Proceedings of the 1st Workshop on Perspectivist Approaches to NLP @LREC2022, pages 117–125
Marseille, 20 June 2022

Change My Mind: how Syntax-based Hate Speech Recognizer can Uncover Hidden Motivations based on Different Viewpoints

Michele Mastromattei*, **Valerio Basile**†, **Fabio Massimo Zanzotto***
* Department of Enterprise Engineering, University of Rome Tor Vergata, Italy
† Department of Computer Science, University of Turin, Italy
michele.mastromattei@uniroma2.it, valerio.basile@unito.it, fabio.massimo.zanzotto@uniroma2.it

Abstract

Hate speech recognizers may mislabel sentences by not considering the different opinions that society has on selected topics. In this paper, we show how explainable machine learning models based on syntax can help to understand the motivations that induce a sentence to be offensive to a certain demographic group. To explore this hypothesis, we use several syntax-based neural networks, which are equipped with syntax heat analysis trees used as a post-hoc explanation of the classifications and a dataset annotated by two different groups having dissimilar cultural backgrounds. Using particular *contrasting trees*, we compared the results and showed the differences. The results show how the keywords that make a sentence offensive depend on the cultural background of the annotators and how this differs in different fields. In addition, the syntactic activations show how even the sub-trees are very relevant in the classification phase.

Keywords: Hate speech recognizer, Explainable models, Perspectivism

1. Introduction

Hate speech recognizers (HSRs) (Warner and Hirschberg, 2012; Djuric et al., 2015; Gambäck and Sikdar, 2017) can be a great tool to contrast offensive terms, limit negative debates, and protect ethnic minorities. Indeed, these recognizers are excellent for spotting sentences containing offensive words as, over the years, several datasets focussing on this phenomenon have been released and used as training models (Waseem and Hovy, 2016; Davidson et al., 2017; Founta et al., 2018; Basile et al., 2019). However, these standardized datasets focus purely on offensive terms and indicate sentences as *hate speech* only because they contain words typically labeled as hate. This becomes a serious problem as hate speech recognition is done only focusing on *trigger words*. The context where sentences are written is disregarded as well as the addressees of the messages in these sentences.

Anyway, focusing on trigger words, HSRs increase the probability of tagging sentences from dialects of specific ethnic communities as hate speech. The result is that users who should be protected may risk banning (Sap et al., 2019). This is because some words are not offensive to some groups of people with particular ethnic backgrounds. On the contrary, the use of apparently inoffensive words can have a huge offensive impact on the society of other ethnic groups. So, the problem of hate speech and automatic hate speech detectors cannot be summarized in the classification of offensive elements but has a broader impact that includes who reads those sentences and how they are written.

Word-based and transformer-based models have a debias problem that is difficult to mitigate or easy to escape (Hosseini et al., 2017), and a typical solution is to use regularization techniques as in the case of transformers, by fashioning their attention mechanism (Kennedy et al., 2020). Although, attention seems to capture syntactic information (Eriguchi et al., 2016; Chen et al., 2018; Strubell et al., 2018; Clark et al., 2019), it is not clear how these regularizations reduce the use of trigger words.

In this paper, we want to find out - through syntactic models - what are the substructures that make a sentence labelable as hate speech, comparing explainable models trained on the same dataset but labeled by groups of people with different backgrounds. Our results show how the hate speech phenomenon is quite subjective and how the underlying motivations are different according to the cultural background of the annotators.

2. Background and related works

Methods to improve the interpretability of the predictions of supervised machine learning models and deep learning models are generally found in the literature around Explainable AI (XAI) (Samek et al., 2017; Samek and Müller, 2019; Vilone and Longo, 2020). For text classification tasks such as sentiment analysis or hate speech detection, methods have been proposed that work at the lexical level (Clos et al., 2017) or by highlighting subsequences of text that contribute to the final label (Perikos et al., 2021). Most modern models of neural interpretability rely on attention-based techniques (Bodria et al., 2020), using auxiliary tasks such as Aspect-Based Sentiment Analysis for Document-Level Sentiment Analysis interpretability (Silveira et al., 2019), or external knowledge (Zhao and Yu, 2021). While it has been postulated that attention-based models learn syntactic structure to a certain degree (Manning et al., 2020), the role of syntax in the interpretation of the model is still understudied, as opposed to classification (Cignarella et al., 2020).

The massive use of syntax, defined as heat parse trees and used as a post-hoc explanation of the classification, has shown that in the hate speech phenomena, syntax alone is not able to distill the prejudice because it is already intrinsic in the most common hate speech training corpora and so, syntax cannot drive the "attention" of hate speech recognizer to ethically-unbiased features (Mastromattei et al., 2022).

The potential impact of a perspectivist approach towards improving the interpretability of supervised Linear Programming (LP) models has been explored by Basile (2021). In the cited paper, the author proposed a simple method to derive a description of the different perspectives taken by the annotators of a hate speech corpus in the form of bags of words. Fell et al. (2021) further pursue this direction by proposing a method to cluster annotators, providing at the same time word clouds highlighting the terms that trigger a sensible response by different groups of people.

3. Methods and data

To explore perspectivism in explainability models we used the following steps: 1) a structured dataset having polarized labels (Sec. 3.1), 2) two or more explainable models (Sec. 3.2) and 3) an algorithm that explains how to analyze an outcome according to two different viewpoints and how these viewpoints conflicting (Sec. 3.3).

3.1. Brexit Hate Speech Dataset

To validate our method, we tested it on real-world data annotated with hate speech and several other phenomena. We selected the dataset by Akhtar et al. (2021), a corpus of 1,120 English posts from Twitter. The dataset was originally gathered for research on stance detection (Lai et al., 2019), and it has been further annotated with four binary labels: hate speech, (presence of) stereotype, aggressiveness, and offensiveness, adapting the guidelines used for the annotation of the Italian Hate Speech Corpus (Sanguinetti et al., 2018). Interestingly for our work, the Brexit dataset is annotated in its entirety by six different annotators belonging to two distinct social groups. The *target* group is composed of three Muslim immigrants in the United Kingdom, while the *control* group is composed of three Ph.D. students with western backgrounds. The inter-annotator agreement computed on the two groups separately shows that each group is fairly consistent internally (a high intra-group agreement) across all four dimensions, while they agree much less between members of different groups (low inter-group agreement). Using only the hate speech label, the inter-annotator agreement for both groups is a *Fair agreement*, employing the Fleiss' kappa measure.

3.2. Explainable Syntax-based models

Model interpretability is crucial in the study of divisive topics because it increases the trust that humans place in models and also for its fair and ethical decision-making. Especially, in the text-classification task, explainable syntax-based models return syntactic structures that are ideal for understanding sentence labeling and analyzing the substructures that influenced that target.

For this purpose, we used KERMIT (Zanzotto et al., 2020) and KERM-HATE (Mastromattei et al., 2022): two explainable syntax-based models that return heat-colored parse trees according to the values of activation of the model during the evaluation phase. Both models are based on the same components: a KERMIT component (that allows the encoding and the visualization of the activations of universal syntactic interpretations in a neural network architecture) and a transformer model. KERM-HATE differs from KERMIT only for a four-layer fully-connected neural network at the top of the model. KERMIT*viz* (Zanzotto et al., 2020), makes the KERMIT component the most relevant part of the two models. KERMIT*viz* gives the possibility to extract as output not only the classification target but especially the colored parse tree with the activation value of every single node that composes a generic sentence. Thus, KERMIT*viz* allows us to visualize how decisions are made according to activations of syntactic structures.

3.3. Contrasting trees

Using KERMIT and KERM-HATE (Sec.3.2), it is possible to study perspectivism through syntax trees. Given two equal KERMIT-like models and a sentence $\mathcal{S}$, it is possible to derive a syntactic tree (*constrasting tree*) whose activation values are the result of the difference between the activation values of the two models. The final result should be displayed using KERMIT*viz*. In this way, it is visible which are - after the two trees and their activations - the most active subparts and which are the syntactic structures that influence the classification of a sentence for a given model. This analysis is important to understand how salient are the syntactic substructures of a sentence and how they affect the final classification.

To generate a contrasting tree, we used the following method: let $\mathcal{T}_A =< \bar{T}, \bar{V}_A >$ and $\mathcal{T}_B =< \bar{T}, \bar{V}_B >$ two trees obtained from the same sentence $\mathcal{S}$ such that: $\bar{T} = \{\bar{t}_i, ..., \bar{t}_n\}$ is the ordered list of non-empty subtrees that makes up $\mathcal{T}_i$ (with $i = \{A, B\}$) and $\bar{V}_i = \{\bar{v}_{i,1}, ..., \bar{v}_{i,n}\}$ is the list of activation values where $\bar{v}_{i,j}$ is the activation value of subtree $\bar{t}_j$ (with $1 \leq j \leq n$). Thus the contrast tree $\mathcal{T}_{i-k} =< \bar{T}, \bar{V}_{i-k} >$ is obtained from $\mathcal{T}_i - \mathcal{T}_k$ and so $\bar{V}_{i-k} = \{(\bar{v}_{i,1} - \bar{v}_{k,1}), ..., (\bar{v}_{i,n} - \bar{v}_{k,n})\}$ (with $i, k = \{A, B\}$ and $i \neq k$).

In this way $\bar{V}_{i-k}$ contains only the relevant activations $\mathcal{T}_i$ because if $\bar{v}_{i,j} \approx \bar{v}_{k,j} \Rightarrow (\bar{v}_{i,j} - \bar{v}_{k,j}) \approx 0$, while if $\bar{v}_{k,j} >> \bar{v}_{i,j}$ then the result is a negative value and so a *zero activation value*.

4. Experiments

This section describes all the parameters and pretrained models used during our analysis (Sec. 4.1). Finally in

Sec 4.2 all obtained final results are shown and analyzed.

4.1. Experimental set-up

We tested our dataset using several models according to Mastromattei et al. (2022) tests: two *transformer-based* models and three *syntax-based* models. The two transformer-based model are Bert (Devlin et al., 2018) and XLNet (Yang et al., 2019) while the syntax-based are: KERM-HATE (Mastromattei et al., 2022), KERMIT (Zanzotto et al., 2020) and a modified version of KERMIT called KERMIT$_{XLNet}$ in which the original transformer sub-network has been replaced with XLNet. In this way, it is easier to visualize and compare all the models presented because for each transformer-based model, the syntax-based one was also generated. To assess statistical significance, each experiment was repeated 10 times with different seeds for initial weights. The meta-parameters utilized in training phrase are the following: for the syntax-based models (KERM-HATE, KERMIT and KERMIT$_{XLNet}$): (1) the tree encoder is on a distributed representation space R^d with $d = 4000$ and has penalizing factor $\lambda = 0.4$ (Moschitti, 2006); (2) constituency parse trees have been obtained by using Stanford's CoreNLP probabilistic context-free grammar parser (Manning et al., 2014). KERM-HATE's fully-connected four-layers network change the representation space four times: $R^n \rightarrow R^m \rightarrow R^n \rightarrow R^m$ where $m = 2,000$ and $n = 4,000$, before concluding with the final classification layer. (3) the decoder layer is a fully connected layer with the ReLU activation function (Agarap, 2018) applied to the concatenation of the KERMIT sub-network output and the final `[CLS]` token representation of the transformer model. Bert and XLNet model but also the transformer sub-networks component in the syntax-based models were implemented using Huggingface's transformers library (Wolf et al., 2019). For all models, the class weight w_i is inversely proposional to its $class_i$ (C_i) cardinality ($w_i = \frac{1}{|C_i|}$) and the optimizer used is AdamW (Loshchilov and Hutter, 2019) with the learning rate set to $2e^{-5}$. All models used a batch size of 64 and are trained for 3 epochs. The dataset described in Sec. 3.1 was divided into 80% for training and a 20% for testing. The two output datasets were used in the training and testing phase for all models used. Our hardware system consists of: 4 Cores Intel Xeon E3-1230 CPU with 62 Gb of RAM and 1 Nvidia 1070 GPU with 8Gb of onboard memory.

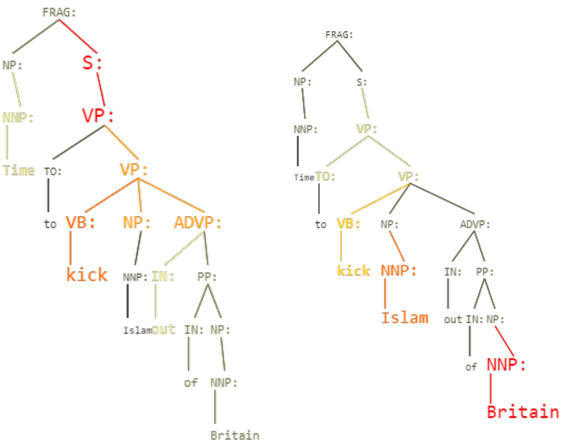
To generate contrasting trees, we used the algorithm described in Sec. 3.3 and - using KERMIT*viz* - showed the final results.

4.2. Result and discussion

In Table 1 we show the results in the testing phase of the five analyzed models. As it can be observed, KERM-HATE and KERMIT result to be the best models obtaining higher performances than the other models. It is important to note that the dataset is strongly unbalanced in favor of the *"no hate speech"* class. For this reason, in order to calm the results obtained and to continue the analysis, we use as visualization model KERMIT and not KERM-HATE, which has lower performances in the F1-measure *"hate speech"* class than KERMIT.

In Figure 1 we graphically show the output of KERMIT, trained using both *control group* (KERMIT$_C$) and *target group* (KERMIT$_T$) labels on the same sentence: *"Time to kick Islam out of Britain"*.

Sentence: *Time to kick Islam out of Britain*



(a) Labeled as **hate speech** for the model trained using the *control group* labels (KERMIT$_C$)

(b) Labeled as **hate speech** for the model trained using the *target group* labels (KERMIT$_T$)

Figure 1: KERMIT colored parse trees output

We can observe that the output of KERMIT$_C$ is composed of subtrees that are much more active than those of KERMIT$_T$, which concentrates on its leaves. If we analyze each tree individually, we discover that the label *"hate speech"* for KERMIT$_C$ (Figure 1a) is generated by the leaf *"time"*, its parent node and by the right subtree of depth 4. KERMIT$_T$, on the other hand, although it has the same label (*"hate speech"*), concentrates more on terminals and on hate/racial keywords, such as *"kick"* and *"Islam"*, but also on *"Britain"* (Figure 1b).

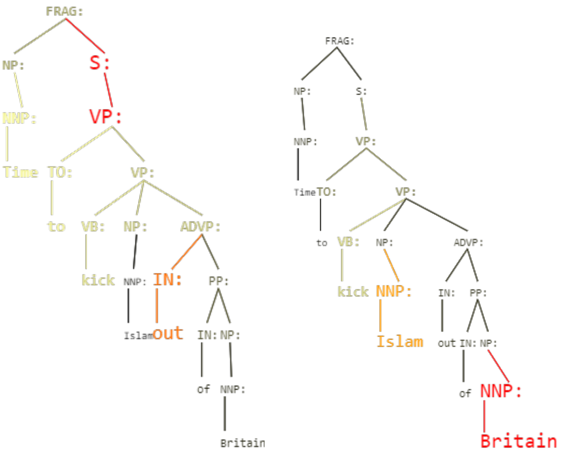
Using these trees, we created their *contrasting trees* to visualize which are the sub-structures keys in KERMIT$_C$ and KERMIT$_T$ excluding the similar activations in both models (Figure 2). The result obtained confirms our analysis done previously on the individual trees (Figure 1) and adds further details. In particular, even if some sub-structures result to be unaltered, in Figure 2a we have a prevalence of active non-terminal nodes compared to Figure 2b which instead continues to concentrate on leaf nodes.

This analysis does not show an isolated case. We performed a *quantitative analysis* of the data by analyzing over 8,600 subtrees from several sentences within

Model	*Control* group			*Target* group		
	Accuracy	**F1 measure**		**Accuracy**	**F1 measure**	
		Macro	**Weighted**		**Macro**	**Weighted**
Bert	0.61 (± 0.33)$^{\diamond}$	0.38 (± 0.15)$^{\diamond}$	0.62 (± 0.31)$^{\diamond}$	0.45 (± 0.16)$^{\diamond}$	0.39 (± 0.19)$^{\diamond}$	0.27 (± 0.36)$^{\diamond}$
XLNet	0.70 (± 0.29)$^{\dagger,\bullet}$	0.42 (± 0.14)$^{\dagger,\bullet}$	0.70 (± 0.28)$^{\dagger,\bullet}$	0.53 (± 0.22)	0.37 (± 0.12)	0.45 (± 0.25)
KERM-HATE	**0.92 (± 0.01)**$^{\diamond,\dagger,*}$	**0.49 (± 0.04)**$^{\diamond,\dagger}$	**0.88 (± 0.08)**$^{\diamond,\dagger,*}$	**0.64 (± 0.11)**$^{\diamond,\triangleleft}$	**0.48 (± 0.05)**$^{\diamond,*}$	**0.61 (± 0.08)**$^{\diamond,*}$
KERMIT	0.81 (± 0.13) $^{\bullet}$	**0.49 (± 0.04)**$^{\bullet}$	0.82 (± 0.08)$^{\bullet}$	0.55 (± 0.12)$^{\triangleleft}$	0.47 (± 0.05)	0.55 (± 0.10)
KERMIT$_{XLNet}$	0.31 (± 0.33)*	0.21 (± 0.19)	0.27 (± 0.36)*	0.56 (± 0.12)	0.46 (± 0.05)*	0.56 (± 0.09)*

Table 1: Performance of all model tested. Mean and standard deviation results are obtained from 10 runs. The symbols $\diamond$, $\dagger$, $*$, $\bullet$ and $\triangleleft$ indicate a statistically significant difference between two results with a 95% of confidence level with the sign test.

Sentence: *Time to kick Islam out of Britain*



(a) Tree obtained subtracting from KERMIT$_C$ activation values those of KERMIT$_T$

(b) Tree obtained subtracting from KERMIT$_T$ activation values those of KERMIT$_C$

Figure 2: Contrasting trees

the dataset. If the prediction was *"hate speech"* for both KERMIT$_C$ and KERMIT$_T$, then KERMIT$_T$ focuses predominantly on tree leaves (the depth of activated subtrees is approximately 1) while the activation of KERMIT$_C$ is more distributed along with the syntax trees, with the average depth of activated subtrees equal to 1.7.

For a more accurate view of other sentences and their activations, in Appendix A we show more examples where both KERMIT$_C$ and KERMIT$_T$ predict the same sentence as *"hate speech"* but also cases where the label between the two models differs (*"no hate speech"* - *"hate speech"*).

5. Conclusion

Hate speech recognizers (HSRs) typically label a sentence as offensive by counting only the number of trigger words. In this paper, we have shown how, using syntax-based explainable models and a dataset labeled by two groups with different backgrounds, it is possible to view the motivations that lead HSRs to classify a sentence in a certain way and how those motivations change. Using contrast trees, we show the salient points that make a sentence offensive for each group. In this way, we can understand the motivations that each group used, giving us a wider and less critical view of their thinking (*"Change my mind"*).

Performing a quantitative analysis of the dataset - we confirmed our hypothesis that sentence labeling depends on the cultural background of each annotator. This implies that the use of syntax is useful in the hate speech phenomena and that the use of common *hate speech corpora* as training datasets, does not include the different aspects of society on a theme so subjective as hate speech.

Acknowledgments

This work is partially funded by the 2019 BRIC INAIL ID32 SfidaNow project.

6. Bibliographical References

Agarap, A. F. (2018). Deep Learning using Rectified Linear Units (ReLU). *CoRR*, abs/1803.0.

Akhtar, S., Basile, V., and Patti, V. (2021). Whose opinions matter? perspective-aware models to identify opinions of hate speech victims in abusive language detection. *CoRR*, abs/2106.15896.

Basile, V., Bosco, C., Fersini, E., Nozza, D., Patti, V., Rangel Pardo, F. M., Rosso, P., and Sanguinetti, M. (2019). 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*, pages 54–63, Minneapolis, Minnesota, USA, June. Association for Computational Linguistics.

Basile, V. (2021). It's the end of the gold standard as we know it. In Matteo Baldoni et al., editors, *AIxIA 2020 – Advances in Artificial Intelligence*, pages 441–453, Cham. Springer International Publishing.

Bodria, F., Panisson, A., Perotti, A., and Piaggesi, S. (2020). Explainability methods for natural language processing: Applications to sentiment analysis. In Maristella Agosti, et al., editors, *Proceedings of the 28th Italian Symposium on Advanced Database Systems, Villasimius, Sud Sardegna, Italy (virtual due to Covid-19 pandemic), June 21-24, 2020*, volume 2646 of *CEUR Workshop Proceedings*, pages 100–107. CEUR-WS.org.

Chen, K., Wang, R., Utiyama, M., Sumita, E., and Zhao, T. (2018). Syntax-directed attention for neural machine translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Cignarella, A. T., Basile, V., Sanguinetti, M., Bosco, C., Rosso, P., and Benamara, F. (2020). Multilingual irony detection with dependency syntax and neural models. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1346–1358, Barcelona, Spain (Online), December. International Committee on Computational Linguistics.

Clark, K., Khandelwal, U., Levy, O., and Manning, C. D. (2019). What does bert look at? an analysis of bert's attention. *arXiv preprint arXiv:1906.04341*.

Clos, J., Wiratunga, N., and Massie, S. (2017). Towards explainable text classification by jointly learning lexicon and modifier terms. In *IJCAI-17 Workshop on Explainable AI (XAI)*, page 19.

Davidson, T., Warmsley, D., Macy, M., and Weber, I. (2017). Automated hate speech detection and the problem of offensive language. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11, pages 512–515.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Djuric, N., Zhou, J., Morris, R., Grbovic, M., Radosavljevic, V., and Bhamidipati, N. (2015). Hate speech detection with comment embeddings. In *Proceedings of the 24th international conference on world wide web*, pages 29–30.

Eriguchi, A., Hashimoto, K., and Tsuruoka, Y. (2016). Tree-to-sequence attentional neural machine translation. *arXiv preprint arXiv:1603.06075*.

Fell, M., Akhtar, S., and Basile, V. (2021). Mining annotator perspectives from hate speech corpora. In Elena Cabrio, et al., editors, *Proceedings of the Fifth Workshop on Natural Language for Artificial Intelligence (NL4AI 2021) co-located with 20th International Conference of the Italian Association for Artificial Intelligence (AI*IA 2021), Online event, November 29, 2021*, volume 3015 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Founta, A. M., Djouvas, C., Chatzakou, D., Leontiadis, I., Blackburn, J., Stringhini, G., Vakali, A., Sirivianos, M., and Kourtellis, N. (2018). Large scale crowdsourcing and characterization of twitter abusive behavior. In *Twelfth International AAAI Conference on Web and Social Media*.

Gambäck, B. and Sikdar, U. K. (2017). Using convolutional neural networks to classify hate-speech. In *Proceedings of the first workshop on abusive language online*, pages 85–90.

Hosseini, H., Kannan, S., Zhang, B., and Poovendran, R. (2017). Deceiving google's perspective api built for detecting toxic comments. *arXiv preprint arXiv:1702.08138*.

Kennedy, B., Jin, X., Mostafazadeh Davani, A., Dehghani, M., and Ren, X. (2020). Contextualizing hate speech classifiers with post-hoc explanation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5435–5442, Online, July. Association for Computational Linguistics.

Lai, M., Tambuscio, M., Patti, V., Ruffo, G., and Rosso, P. (2019). Stance polarity in political debates: A diachronic perspective of network homophily and conversations on twitter. *Data Knowledge Engineering*, 124:101738.

Loshchilov, I. and Hutter, F. (2019). Decoupled weight decay regularization. *7th International Conference on Learning Representations, ICLR 2019*.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The {Stanford} {CoreNLP} Natural Language Processing Toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Manning, C. D., Clark, K., Hewitt, J., Khandelwal, U., and Levy, O. (2020). Emergent linguistic structure in artificial neural networks trained by self-supervision. *Proceedings of the National Academy of Sciences*, 117(48):30046–30054.

Mastromattei, M., Ranaldi, L., Fallucchi, F., and Zanzotto, F. M. (2022). Syntax and prejudice: ethicallycharged biases of a syntax-based hate speech recognizer unveiled. *PeerJ Computer Science*, 8:e859.

Moschitti, A. (2006). Making tree kernels practical for natural language learning. In *11th conference of the European Chapter of the Association for Computational Linguistics*.

Perikos, I., Kardakis, S., and Hatzilygeroudis, I. (2021). Sentiment analysis using novel and interpretable architectures of Hidden Markov Models. *Knowledge-Based Systems*, 229:107332.

Samek, W. and Müller, K.-R. (2019). Towards explainable artificial intelligence. In *Explainable AI: interpreting, explaining and visualizing deep learning*, pages 5–22. Springer.

Samek, W., Wiegand, T., and Müller, K. (2017). Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. *CoRR*, abs/1708.08296.

Sanguinetti, M., Poletto, F., Bosco, C., Patti, V., and Stranisci, M. (2018). An Italian Twitter corpus of hate speech against immigrants. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May. European Language Resources Association (ELRA).

Sap, M., Card, D., Gabriel, S., Choi, Y., and Smith, A. N. (2019). The risk of racial bias in hate speech detection. In *ACL*.

Silveira, T. D. S., Uszkoreit, H., and Ai, R. (2019). Using aspect-based analysis for explainable sentiment predictions. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 617–627. Springer.

Strubell, E., Verga, P., Andor, D., Weiss, D., and McCallum, A. (2018). Linguistically-informed self-attention for semantic role labeling. *arXiv preprint arXiv:1804.08199*.

Vilone, G. and Longo, L. (2020). Explainable artificial intelligence: a systematic review. *CoRR*, abs/2006.00093.

Warner, W. and Hirschberg, J. (2012). Detecting hate speech on the world wide web. In *Proceedings of the second workshop on language in social media*, pages 19–26.

Waseem, Z. and Hovy, D. (2016). Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the NAACL student research workshop*, pages 88–93.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., and Brew, J. (2019). HuggingFace's Transformers: State-of-the-art Natural Language Processing. *ArXiv*, abs/1910.0.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., and Le, Q. V. (2019). Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

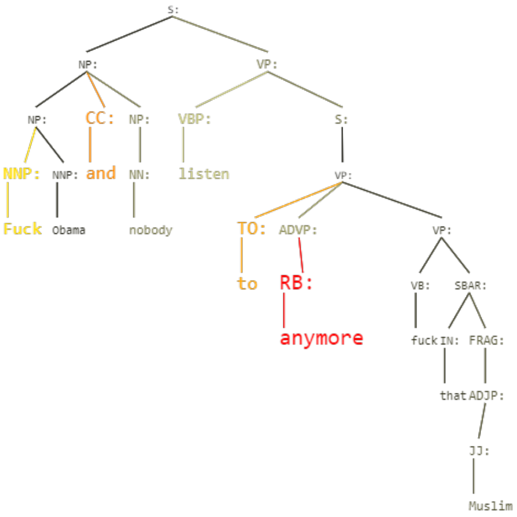
Zanzotto, F. M., Santilli, A., Ranaldi, L., Onorati, D., Tommasino, P., and Fallucchi, F. (2020). Kermit: Complementing transformer architectures with encoders of explicit syntactic interpretations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 256–267.

Zhao, A. and Yu, Y. (2021). Knowledge-enabled bert for aspect-based sentiment analysis. *Knowledge-Based Systems*, 227:107220.

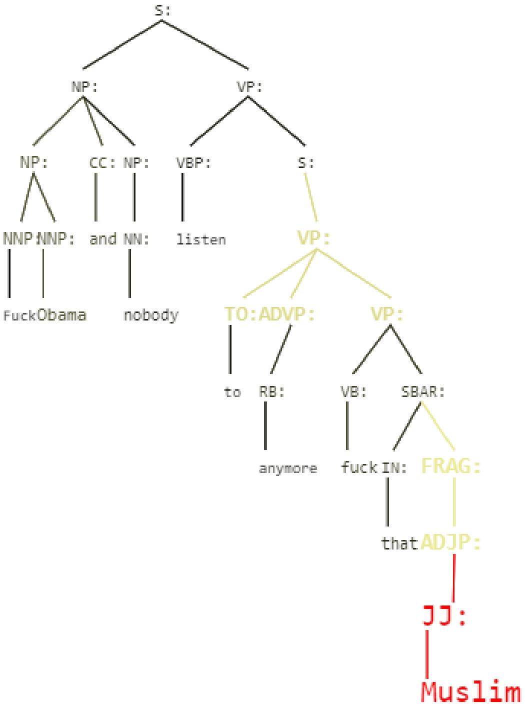
A. Qualitative analysis: extra examples

In this appendix, we show extra qualitative examples using KERMIT$_C$ and KERMIT$_T$ but also *contrasting trees*. We use the same schema used for Fig. 1 and Fig. 2.

Sentence: *Fuck Obama and nobody listen to anymore fuck that Muslim*

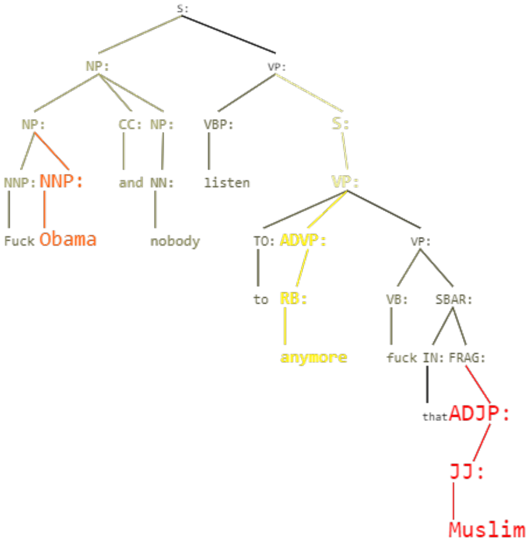


Labeled as **hate speech** for KERMIT$_C$

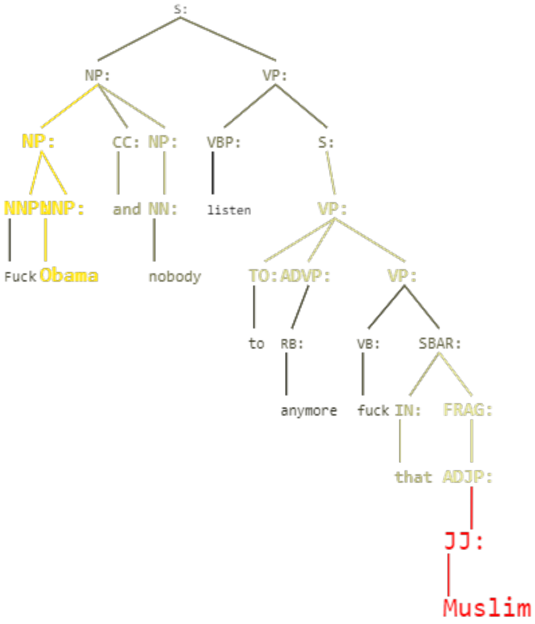


Labeled as **hate speech** for KERMIT$_T$

Contrasting trees

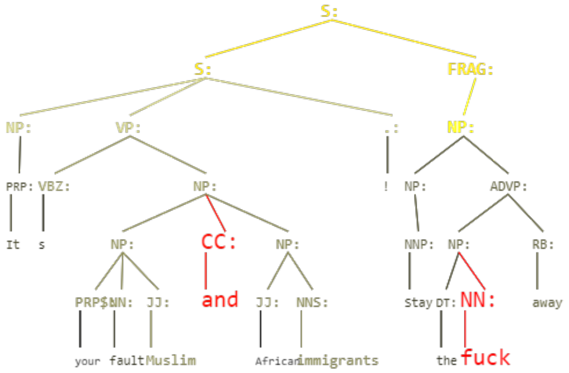


KERMIT$_C$ - KERMIT$_T$

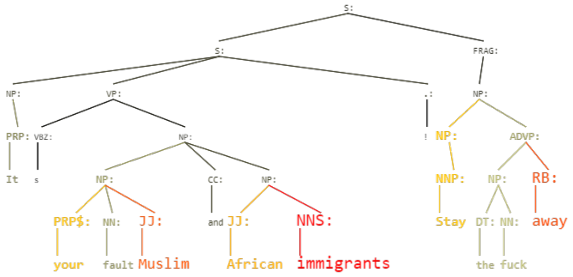


KERMIT$_T$ - KERMIT$_C$

Sentence: *It's your fault Muslim and African immigrants! Stay the fuck away*

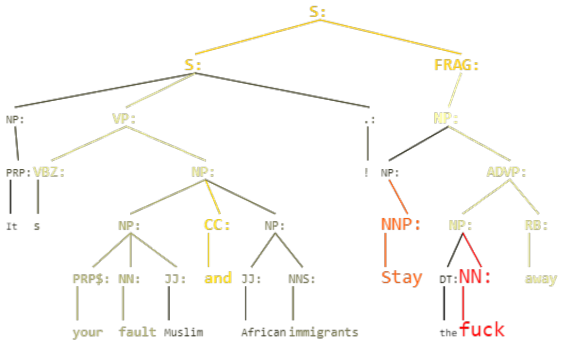

S:
S: FRAG:
NP: VP: .: NP:
PRP: VBZ: NP: ! NP: ADVP:
It s NP: CC: NP: NNP: NP: RB:
PRP$: NN: JJ: and JJ: NNS: Stay DT: NN: away
your fault Muslim African immigrants the fuck

Labeled as **hate speech** for KERMIT$_C$

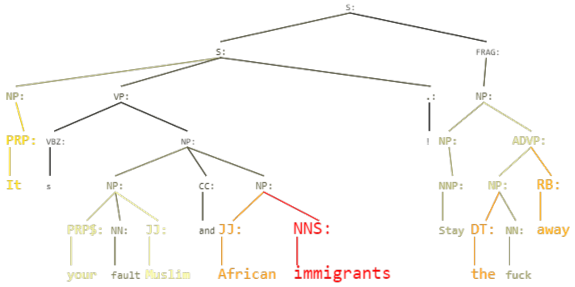

S:
S: FRAG:
NP: VP: .: NP:
PRP: VBZ: NP: ! NP: ADVP:
It s NP: CC: NP: NNP: NP: RB:
PRP$: NN: JJ: and JJ: NNS: Stay DT: NN: away
your fault Muslim African immigrants the fuck

Labeled as **hate speech** for KERMIT$_T$

Contrasting trees

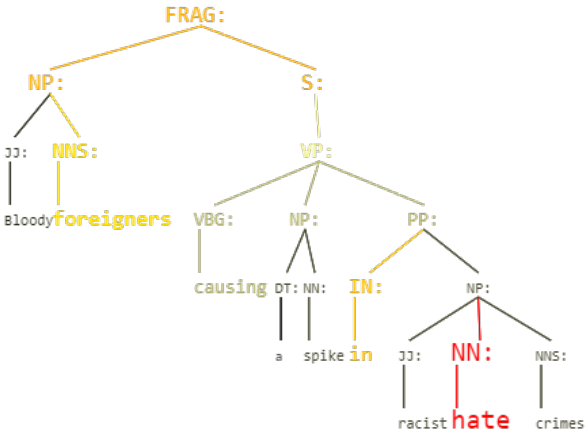

S:
S: FRAG:
NP: VP: .: NP:
PRP: VBZ: NP: ! NP: ADVP:
It s NP: CC: NP: NNP: NP: RB:
PRP$: NN: JJ: and JJ: NNS: Stay DT: NN: away
your fault Muslim African immigrants the fuck

KERMIT$_C$ - KERMIT$_T$

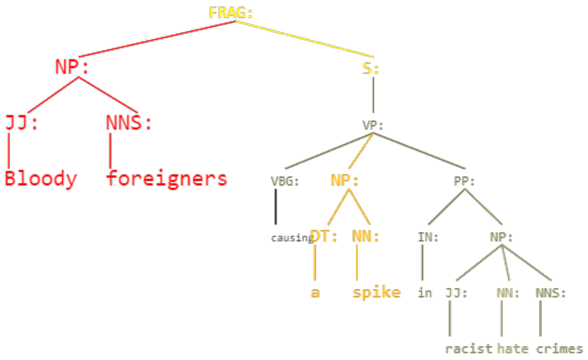

S:
S: FRAG:
NP: VP: .: NP:
PRP: VBZ: NP: ! NP: ADVP:
It s NP: CC: NP: NNP: NP: RB:
PRP$: NN: JJ: and JJ: NNS: Stay DT: NN: away
your fault Muslim African immigrants the fuck

KERMIT$_T$ - KERMIT$_C$

Sentence: *Bloody foreigners causing a spike in racist hate crimes*

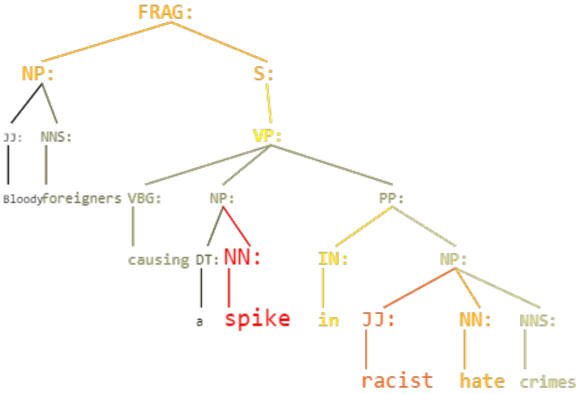


Labeled as **no hate speech** for KERMIT$_C$

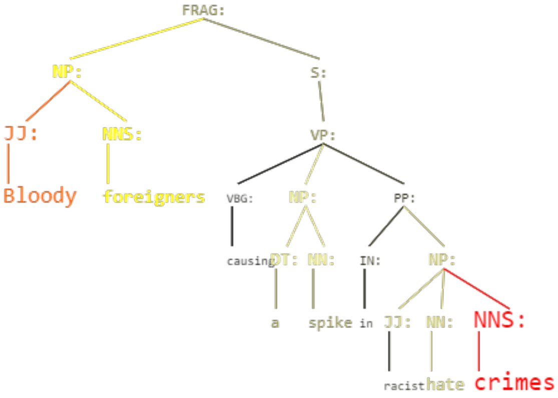


Labeled as **hate speech** for KERMIT$_T$

Contrasting trees

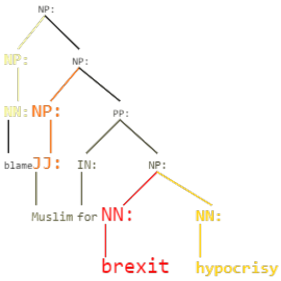


KERMIT$_C$ - KERMIT$_T$

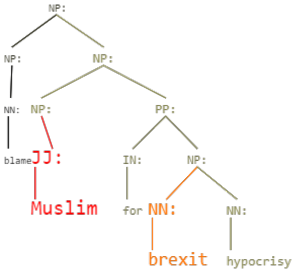


KERMIT$_T$ - KERMIT$_C$

Sentence: *blame Muslim for brexit hypocrisy*

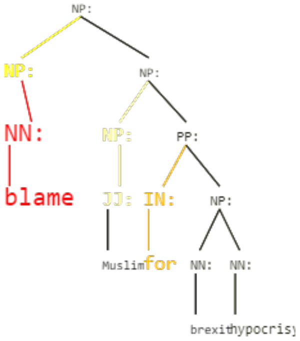


Labeled as **no hate speech** for KERMIT$_C$

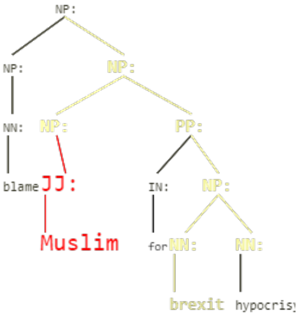


Labeled as **hate speech** for KERMIT$_T$

Contrasting trees



KERMIT$_C$ - KERMIT$_T$



KERMIT$_T$ - KERMIT$_C$