Visualizing Logic Explanations for Social Media Moderation

Demonstration Track

Marc Roig Vilamala
Cardiff University, United Kingdom
RoigVilamalaM@cardiff.ac.uk

Dave Braines
IBM Research Europe, United Kingdom
dave_braines@uk.ibm.com

Federico Cerutti
University of Brescia, Italy
federico.cerutti@unibs.it

Alun Preece
Cardiff University, United Kingdom
PreeceAD@cardiff.ac.uk

ABSTRACT

Autonomous artificial moderators can be useful to monitor social media for content that violates platform policies, but such artificial moderators can be confidently wrong about their decisions. While creating an approach that makes no mistakes is effectively impossible, being able to generate explanations for any given decision can simplify the task of detecting when the system is wrong. In this work we present LiveEvents, a neuro-symbolic agent capable of generating explanations based on which rules have lead to its decisions. We deliver these explanations via Cogni-Sketch, which provides users with an interactive visual representation, allowing them to easily understand the explanations given by the system.

KEYWORDS

Explainable AI; Neuro-symbolic; Situation Understanding

ACM Reference Format:
Marc Roig Vilamala, Dave Braines, Federico Cerutti, and Alun Preece. 2023. Visualizing Logic Explanations for Social Media Moderation: Demonstration Track. In *Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), London, United Kingdom, May 29 – June 2, 2023*, IFAAMAS, 3 pages.

1 INTRODUCTION

Social media platforms employ autonomous artificial moderator agents to monitor and remove content that violates their policies, such as hate speech or disinformation. Such autonomous moderators can be confidently wrong about their decisions, keeping content that should be removed, while removing acceptable content. In this paper, we introduce LiveEvents, an autonomous artificial moderator agent which can detect contents that need to be removed from the stream of social media posts, and generate an explanation for such decisions. While LiveEvents may still be confidently wrong, these explanations can make it easier for human moderators to identify when this is happening. Built upon the probabilistic complex event processing engine ProbCEP [8], LiveEvents employs a neuro-symbolic combination of pre-trained neural networks and a declarative layer; the former extracts information from the input stream of data, while the latter allows users to define rules specifying when the situations of interest are happening. LiveEvents enables human moderators to easily add or change the neural networks (NNs) being used, and to easily modify the declarative layer.

Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), A. Ricci, W. Yeoh, N. Agmon, B. An (eds.), May 29 – June 2, 2023, London, United Kingdom.

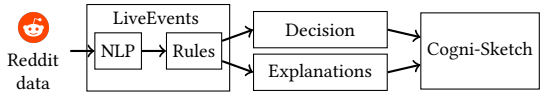

Figure 1: Architecture for the proposed system.

LiveEvents' explanations support human-AI collaboration by visually showing through Cogni-Sketch [1, 2] which rules have lead to a given decision. Cogni-Sketch is a platform designed to allow information and knowledge sharing between human and machine agents, where both are able to read and write their knowledge to an environment, allowing them to provide explanations or new knowledge to each other. In particular, LiveEvents uses Cogni-Sketch to show the user a graph that visualises the decisions and explanations, helping the user understand any given decision and identify *when* and *why* the system is wrong.

In this work, we explore a case where a Subreddit (a community within Reddit) has been suggested for *quarantining*[1] by LiveEvents, which would prevent its content from being accidentally viewed.

2 ARCHITECTURE AND DEMONSTRATION

Figure 1 shows our proposed architecture. First, Reddit comments are fed into LiveEvents, which uses pre-trained NNs to extract relevant features. Here, we use a Natural Language Processing (NLP) model. The output from this agent is used by the symbolic layer, where rules generate a decision and explanation. These are shown to the user via Cogni-Sketch. The following sections explore each part in more detail.

2.1 LiveEvents

LiveEvents[2] is designed to allow the use of any pre-trained NN on any type of input stream, and even allows multiple NNs to be used at the same time as pluggable agents, making the system very flexible. For our example, LiveEvents uses a BERT [4] model trained with the ETHOS Hate Speech Dataset [7] as the NLP model, which is able to identify whether a comment contains hate speech.

The output from these agents is then fed into a logic layer implemented in ProbLog [3, 5], a probabilistic logic programming language. This logic layer contains the rules that define under which conditions a given list of situations of interest occur. These situations can last for a period of time. For our demonstration,

[1] For more information on Reddit's quarantining policy see https://reddit.zendesk.com/hc/en-us/articles/360043069012-Quarantined-Subreddits (on 2nd of February 2023)
[2] Code for LiveEvents available at https://github.com/MarcRoigVilamala/LiveEvents

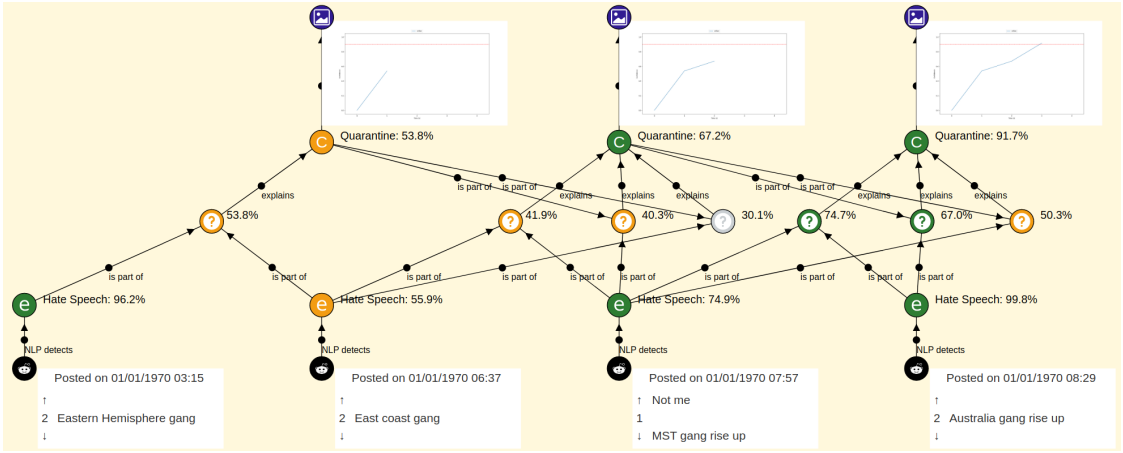

Figure 2: Cogni-Sketch visualization of LiveEvents decision and explanation. See video demonstration for better detail.

```
1  nlp(2, hate), nlp(0, hate).
2  nlp(2, hate), nlp(1, hate), not(nlp(0, hate)).
```

Listing 1: Example of explanations for *quarantining*.

LiveEvents outputs a value indicating whether a Subreddit should be quarantined. This value will change over time.

LiveEvents is also able to generate explanations for its decisions, based on these rules, using k-best explanations [6], which gives the k most likely non-overlapping explanations for a decision. Listing 1 shows an example, giving explanations for quarantining a subreddit. The first explanation is that comments 0 and 2 contain hate speech (line 1). Then, line 2 could seem to indicate that partly *because* comment 0 is not hate speech, we should quarantine, which seems counter-intuitive. However, line 2 should be read more like: *even if* comment 0 is not hate speech, since comments 1 and 2 are, we should still quarantine. In this case, `not(nlp(0, hate))` is added to remove any overlap with the first explanation.

While technical users may be able to understand and use this, we believe it would likely confuse the average user. As such, we provide a *simplified explanation* instead, which we generate by treating the given explanations as a disjuntive normal form formula, which is simplified according to logic rules. For Listing 1, this would remove the `not(nlp(0, hate))` clause from line 2, making it clear that we should quarantine *because* comments 1 and 2 contain hate speech.

2.2 Cogni-Sketch

After generating the decision and explanations, LiveEvents creates a JSON object, which is sent to Cogni-Sketch[3]. This creates an iteractive visual depiction that allows the user to understand the system's decision. Figure 2 shows an example where the system has incorrectly decided to quarantine the subreddit[4]. This example includes synthetically generated comments for illustration.

Cogni-Sketch shows nodes chronologically from left to right (Figure 2). The different types of node, from bottom to top, are (i) Reddit comments, (ii) NLP predictions, (iii) explanations, (iv) decisions (with their confidence) and (v) graphs showing the change in confidence over time. Graphs show a threshold at 90%, which needs to be surpassed to act on the decision. NLP, explanation and decision nodes have different colours to indicate different likelihoods.

In some cases, explanations are only based on the fact that multiple comments in a row contain hate speech (e.g. the leftmost explanation). However, to also take into consideration the previous confidence, some explanations also take into account the confidence in the previous iteration (e.g. the two rightmost explanations). This gives the approach a recursive nature, where each iteration depends on the previous one.

The graph visualization from Figure 2 allows human moderators to easily identify that the system is wrong, as it is confident that many innocuous comments contain hate speech. In particular, the agent seems to be confident that any comment containing the word 'gang' is hate speech. While this is somewhat understandable in certain contexts, it clearly does not apply to those comments.

3 CONCLUSIONS AND FUTURE WORK

We have demonstrated that the proposed architecture can allow users to easily identify when the system is wrong, and why. The architecture proposed also makes it possible to easily modify the system to prevent it from making the same mistake in the future, or making it easier for users to detect it. This could be done by defining rules that require the agreement of multiple NLP agents, which would minimize the biases of any individual agent. Another approach, if we detect that the system tends to commonly make the same mistakes, would be to add emphasis on such cases for human supervision. For instance, to address the example from Figure 2, notes could be added to ensure human moderators pay especial attention to the classification of comments with the word 'gang'.

Finally, as future work, we could look at automatically applying the solutions described above based on user feedback on when the system is wrong. Another area of future work would be learning rules automatically (through methods like inductive logic programming), thus removing the need for experts to define the rules.

[3]Code for Cogni-Sketch available at https://github.com/dais-ita/cogni-sketch
[4]For a video demonstration, see https://youtu.be/rXOWDYeJVMA

REFERENCES

[1] Dave Braines, Federico Cerutti, Marc Roig Vilamala, Mani B. Srivastava, Lance M. Kaplan, Alun D. Preece, and Gavin Pearson. 2020. Towards human-agent knowledge fusion (HAKF) in support of distributed coalition teams. *CoRR* abs/2010.12327 (2020). arXiv:2010.12327 https://arxiv.org/abs/2010.12327

[2] Dave Braines, Alun Preece, Colin Roberts, and Erik Blasch. 2021. Supporting Agile User Fusion Analytics through Human-Agent Knowledge Fusion. In *2021 IEEE 24th International Conference on Information Fusion (FUSION)*. 1–8. https://doi.org/10.23919/FUSION49465.2021.9627072

[3] Luc De Raedt, Angelika Kimmig, and Hannu Toivonen. 2007. ProbLog: A probabilistic Prolog and its application in link discovery, In IJCAI. *IJCAI International Joint Conference on Artificial Intelligence*, 2462–2467. www.ncbi.nlm.nih.gov/Entrez/

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR* abs/1810.04805 (2018). arXiv:1810.04805 http://arxiv.org/abs/1810.04805

[5] Daan Fierens, Guy Van den Broeck, Joris Renkens, Dimitar Shterionov, Bernd Gutmann, Ingo Thon, Gerda Janssens, and Luc De Raedt. 2015. Inference and learning in probabilistic logic programs using weighted Boolean formulas. *Theory and Practice of Logic Programming* 15, 03 (2015), 358–401.

[6] ANGELIKA KIMMIG, BART DEMOEN, LUC DE RAEDT, VÍTOR SANTOS COSTA, and RICARDO ROCHA. 2011. On the implementation of the probabilistic logic programming language ProbLog. *Theory and Practice of Logic Programming* 11, 2-3 (2011), 235–262. https://doi.org/10.1017/S1471068410000566

[7] Ioannis Mollas, Zoe Chrysopoulou, Stamatis Karlos, and Grigorios Tsoumakas. 2022. ETHOS: a multi-label hate speech detection dataset. *Complex & Intelligent Systems* (Jan. 2022). https://doi.org/10.1007/s40747-021-00608-2

[8] Marc Roig Vilamala, Liam Hiley, Yulia Hicks, Alun Preece, and Federico Cerutti. 2019. A Pilot Study on Detecting Violence in Videos Fusing Proxy Models. In *2019 22th International Conference on Information Fusion (FUSION)*. 1–8.