, H. He, A. Li, M. He, Z. Liu, Z. Wu, D. Zhu, X. Li, N. Qiang, D. Shen, T. Liu, B. Ge, Summary of ChatGPT/GPT-4 Research and Perspective Towards the Future of Large Language Models, 2023. URL: http://arxiv.org/abs/2304.01852. doi:10.48550/arXiv.2304.01852, arXiv:2304.01852 [cs].

[11] F. Celli, M. Lai, A. Duzha, C. Bosco, V. Patti, Policycorpus xl: An italian corpus for the detection of hate speech against politics., in: CLiC-it, 2021.

[12] M. Heath, Orthography in social media: Pragmatic and prosodic interpretations of caps lock, Proceedings of the Linguistic Society of America 3 (2018) 55–1–13. URL: https://journals.linguisticsociety.org/proceedings/index.php/PLSA/article/view/4350. doi:10.3765/plsa.v3i1.4350.

[13] S. Chan, A. Fyshe, Social and emotional correlates of capitalization on Twitter, in: Proceedings of the Second Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media, Association for Computational Linguistics, New Orleans, Louisiana, USA, 2018, pp. 10–15. URL: https://aclanthology.org/W18-1102. doi:10.18653/v1/W18-1102.

[14] S. Hochreiter, J. Schmidhuber, Long Short-Term Memory, Neural Computation 9 (1997) 1735–1780. URL: https://doi.org/10.1162/neco.1997.9.8.1735. doi:10.1162/neco.1997.9.8.1735. arXiv:https://direct.mit.edu/neco/article-pdf/9/8/1735/813796/neco.1997.9.8.1735.pdf.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in neural information processing systems 30 (2017).

[16] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, T. Mikolov, Learning word vectors for 157 languages, in: Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018), 2018.

[17] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, CoRR abs/1810.04805 (2018). URL: http://arxiv.org/abs/1810.04805. arXiv:1810.04805.

[18] S. Schweter, Italian bert and electra models, 2020. URL: https://doi.org/10.5281/zenodo.4263142. doi:10.5281/zenodo.4263142.

[19] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, CoRR abs/1911.02116 (2019). URL: http://arxiv.org/abs/1911.02116. arXiv:1911.02116.

[20] M. Polignano, P. Basile, M. de Gemmis, G. Semeraro, V. Basile, AlBERTo: Italian BERT Language Understanding Model for NLP Challenging Tasks Based on Tweets, in: Proceedings of the Sixth Italian Conference on Computational Linguistics (CLiC-it 2019), volume 2481, CEUR, 2019. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85074851349&partnerID=40&md5=7abed946e06f76b3825ae5e294ffac14.

[21] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980 (2014).