Neural Sandbox Framework for Classification: A Concept Based Method of Leveraging LLMs for Text Classification

Mostafa Mushsharat
North South University
Dhaka, Bangladesh
mostafa.mushsharat@northsouth.edu

Nabeel Mohammed
North South University
Dhaka, Bangladesh
nabeel.mohammed@northsouth.edu

Mohammad Ruhul Amin
Fordham University
New York, USA
mamin17@fordham.edu

Abstract

We introduce a neural sandbox framework for text classification via self-referencing defined label concepts from a Large Language Model(LLM). The framework draws inspiration from the define-optimize alignment problem, in which the motivations of a model are described initially and then the model is optimized to align with these predefined objectives. In our case, we focus on text classification where we use a pre-trained LLM to convert text into vectors and provide it with specific concept words based on the dataset labels. We then optimize an operator, keeping the LLM frozen, to classify the input text based on how relevant it is to these concept operator words (cop-words). In addition to exhibiting explainable features, experiments with multiple text classification datasets and LLM models reveal that incorporating our sandbox network generally improves the accuracy and macro f1 when compared to a baseline. The framework, not only improves classification but also provides insights into the model's decision making based on the relevance scores of provided cop-words. We also demonstrated the framework's ability to generalize learned concepts and identify potential biases through spurious relations. However, we found that the model's incentives may not always align with human decisions.

1 Introduction

In recent years, we've observed impressive advancements in various Natural Language Processing (NLP) tasks, largely attributable to the adoption of Large Language Models (LLMs). LLMs pretrained on tasks such as masked language modeling or next sentence prediction, has been considered to produce embedding that are highly adaptable to be used to perform a variety of tasks where minimal data or context is available[1]. However, with the increasing popularity of these models, the issue of Alignment of AI models with human goals have become more prominent than ever. Alignment is mostly discussed with the assumption that the AI system is a delegate agent, which is perhaps due to a perception that language agents would have limited abilities to cause serious harm [2]. However, this position has been challenged by [3] justifying multiple paradigms of risks and dangers associated with language models.

R0-FoMo: Workshop on Robustness of Few-shot and Zero-shot Learning in Foundation Models at NeurIPS 2023.

Development in explainability and interpretability may be one of the key approach in assessing and mitigating the risks and biases associated with language models. A popular way to approach this problem is to develop auxiliary models that offer post-hoc explanations for a pre-trained model by learning a second, typically more interpretable model that acts as a proxy. Notable examples include LIME, as proposed by [4] , which employs input perturbation to create auxiliary models. Auxiliary model-based approaches are model-agnostic and can provide both local (as demonstrated by [5]) and global (as illustrated by [6]) explanations. However, it's important to note that auxiliary models and the original models may employ entirely different mechanisms for making predictions, raising concerns about the fidelity of auxiliary model-based explanations [7].

A more inherent way for explainability is exploring feature importance. These approaches can be founded upon diverse feature types, including manually crafted features, as exemplified in [8], lexical features such as words and n-grams, as demonstrated in studies like [9]. Feature importance-based explanations often leverage common techniques like attention mechanisms, as introduced by [10], and first-derivative saliency, as presented by [11]. These methods are limited because they rely solely on the input tokens for understanding predictions.

In contrast, our method incorporates provided objective definitions of a set of concepts that can be further expanded. This allows the classifier to choose relevant concepts from this set while labeling input text, and exhibit these concept scores after classification, making our method more versatile than those based solely on word tokens. Figure: 1 provides a demonstration for an input text classification task within our architecture. The concept scores in relation to the input text can be extracted to serve as an explanation of the classification decision; Figure: 2 provides a visual comparison of saliency map explanations against our framework.

In addition to this we also compare the classifier's decision of chosen concepts for an input text to human labels, by meticulously choosing datasets that have a hierarchical classification labels. We train the model supervising only on the higher level class labels while providing lower level class labels as their concepts. This allows us to understand incentives – secondary objectives that the model might adopt in order to learn and influence parts of the environment in pursuit of the primary objective [12].

Furthermore, due to the nature of the architecture of the sandbox framework, we can essentially test the model, post training, with alternate concept definitions different than the ones used in training. This allows us to evaluate the classifier on its learned representation and find out spurious corelations to irrelevant concepts.

Our contribution in this paper can be summarized as:

- We introduce a sandbox framework for text classification that utilizes a frozen large language model in relation to label concepts. This framework utilizes the similarity of the model's responses to predefined objective definitions of concepts called cop-words, which are determined based on the labels provided. These similarity scores are used to perform classification for an input text.

- Our experiment with the sandbox framework architecture resulted in improved text classification. We trained this framework using frozen pre-trained LLMs, including "bert-base-uncased," "roberta-large," and "t5-encoder-large," on datasets such as GoEmotion and IMDB. As a baseline, we trained a basic fully connected layer classifier on the same datasets and models. Generally, our framework enhanced performance, increasing accuracy by 0.12% to 6.31% and macro F1 by 0.3% to 8.82%, except for the "bert-base-uncased" model on the IMDB dataset, where we observed a slight drop of 0.1% in accuracy and 0.13% in macro F1.

- Evaluating models using foreign cop-words (which have different concept definitions than the ones used during training) demonstrates that the model's performance remains mostly intact, indicating that the model's representations are conceptually aligned with the original cop-word definitions. Further tests involves assessing models trained on sentiment datasets having positive/negative labels, with "neutral" cop-words extracted from SentiWordNet[13]. These tests revealed some spurious correlations with these irrelevant definitions. Additionally, by injecting predefined bias-related terms, we identified potential biases in the model.

- Finally, we provide evidence that the models' secondary motivations diverge significantly from human judgments. While most models achieve high accuracy (beyond 80%) on the supervised objective, their accuracy is below 10% when their unsupervised cop-word similarity scores are evaluated against human labels.

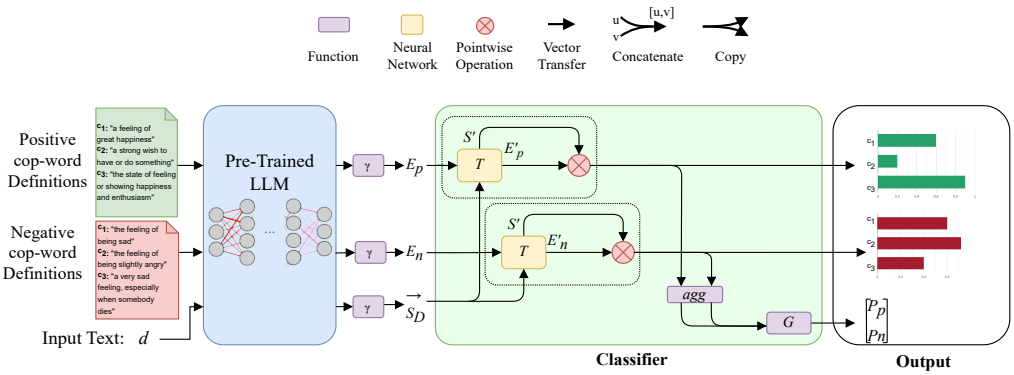


Figure 1: Demonstration of the sandbox framework architechture for a binary classification task with labels Postive/Negative. The sample input d and cop-word definitions are fed into the same frozen LLM (Blue). The γ function outputs the input representation (average pooled embedding of all last layer embedding). These are then projected in a new learnable space in the classifier. The $similarity$ scoring using *cosine similarity* (Red) produces a set of input scores on all cop-words. These scores are then put under the agg: *Max* on *Relu*, to produce an aggregate score for the sentence that is used in the activation function *G* function to produce probabilities for each task: $p_{positive}, p_{negative}$. Along with the probabilities the scores produced for each cop-words are also produced as output relaying the model's understanding of the input d's relevance to the cop-words.

2 Proposed Methodology

In AI and linguistics, a fascinating challenge is defining complex concepts by interconnecting multiple related ones, mirroring human cognition's interconnected knowledge structures for a deeper understanding of the world. One effective method for defining concepts is using semantic networks or knowledge graphs, where interconnected relationships with related concepts provide a more comprehensive definition, as seen with the example of "bird" linked to "feathers," "wings," "flight," and "beak". This approach finds support in cognitive science and computational linguistics. In the work of [14], the authors introduced the hierarchical semantic network model, demonstrating how concepts could be organized in a tree-like structure, with higher-level concepts encompassing more specific ones.

Now, in the task of text classification, where each input text needs to be classified to any of a given set of possible labels; each of these labels are essentially concepts that can be broken down into multiple concept words. To construct a text classifier, we may use defined concept words for each label then check how relevant these concepts are to the input text. Finally we can predict the label whose concepts best matches input text.

To illustrate this further, let's formulate the role of a large language model that takes a document d as input and produces an sequence of embedding in n-dimensional vector space, $\mathbb{R}^n$, in text classification.

We can define such language model as: $LLM([d]) = D = (\vec{e_1}, \vec{e_2}, \vec{e_3}, ..., \vec{e_{n_s}})$ where $\vec{e_i}$ is in $\mathbb{R}^n$ and $d \in \{$x:x is a document of the input dataset$\}$. And,$\gamma(D) = \vec{s_D}$ where $\vec{s_D}$ is in $\mathbb{R}^n$ and γ is a representation function. We can take the average pooled last layer embedding as the representation function.

Any categorical label can be defined by multiple concepts. On that idea we extrapolate any data label $y^{(i)}$, to have multiple concepts, $C_{y^{(i)}} = \{c_1, c_2, c_3, \ldots, c_{n_C}\}$, such that; $\forall_i \exists_j \bullet P(C^{(j)}, y^{(i)})$, where $P(A, B) = A$ is a concept word of label B.

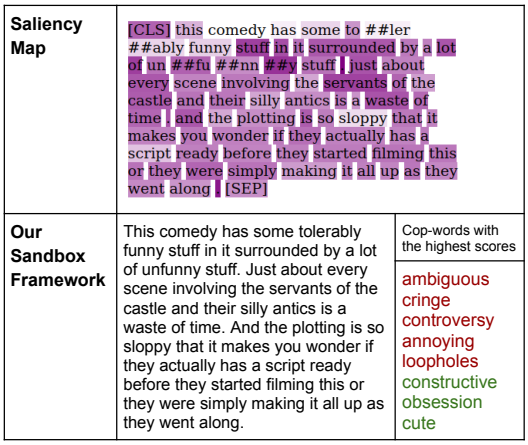


Figure 2: Comparison of output of saliency map and our sandbox framework scores for explanation of model prediction. The intensity of the color is proportional to the saliency score associated with the token of the sentence. Whereas, our framework outputs cop-words scores from where we can sort for the highest associated concepts with the input sentence as a whole.

Table 1: Primary and Secondary Objective Accuracy on the dataset GoEmotion. While the models perform well on the supervised primary objective classification task, the unsupervised secondary objective is not at all aligned to human labels.

	GoEmotion	
	Primary Objective	*Secondary Objective*
bert-base-uncased	83.47	**8.48**
roberta-large	82.53	**7.06**
t5-encoder-large	83.19	**9.43**

Now, we design an architecture to train our dataset for classification on the labels in y, on the basis of C. We may do this by using a similarity function like cosine similarity to find the similarity of a concept to d. However, we will face two problems while using transformer based LLMs: First, we understand that unlike models that produce static embedding, transformer models are contextual, so instead of using embedding of words in a contextual sentence, we decide on defining descriptive texts for a concept word and use that through our LLM and γ for the concept's embedding. Next, since we will use frozen encoder LLMs for our architecture which generally struggle to effectively capture complex and sparse factual information in text [15][16], to mitigate this we use a learnable operator for our embedding. Thus we can call the set of concept words for a label as concept operator words(cop-words) which are defined using description documents.

Considering a binary classification task, where the labels are either positive or negative as demonstrated in Figure: 1. We can have the representation of the label's cop-words on our language model space, passing them through our LLM then γ, consequentially. We may create an embedding tensor with these representation vectors from cop-words for each label. Consequently, in our case, for positive label and negative label we will have: $E_{positive}$ and $E_{negative}$ where $E \in \mathbb{R}^{n_n \times n_m}$ and n is the hidden embedding size and m is the number of cop-words for the label. Then, we transform these embedding with our operator Transformation tensor, $T \in \mathbb{R}^{n_n \times n_n}$; $E'_{positive} = E_{positive} \cdot T$ and $E'_{negative} = E_{negative} \cdot T$. Similarly, we can find the image of input sentences to classify, $s^{(i)}$ under T: $s^{(i)'} = s^{(i)} \cdot T$.

To classify any input sentence, $s^{(i)'}$ we need to simply find its $similarity$ with E of each label, $f(s^{(i)'}, E'_{label}) = \forall\{j \in E'_{label} : 0 < j < n_m\}, similarity(s^{(i)'}, E'_{label})$. We use cosine similarity for the $similarity$ function.

The resulting set, f, is a similarity score of n_m cop-words for each $label$. Since we have two classes: positive and negative, we will have two set of scores $f_{positive}$ and $f_{negative}$. We can calculate the aggregate score for each label, $agg(f_{label})$, with this set. For agg we do Relu[17] on top of: *Max* which will return the maximum score from the vector. Passing these aggregate values through an activation function, G, we obtain the probabilities for positive and negative for the input sentence: $\hat{y_{(i)}} = G(f_{positive}, f_{negative})$. We use the non linear activation function Softmax[18] for G. Finally, the loss can be formulated by $L(y_i, \hat{y_{(i)}})$, where we use the Cross Entropy Loss CE. This loss is now used in backpropagration to optimize T over the training data.

Table 2: Performance of Models of the two datasets compared to the simple classifier baseline with our sandbox framework. The accuracy and f1-macro scores are separated by / presented in %. The second row shows performance scores using native cop-words(cop-words used in training). The third row: native cop-words definitions paraphrased, and fourth: performance using foreign cop-words in the same domain.

	IMDB			**GoEmotion**		
	bert-base-uncased	*roberta-large*	*t5-encoder-large*	*bert-base-uncased*	*roberta-large*	*t5-encoder-large*
Simple Classifier	85.04/84.30	89.74/89.34	90.92/89.23	83.07/80.86	81.73/79.59	76.88/72.80
Native cop-words	84.94/84.43	90.02/89.63	91.04/90.75	83.47/81.51	82.53/80.61	83.19/81.62
Paraphrased cop-words	85.12/84.62	88.84/88.43	91.06/90.79	83.66/81.94	82.56/80.75	81.89/80.80
Foreign cop-words	79.82/79.27	87.48/87.01	90.32/89.96	83.35/81.30	80.28/75.86	82.21/81.04

3 Experiments

3.1 Datasets

IMDB Movie Review Dataset [19] The dataset includes 50,000 sentences categorized into positive and negative sentiments. We use 80% of samples for training, with 10% for validation and 10% for testing. Cop-words representing emotional reactions were created since the IMDB dataset lacks emotion labels. The cop-words used can be found in Appendix: A.1.

GoEmotions[20] GoEmotions, a dataset with 27 emotion labels, is used, and we only include instances with one emotion label (excluding ambiguous and neutral). The data is split into 80% for training, 10% for validation, and 10% for testing. Emotion labels are chosen as cop-words, while sentiment labels, mapped from the emotion-sentiment mapping provided, are used for classification. This mapping transforms the dataset into a binary sentiment classification task.

Cop-word definitions To define cop-words we make use of definitions documents from the Oxford Dictionary [21].

3.2 Experimental Setup

We use the models: "bert-base-uncased", "roberta-large", and "t5-encoder-large" presented in the works, [22],[23] [24] as our frozen LLMs. The base model produces embedding in $\mathbb{R}^{768}$ and large models in $\mathbb{R}^{1024}$.

The formal way to set up a classification architecture with large language models is to connect a fully connected layer with dimension n_y x d, where d is the hidden layer dimension of the model's output. We set up a fully connected layer with our LLM of dimension nx2 as the classification layer and use this as a **Simple Classifier** baseline to compare performance.

To assist training, gradients are normalized in each step using gradient clipping normalization. All experiments were conducted using a linear rate scheduler and AdamW[25] optimizer starting from 0.001 for 8 epochs; retaining the best parameters of the best accuracy observed in validation set. The language model's weights are frozen so that only our Transformation tensor T consists of learnable parameters.

Primary and Secondary Objective Since we supervise the training with the sentiment labels: positive/negative, this is the primary objective of the models. The decision of sentiment comes from the cop-word relevance scores from $f_{positive}$ and $f_{negative}$, with the maximum score's cop-word being the deciding factor for the sentiment. We can extract this cop-word and use it to match with human labels that were not used in training. This test will subsequently serve as a measure of alignment of the model's unsupervised decisions to human labels. This task is the secondary objective.

4 Evaluation

We can observe from Table: 2, performance of models where our sandbox framework is used, generally exhibit better performance compared to a simple fully connected classifier with the exception of "bert-base-uncased" with the IMDB dataset, where performance slightly drops by 0.1% in accuracy and 0.13% in macro f1. Interestingly, this model beats the baseline when a foreign injection with paraphrased cop-words is used. See 4.1. We notice that using larger models with our framework have more significant improvement with the baseline in both macro f1 scores and accuracy. The largest difference with the baseline can be observed with the model "t5-encoder-large", where on the dataset GoEmotions: there is an increase of 6.31% in accuracy and 8.82% increase in macro f1.

4.1 Foreign Injection of cop-words

The E tensor is a flexible tensor in our architecture such that clipping it and injecting new cop-words, does not hamper the ensemble of the system. This provides an opportunity to test our framework on foreign cop-words not used in training. We call the cop-words used in training, native cop-words while the ones injected afterwards, foreign cop-words.

Foreign injection of Native cop-words paraphrased We perform a test of Foreign injection where we construct cop-words definitions for the native definitions by paraphrasing them using the tool automatic paraphrasing tool Parrot[26]. We then inject these cop-words to the model and perform testing to see the impact of grammatical nuances to the performance of the model. As we can see from Table: 2, the performance of the model still retains when a paraphrased version of the native cop-words are used. This provides evidence that the learned operator T has low correlations to grammatical features. In one occasion, in our experiments, the paraphrased native cop-word beat both the baseline and the native cop-words used in training for the model "bert-base-uncased" with the IMDB dataset. Because, the discrepancies are low between the scores, this prompts for further investigation with a larger knowledge base as cop-words to properly conclude any reasoning. We leave this as a further scope for this paper.

Foreign injection of alternate cop-words on same domain By formulating alternate cop-words that fall in the domain of positive concepts and negative concepts, we select new foreign words and replace them with our E tensor. The selected foreign cop-words can be found in the Appendix. In Table: 2, we can observe that nearly all models retain performance very well when tested by injecting Foreign cop-words. This implies that the foreign words have features under T projection that are similar to the native cop-words used.

Neutral Foreign cop-words injection To validate our results we stress the models trained on GoEmotions and IMDB with foreign injections of randomly chosen 300 pairs of neutral cop-words from the SentiWord3.0 [13] corpus which provides a large corpus of words with definitions labeled with sentiment: positive, negative or neutral. Since the models were trained on positive and negative labels, cop-words that are neutral should be irrelevant and useless as a classifier concept words. Ideally, an irrelevant set of cop-words should perform close to 50% in macro f1 for a binary classification task. However, as we can observe from the distribution of the macro f1 scores of these neutral pairs in Figure:3 of Appendix, even when multiple pairs of neutral cop-words perform close to the native performance, most of the pairs in the 300 chosen, underperforms. For a neutral cop-word, the closer the performance score is to the native performance, the more spurious it is. We try to look at the mean scores of cop-words in our test set to analyse similarities and differences between native, foreign, neutral and spurious cop-words. This is further elaborated in the Section: A.2 of the Appendix.

Now, to test if these spurious correlations exist in terminologies that may have potential biases, we further perform testing with 22 potential bias terminologies as discussed in Section: A.3. We use the cop-words of these bias terms to find that most of our models show positive spurious correlations with most of the bias terms. In Table: 3 of the Appendix, we can observe the bias terms that are non spurious or negatively spurious to the model in our experiments. Section: A.3 of the Appendix offer a more comprehensive explanation of this process.

4.2 Alignment with Human Decisions

The dataset: GoEmotion is an emotion classification dataset where we map each emotion to its sentiment to train and optimize tensor T on binary sentiment classification task (the model's primary

objective) while providing the emotions for each label as their cop-words. Even though the model receives cop-words of the emotions as the E tensor, it is not supervised per row to its exact emotion label. For an input sentence the operator transformation tensor T optimizes itself to make any of the provided cop-word's relevance score highest and chooses that for the primary objective, since we are using Max for the agg function. During testing, we evaluate the max cop-word per emotion for the input sentences and match it with their actual emotion label(secondary objective) according to Section:3.2. In Table: 1, we can observe, the model despite being able to perform very well on the primary objective(supervised higher level labels), is not aligned at all to the secondary objective(unsupervised lower level labels). While the model excels at tasks where it's given clear examples with labels, its alignment is not close to human understanding in terms of specific representation. The model might not be capturing the same patterns or latent features that a human would consider relevant or meaningful, all the while performing quite well on its primary objective. This raises concern for other high performance models in classification and opens a paradigm for quantifying the dissimilarity.

5 Discussion and Further Improvement

Our sandbox framework presents itself as an explainable process of text classification through cop-word scores. In addition to this we also find, upon testing this method's performance on multiple datasets, its advantage on a baseline in terms of performance and explain the capability of the framework to perform sensible operations while performing classification. Our experiments use frozen LLMs on top of the classifier training the operator T only. As opposed to fully finetuning, this method aims to keep the generalizability of the LLM intact and leverage inherent knowledge that the language model "learns" during pre-training and perform subsequent analyses that shed light on this knowledge [27]. While our choice of models for the experiments were encoder models due to their effectiveness in text classification task, it is worth exploring results with sandbox framework leveraging larger encoder-decoder models as they have known to be superior recently in similar tasks [28].

We further demonstrate the misalignment of model decisions with human labels with its unsupervised decisions. Regardless, through multiple evaluations with cop-word injections we confirm the model's ability to understand domain knowledge and demonstrate some unusual spurious correlations with seemingly irrelevant neutral cop-words. This also allowed us to use bias criterias and identify concerning spurious correlations with them. The scope of our experiments fall in the text classification task of the datasets that we used, however, there is potential of the framework to be used in more complex tasks by formalizing the objectives and calculations inside the sandbox. Furthermore, incorporating a larger set of cop-words with an appropriate knowledge base of concepts for the classification task may be a scope of further findings, both in terms of performance and explainability. Our results in qualitative analysis of the explainability method also brings awareness to construct a fair method of evaluation for other methods of model interpretability. This is important for the sake of improving AI safety and reducing biases.

6 Acknowledgement

We would like to express our appreciation for the financial support provided by Giga Tech Limited, BEXIMCO; whose scholarship played a crucial role in facilitating and advancing this research. We also extend our gratitude to the reviewers whose insightful feedback helped enhance the clarity of the paper.

References

[1] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[2] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

[3] Zachary Kenton, Tom Everitt, Laura Weidinger, Iason Gabriel, Vladimir Mikulik, and Geoffrey Irving. Alignment of language agents. *arXiv preprint arXiv:2103.14659*, 2021.

[4] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

[5] David Alvarez-Melis and Tommi Jaakkola. A causal framework for explaining the predictions of black-box sequence-to-sequence models. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 412–421, Copenhagen, Denmark, September 2017. Association for Computational Linguistics.

[6] Ninghao Liu, Xiao Huang, Jundong Li, and Xia Hu. On interpretation of network embedding via taxonomy induction. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1812–1820, 2018.

[7] Yonatan Belinkov and James Glass. Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 7:49–72, 2019.

[8] Nikos Voskarides, Edgar Meij, Manos Tsagkias, Maarten De Rijke, and Wouter Weerkamp. Learning to explain entity relationships in knowledge graphs. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 564–574, 2015.

[9] Amirata Ghorbani, Abubakar Abid, and James Zou. Interpretation of neural networks is fragile. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 3681–3688, 2019.

[10] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[11] Jiwei Li, Xinlei Chen, Eduard Hovy, and Dan Jurafsky. Visualizing and understanding neural models in nlp. *arXiv preprint arXiv:1506.01066*, 2015.

[12] Tom Everitt, Marcus Hutter, Ramana Kumar, and Victoria Krakovna. Reward tampering problems and solutions in reinforcement learning: A causal influence diagram perspective. *Synthese*, 198(Suppl 27):6435–6467, 2021.

[13] Stefano Baccianella, Andrea Esuli, Fabrizio Sebastiani, et al. Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In *Lrec*, volume 10, pages 2200–2204, 2010.

[14] Allan M Collins and M Ross Quillian. Retrieval time from semantic memory. *Journal of verbal learning and verbal behavior*, 8(2):240–247, 1969.

[15] Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China, November 2019. Association for Computational Linguistics.

[16] Robert Logan, Nelson F. Liu, Matthew E. Peters, Matt Gardner, and Sameer Singh. Barack's wife hillary: Using knowledge graphs for fact-aware language modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5962–5971, Florence, Italy, July 2019. Association for Computational Linguistics.

[17] Abien Fred Agarap. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018.

[18] John Bridle. Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters. *Advances in neural information processing systems*, 2, 1989.

[19] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

[20] Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. Goemotions: A dataset of fine-grained emotions. *arXiv preprint arXiv:2005.00547*, 2020.

[21] Angus Stevenson. *Oxford dictionary of English*. Oxford University Press, USA, 2010.

[22] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

[23] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[24] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

[25] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[26] Prithiviraj Damodaran. Parrot: Paraphrase generation for nlu., 2021.

[27] Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*, 2019.

[28] Yova Kementchedjhieva and Ilias Chalkidis. An exploration of encoder-decoder approaches to multi-label classification for legal and biomedical text. *arXiv preprint arXiv:2305.05627*, 2023.

[29] Nikhil Garg, Londa Schiebinger, Dan Jurafsky, and James Zou. Word embeddings quantify 100 years of gender and ethnic stereotypes. *Proceedings of the National Academy of Sciences*, 115(16):E3635–E3644, 2018.

A Appendix

A.1 Cop-words

Native Cop-words for IMDB Dataset:

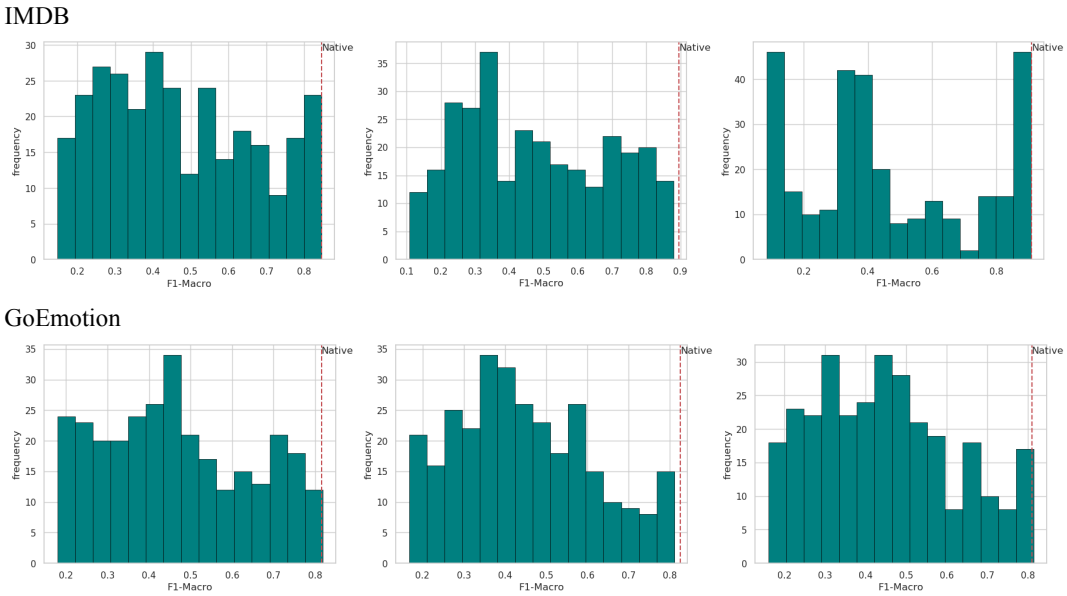
positive - acclaimed, accurate, adventurous, astonishing, authentic, beautiful, calming, catchy, charismatic, cheerish, coherent, constructive, cool, cute, daring, eloquent, enthusiastic, romantic, flawless, humorous, inspirational, love, modern, motivated.

negative - ambiguous, angry, annoying, appalling, awful, barbaric, bizarre, blasphemous, brainless, chaotic, contradictory, controversy, cringe, cruel, degrading, disturbed, failed, fake, gimmick, hateful, hideous, inadequate, inappropriate, incoherent, loopholes.

Foreign Cop-words for IMDB and GoEmotion Dataset:

positive - captivating, enjoy, outstanding, thoughtful, fun, pleasant, warm, enticing, realistic, friendly, obsession, phenomenal, relistic, refreshing, vibrant, wholesome.

negative - biased, horrible, bored, disappointed, frustrate, hostile, ridiculous, malign, rude, unpleasant, meaningless, obscure, offensive, pathetic, weird.



(a) bert-base-uncased (b) roberta-large (c) t5-encoder-large

Figure 3: Distribution of F1 Scores on iterated subsampled foreign injection of neutral cop-words from sentiword corpus. The dotted red vertical line represents the f1 score of the appropriate model with its native cop-words.

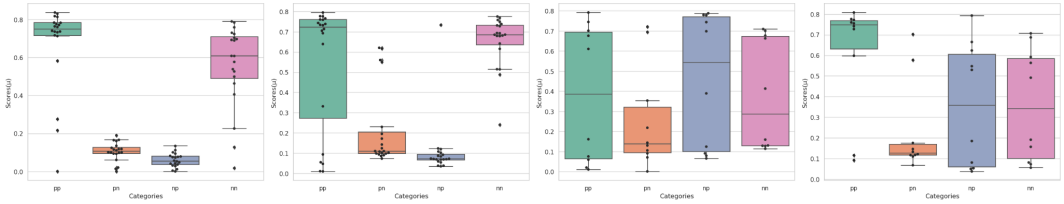
A.2 Cop-word mean scores in binary sentiment classification

Fundamentally, a positive cop-word should have a higher score with positive sentences in the dataset than the negative sentences. Likewise, a negative cop-word should have higher score with negative sentences than positive. To analyse the cop-words scores on behavior such as this, we produce the mean score for each cop-word on the positive sentences of the test set and the negative sentences of the test set. Figure: 4 provides a visualization of box plot for mean scores of positive cop-words with positive sentences, positive cop-words with negative sentences, negative cop-words with positive sentences, and negative cop-words with positive sentences of the sandbox framework trained on "bert-base-uncased" with datasets IMDB and GoEmotions. For native cop-words, the range of positive cop-words with positive sentences is higher than positive cop-words with negative sentences, neglecting a few outliers. This trend is also parallel with negative cop-word mean scores. This is expected as to perform well on the dataset the cop-words scores have to be conceptually accurate. We even see this trend with foreign cop-words where the model also performs well as discused in previous sections. Even if the range of positive cop-word mean scores with positive sentences spreads out for the foreign cop-words in IMDB, we can still see the interquartile range is higher then the negative sentence range. Intuitively, in the case of neutral injection of cop-words where the relation is non-spurious, we can see the ranges of the plots overlapping substantially. However, we can observe that in terms of neutral cop-words with spurious correlations to the native set, at least of the pair is highly overlapping. Here in the Figure: 4, the positive cop-word pairs seem to be spuriously correctly co-relating with positive/negative sentences in both the datasets. However, as we notice the negative cop-words still show substantial overlapping.

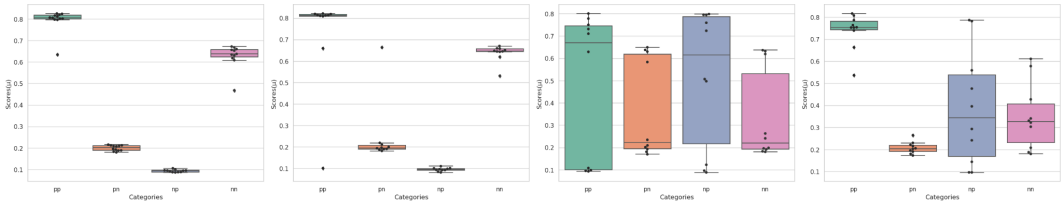
A.3 Bias Terminologies

We devise a set of 22 terminologies that may have potential biases including: Activist, Advocate, Chubby, Colored, Dialogue, Gender, Homosexual, Indian, Industry, Islam, Jew, Marriage, Media, Misgendering, Money, Non-professional and professional occupations from the work [29], Oriental, Orientation, Retarded, Society, and Woman. We produce and define cop-words for these terms using ChatGPT. These cop-words are used to find spurious correlations associated with bias terminologies.

IMDB



GoEmotions



(a) Native (b) Foreign (c) Neutral (d) Spurious

Figure 4: Box Plots of mean scores of positive and negative cop-words with postive and negative sentences in the test set of IMDB and GoEmotions trained on "bert-base-uncased". On each plot there are four box-plots: pp(Positive mean scores with Positive Sentences), pn(Positive mean scores with Negative Sentences), np(Negative mean scores with Positive Sentences), nn(Negative mean scores with Negative Sentences).

We understand, by experimentaion, that a model will perform well even if the cop-words provided have a non spurious neutral cop-words paired with a set of either positive or negative cop-words. Now, instead of using a set of positive/negative cop-words, we pair a non-spurious set, found from our previous randomised testing, with a bias term's cop-words. We do this by injecting the bias cop-words as both a positive with a non-spurious set, and negative with a non-spurious set. If the macro f1 is more than 0.6 for a setting we can conclude the cop-words of that bias term has spurious correlations. For example, for the model "bert-base-uncased" trained on GoEmotions dataset [20], we find from the random neutral stress testing the pairs ['psychically', 'valgus', 'profile', ..] as positive and ['mole mol gram molecule', 'waste', ...] as negative, the model shows as f1 score of 0.30. We take this as the base non spurious pair. When we test the model again with the cop-words of the bias term "Woman" as positive and ['mole mol gram molecule', 'waste', ...] as negative, the model assumes a score of 0.67 f1 score. Thus, we conclude the model is positively biased towards cop-words of "Woman".

Table 3: The table displays which of the 22 bias terminologies have spurious corelations with the models in our experiments. For each model, terms written in plain text are non spurious for the model and terms written in double quotations and red text are spurious for the model as negative concepts. Any terms of the 22 not displayed are spurious as positive concepts.

"bert-base-uncased" w/ GoEmotion	**"roberta-large" w/ GoEmotion**	**"t5-encoder-large" w/ GoEmotion**
Oriental Gender	Indian Colored Non-Professional Occupations Society Money Woman Activist Jew Homosexual Industry Oriental Professional Occupations Gender Media	Advocate Colored Non-professional Occupations Woman Jew Homosexual Media
"bert-base-uncased" w/ IMDB	**"roberta-large" w/ IMDB**	**"t5-encoder-large" w/ IMDB**
Woman Chubby Misgendering Industry Gender		Non-professional Occupations Woman Chubby Oriental "Retarded" "Professional Occupations"