Fine-tuning Language Models with Conditioning on Two Human Preferences

Group 19
Department of Computer Science, University College London

Abstract

This paper investigates the effectiveness of fine-tuning language models (LMs) with multiple control tokens to reduce generated content that is misalignment to human preferences; specifically, content that is toxic and negative. The performance of our proposed LM is evaluated based on its ability to generate non-toxic and positive content. Experiment results demonstrate that conditioning with multiple control tokens is feasible and can improve the LM alignment with human preferences. Therefore, these findings suggest that fine-tuned LMs have the potential to generate content that is free of bias or offensive language, which could be useful in developing safe language models for public use. Further research, however, is needed to optimise conditioning on multiple tokens.

1 Introduction

Language models (LMs) have demonstrated impressive capabilities in generating human-like responses in natural language processing tasks. With the introduction of ChatGPT to the mass market, there has never been a broader audience utilising these models (Hu, 2023). However, LMs, including GPT-3, have been shown to produce offensive, inappropriate, and bias responses (Brown et al., 2020; Gehman et al., 2020; McGuffie and Newhouse, 2020), highlighting the need for effective methods to control their language output. OpenAI (2023) argue that safety is a primary research area of interest.

Detoxification techniques fall into two categories; decoding-based and data-based approaches. Park and Rudzicz (2022) describe decoding detoxifiers as techniques that adjust the probability of certain tokens (words) to avoid toxic outputs by the LM. An example of a decoding-based approach is word filtering, where the probability of an LM generating a word from a given banned dictionary is set to zero (Xu et al., 2021b). Similarly, Cao et al. (2023) introduced a method, rectification, that reduces the probability of words according to how likely they are to cause a toxic output in the future.

Decoding-based approaches are computationally easy to implement as they do not change the weights within the model (Cao et al., 2023). However, Xu et al. (2021a) found that they hinder the fluency of the model. This result was amplified when the LM was prompted with words describing or used by minority groups. This highlights the bias in detoxifiers and the importance of context (Pavlopoulos et al., 2020).

Data-based detoxifiers include pretraining or fine-tuning LMs on specific datasets that are designed to filter out inappropriate language or behaviour. This is necessary as it was found that the web text corpora that GPT-2 was trained on contains a significant amount of toxic content (Gehman et al., 2020). Furthermore, Bender et al. (2021) argue that due to the profile of participants in online spaces, models that are trained on internet content are vulnerable to bias and overrepresentation of specific views. This data-based approach has been shown to improve toxicity performance (Gururangan et al., 2020). In fact, Gehman et al. (2020) found that pretraining an LM on non-toxic data was the most effective intervention for reducing the toxicity of text generations by GPT-2.

The downside of pretraining an LM on non-toxic data is the loss of diversity in the training data. Hendrycks et al. (2020) found that a broader range of pretraining data improved the LM robustness. This tells us that, while we might not want LMs to repeat some of the behaviour they are trained on, they benefit from seeing all the data.

Another data-based approach that is implemented during training is conditioning on control tokens. This involves adding tokens to the input sequence that provide additional information to the LM. These tokens can be used to convey several types of information, such as style and content (Keskar et al., 2019). This can help guide the LM to generate responses aligned with the user's preference (Ficler and Goldberg, 2017). Korbak et al. (2023) utilised conditional training to indicate human preferences, including non-toxic text. Therefore, the LM is trained on diverse data, but is told at the time of training which sentences are 'toxic 'and which are 'nontoxic '.

Toxicity comes in many forms (Cao et al., 2023). Blodgett et al. (2020) also found that the definition and threshold of biases changes between researcher and perspective. This implies that a single toxic classifier cannot accurately flag all types of toxic language, bias and unpreferable behaviour generated by an LM. Furthermore, it is common to find embedded bias within the training data and labels used by classifiers (Bender et al., 2021; Sap et al., 2019; Hovy and Prabhumoye, 2021). Therefore, it is more appropriate to consider multiple measures of human preference to encapsulate multiple points of view (Welbl et al., 2021; Xu et al., 2021b).

Our research This research investigates the effectiveness of conditioning on multiple control tokens as a method of aligning LM generated text with human preference. We fine-tune GPT-2 on data containing two binary indicators; toxicity and sentiment. By extending the conditioning from one control token to two, we determine the effectiveness of multiple token conditioning, with the view to extending this to more tokens in the future.

The performance of the model is determined by its misalignment score. This is defined as the percentage of model generated sentences that contradict the conditioning token. For example, if the prompt is conditioned on 'nontoxic', then the model's misalignment score is the percentage of generated sentences classified as toxic. Thus, the lower the misalignment score, the more effective the conditioning has been. The impact of increasing the number of tokens during fine-tuning is also considered.

2 Related work

Korbak et al. (2023) used conditioning during the pretraining of an LM. To define the condition, a toxic classifier was used to label training sentences. These labels were turned into a control token that was pre-pended to the start of the training sentence. The text generation was then conditioned on the desired human preference by including the control token in the prompt. It was found that conditional training exceeded the performance of alternative detoxifying methods in both human preference alignment and general capability. Our research intends to extend the work of Korbak et al. (2023) by evaluating the effectiveness of conditioning with multiple control tokens to align the behaviour of an LM with multiple human preferences.

Previous research into the effectiveness of conditioning to control LM behaviour looked at conditioned LMs (Keskar et al., 2019; Ficler and Goldberg, 2017) as well as conditioned neural machine translations (Sennrich et al., 2016). While these have impressive performance for controlling the output, it is not a practical solution for existing LMs as all training includes control tokens. Therefore, the alternative method of fine-tuning LMs with control tokens (Korbak et al., 2022) enables conditioning to be applied to a range of LMs.

Fine-tuning can also be achieved through reinforcement learning from human feedback (RLHF) (OpenAI, 2023). One approach of RLHF is comparative feedback (Ziegler et al., 2020), where human evaluators choose the best LM generated response. Scheurer et al. (2023) proposed Imitation learning from Language Feedback, which incorporates both language and comparison feedback.

3 Methods

Language modelling LMs assign probabilities to sentences of words, such as sentences. This research will exclusively focus on sentences. For a given sentence $\mathbf{w} = (w_1, w_2, ..., w_l)$, and an LM initialised with some weights, θ, the probability chain rule states that the sentence $p_\theta(\mathbf{w})$ can be modelled as

$$p_\theta(\mathbf{w}) = \prod_{t=1}^{l} p_\theta(w_t | w_1, ..., w_{t-1})$$

In other words, the goal of language modeling is to model the probability of the next word in the sentence, given the history of preceding words.

MLE For many state-of-the-art LMs (Radford and Narasimhan, 2018), the weights θ are fine-tuned by maximum likelihood estimation (MLE). That is, given a data set of training documents $D = \{x^1, x^2, ..., x^{|D|}\}$, LMs find the optimal set of weights $\pi = \arg\max_\theta \mathcal{L}(D)$ that maximise the log-likelihood function $\mathcal{L}(D)$:

$$\mathcal{L}(D) = \log p_\theta(D) \qquad \mathcal{L}_{MLE}(D) = \log p_\pi(D)$$

where $\log p_\pi(D)$ can be rewritten as

$$\log p_\pi(D) = \sum_{k=1}^{|D|} \sum_{i=1}^{|x^k|} \log p_\pi(x_i^k | x_{<i}^k)$$

Here, x_i^k denotes the i^{th} sentence in the k^{th} document while $x_{<i}^k$ denotes all sentences prior to x_i^k in the same document.

Conditional language modelling A conditional language model assigns probabilities to sentences, *given some context*, say c. In a similar manner to unconditional LMs, the probability of a sentence given some context c, denoted by $p_\theta(\mathbf{w}|c)$, can be modelled as

$$p_\theta(\mathbf{w}|c) = \prod_{t=1}^{l} p_\theta(w_t | c, w_1, ..., w_{t-1})$$

Therefore, to put it plainly, the aim is to model the probability of the next word, given the preceding words *and the condition*.

Conditional MLE To extend the MLE approach to fine-tuning, a control token c is added at the beginning of each sentence to condition the type of language to be generated. This has been demonstrated to successfully improve model alignment (Keskar et al., 2019; Korbak et al., 2023). Conditional LMs therefore learn $\pi|c$ the maximum of the conditional log-likelihood function $\mathcal{L}(D|c)$. Put simply, the conditional MLE is defined by

$$\begin{aligned} \mathcal{L}_{Cond.MLE}(D|c) &= \log p_{\pi|c}(D) \\ &= \sum_{k=1}^{|D|} \sum_{i=1}^{|x^k|} \log p_{\pi|c}(x_i^k | c_i^k, x_{<i}^k) \end{aligned}$$

where c_i^k is the control token for the i^{th} sentence in the k^{th} document.

Furthermore, conditional LMs are able to extend to more than one condition. In particular, the LMs we train in our study will be conditioned by up to two conditions, i.e. control tokens. That is, they learn $\pi|c_1, c_2$ the maximum of the conditional log-likelihood function $\mathcal{L}(D|c_1, c_2)$, defined by

$$\mathcal{L}_{Cond.MLE}(D|c_1, c_2) = \log p_{\pi|c_1,c_2}(D)$$
$$= \sum_{k=1}^{|D|} \sum_{i=1}^{|x^k|} \log p_{\pi|c_1,c_2}(x_i^k | c_{1i}^k, c_{2i}^k, x_{<i}^k)$$

where c_{1i}^k and c_{2i}^k are the control tokens with respect to the first and second condition respectively (for the i^{th} sentence in the k^{th} document).

Conditional fine-tuning To conditionally fine-tune, sentences in the training data set are prepended with control tokens c using pre-trained classifiers. In our study, we adopt two classifiers to fine-tune GPT-2 to align with the two human preferences; using non-toxic language and expressing positive sentiment. For both human preferences, control tokens are indicated by <|condition|>, and are determined by the label assigned by the corresponding classifier. Firstly, we utilise the pre-trained toxicity classifier Detoxify (Hanu and Unitary team, 2020) to classify sentences as <|nontoxic|> if their toxicity score is less than or equal to a threshold value t, or <|toxic|> otherwise. We treat the threshold value t as a hyperparameter, following the approach taken by Korbak et al. (2023). Secondly, we utilise VADER (Hutto and Gilbert, 2014), a pre-trained sentiment classifer, to classify sentences as <|pos|> (positive) or <|neg|> (negative), by taking the label with the highest probability. To fine-tune GPT-2 to align with both human preferences, we pre-prend sentences with a <|nontoxic|> or <|toxic|> token, *and* a <|pos|> or <|neg|> token.

4 Experimental set up

4.1 Tasks

The model fine-tuned and conditioned on two control tokens (M-TS) is evaluated against three tasks: (i) generating nontoxic content; (ii) generating positive content; and (iii) generating nontoxic and positive content. The model's performance is compared to models that are fine-tuned only on the data (M-B), and models that are fine-tuned on a single control token (M-T and M-S).

The impact of the number of tokens in the fine-tuning data is evaluated through a final task (iv). This task involves fine-tuning M-TS with increasing amounts of tokens. The resulting models are compared to understand the effect of increasing the number of tokens on performance.

4.2 Data

To fine-tune the LM, we use randomly sampled sentences from the diverse data set *the Pile* (Gao et al., 2021). To ensure data compatibility with the LM, we exclude data sources that contain non-compatible content. This content includes coding and multi-lingual information from sources such as GitHub and EuroParl.

We take a 2% sample from the remaining data sources. The sample is processed by removing special characters; adding an <|endoftext|> token to the end of each sentence; and removing short, low-quality sentences. Control tokens for toxicity; <|toxic|> and <|nontoxic|>; and sentiment; <|pos|> and <|neg|>; are pre-pended to the processed sentences, based on the classification provided by Detoxify and VADER, respectively. Classifier tokens are not added to a random 1% of the sampled sentences to maintain alignment with the LM, as per Korbak et al. (2023). The resulting training data set is comprised of 800K sentences, and 25M tokens.

Finally, we create a validation data set of 400 random sentences that are sampled and cleaned as above. The sentences in the validation set are used as prompts for the models at evaluation. Sentences are chosen to be evenly distributed across the classifier groups (the four combinations of toxicity and sentiment) to prevent any bias.

4.3 Model

GPT-2 small (Radford et al., 2019), pretrained with 1.24M parameters, is selected as the base model to fine-tune. The GPT-2 tokenizer is retained, and the four control tokens from the fine-tuning data set; <|nontoxic|>, <|toxic|>, <|pos|>, and <|neg|>; are added as special tokens to the tokenizer. GPT-2 small's head is extended to facilitate the additional control tokens, enabling the model to generate text given some condition(s). We fine-tune the model using 64-sentence batches with a maximum length of 64 tokens, along with an attention mask. Finally, we use the conditional MLE approach to fine-tune, as described in Section 3.

Four models are fine-tuned, according to different combinations of pre-pended control tokens. One model is trained for each of the three tasks (i)-(iii) highlighted in Section 4.1, in addition to a base model for comparison. These four models are given by:

- M-B: The base model, only fine-tuned on the data set without any control tokens
- M-T: Toxicity Model, fine-tuned on data with one binary toxicity control token
- M-S: Sentiment Model, fine-tuned on data with one binary sentiment control token
- M-TS: Toxicity and Sentiment Model, fine-tuned on data with two binary control tokens; one binary token for toxicity and one binary token for sentiment.

Each model undergoes fine-tuning for five epochs on the training data set described in Section 4.2. The best epoch of each model is then chosen for evaluation. For task (iv), an additional 1.6M sentences are used to fine-tune M-TS, to examine the effect of fine-tuning with increasing amounts of tokens. In Section 5, the performance of the fine-tuned models described above is discussed.

4.4 Evaluation

Text generation is used to evaluate the fine-tuned LMs. To achieve this, each LM is prompted with 400 random sentences from the validation set. To ensure fairness, the same random sentences and hyperparameters are used to evaluate each LM.

To evaluate the models without conditioning, each sentence from the evaluation data set is used to prompt the model without any changes to the sentence. To evaluate the models with one condition, the corresponding control token is appended after each sentence from the validation set, i.e. one of <|nontoxic|>, or <|pos|>. Finally, to evaluate the models with two conditions, both control tokens are appended in concatenation, i.e. <|nontoxic|><|pos|>.

Generated text for each input prompt is limited to 40-70 tokens and is segmented into sentences using Spacy (Montani et al., 2023). This equates to between one to three sentences (excluding the prompt itself). The sentences are classified using Detoxify and VADER (Hanu and Unitary team, 2020; Hutto and Gilbert, 2014) to identify undesirable content. The frequency of undesirable content determines the following misalignment evaluation metrics:

- Toxicity misalignment: Percentage of generated sentences classified as toxic when the LM is conditioned with the <|nontoxic|> token.
- Sentiment misalignment: Percentage of generated sentences classified as negative when the LM is conditioned with the <|pos|> token.
- Toxicity *and* Sentiment misalignment: Percentage of generated sentences classified either as toxic *or* negative when the LM is conditioned with <|nontoxic|><|pos|> tokens.

Model	Model description	Control types	Epochs
M-B *baseline model*	GPT-2 small fine-tuned with no added control tokens	None	1
M-T *toxicity model*	GPT-2 small fine-tuned with one binary control token	<|nontoxic|> <|toxic|>	1
M-S *sentiment model*	GPT-2 small fine-tuned with one binary control token	<|pos|> <|neg|>	5
M-TS *toxicity and sentiment model*	GPT-2 small fine-tuned with two binary control tokens	<|nontoxic|> <|pos|> <|nontoxic|> <|neg|> <|toxic|> <|pos|> <|toxic|> <|neg|>	1

Table 1: Final models to be evaluated for experiments (i) - (iii). For these experiments, all models are fine-tuned with the same number of sentences.

5 Experiments

To select the best version (epoch) of each of the four models, we compare the evaluations of each version as described in Section 4.4. For each model, we select the version that has the lowest misalignment to the relevant human preference, according to the condition it is fine-tuned on. Specifically, for M-T and M-S, we take the version that scores the lowest toxicity misalignment and sentiment misalignment respectively. For M-B, we take the version that has the lowest sum of toxicity misalignment and sentiment misalignment. Finally, for M-TS, we select the version that scores the lowest misalignment on toxicity *and* sentiment misalignment.

Using the best versions of each model, shown on Table 1, we evaluate the performance of M-TS against three tasks: (i) generating nontoxic content; (ii) generating positive content; and (iii) generating nontoxic and positive content.

5.1 One condition: Toxicity

In this task, we condition the models M-B, M-T and M-TS on <|nontoxic|> only. Therefore, we are conditioning the models to produce non-toxic sentences. We compare M-TS against the other models in terms of toxicity misalignment.

Results The results of the experiment are shown in Table 2. As expected, conditioning for non-toxicity in the baseline model M-B does not make any significant improvement of the toxicity of sentences. We see a good improvement in the performance of M-TS with over 10% less toxic sentences generated when conditioned on <|nontoxic|> compared to non conditioning. Overall, when conditioning on <|nontoxic|>, M-TS generates around 6% less toxic sentences compared to M-B.

Model	No condition	<|nontoxic|>
M-B	34.5%	32.2%
M-T	26.1%	23.5%
M-TS	37.4%	26.0%

Table 2: Toxicity misalignment

The performance of M-TS is worse than that of M-T when conditioning on <|nontoxic|>. However, the difference is only 2.5%. Therefore, it is difficult to know if this is a significant result or caused by noise in the results. Furthermore, when no condition is given, M-T generates much fewer toxic sentences. This is an unexpected result as we expect all models to generate approximately the same number of toxic sentences when no condition is given.

5.2 One condition: Sentiment

In this task, we condition the models M-B, M-S and M-TS on <|pos|> only. Thus, we are prompting the models with a positive sentiment condition. We compare M-TS against M-B and M-S in terms of sentiment misalignment.

Results The results of the experiment are shown in Table 4. We see a very similar trend to the first experiment. That is, when conditioned only on the <|pos|> token, the performance of M-TS is worse

Content warning: this table contains LM samples that are offensive in nature

Model	Generated text	Toxicity	Sentiment
M-TS *No conditioning*	Bring what I would do now if you'd love to say so a splash of crap because it's too late in this winter period as fast!	toxic	negative
M-TS *With conditioning*	down a few years ago to get back in your pocket so the car would be available for you.Now this will only require him with our strength but when we are open it right now even if I am able of giving me whatever those situations have that they should not always carry?	nontoxic	positive

Table 3: Examples of generated sentences by M-TS with no conditioning, and when conditioned on <|nontoxic|><|pos|>. Both sentences were prompted by the toxic, negative sentence "You fucking dickhead".

than M-S but better than M-B.

In this experiment, the difference between the sentiment misalignment of M-TS and M-S is larger than experiment (i) at around 4% when conditioned. The difference between M-TS and M-B when conditioned is also larger at just under 13%. When no condition is given, all three models produce a similar number of sentences with negative sentiment, as expected.

Model	No condition	<|pos|>
M-B	56.1%	54.6%
M-S	55.6%	37.8%
M-TS	56.4%	41.9%

Table 4: Sentiment misalignment

5.3 Two conditions

In this task, all four models; M-B, M-T, M-S, and M-TS; are conditioned on <|nontoxic|><|pos|>. Therefore, we are conditioning the models to generate sentences that are both non-toxic *and* positive. We compare the performance of M-TS against the other three models in terms of toxicity *and* sentiment misalignment.

Results The results of the experiment are shown in Table 5. When conditioned on the sequence of these two tokens, <|nontoxic|><|pos|>, the performance of M-TS is superior to that of M-B, M-T and M-S. In particular, it generates almost 16% less misaligned sentences than M-B when conditioned on the two tokens. While M-S shows around 10% improvement in misalignment when conditioned compared to no conditioning, it still has around 8% more misalignment than M-TS.

Model	No condition	<|nontoxic|><|pos|>
M-B	67.3%	68.3%
M-T	64.3%	66.4%
M-S	70.9%	60.3%
M-TS	71.2%	52.4%

Table 5: Toxicity *and* Sentiment misalignment

We would expect all of the models to have a similar misalignment score when no condition is given. However, there is a range of almost 7% of misalignment scores. This result is once again unexpected.

5.4 Increasing tokens

In this experiment we evaluate the effect of increasing the number of fine-tuning sentences on misalignment. This is applied to the M-TS model. The performance of the resulting models is evaluated against the misalignment for toxicity *and* sentiment.

Results Figure 1 shows the results for this experiment. The plot shows the misalignment score against the number of fine-tuning sentences. The number of fine-tuning sentences is given as a percentage of the number of sentences used in the fine-tuning of the original M-TS used in experiments (i) - (iii). Thus, 100% of the data corresponds to 800K sentences and the maximum amount of sentences we used to fine-tune is 2.4M (corresponding to 300%).

Figure 1 shows that increasing the fine-tuning sentences improves the performance up to 200%.

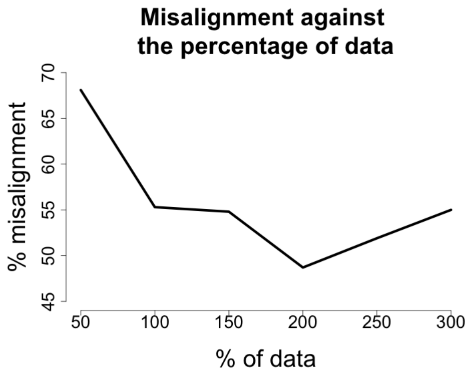

Figure 1: A line graph to show the percentage of generated misaligned sentences when M-TS is fine-tuned with increasing amounts of data.

This is shown by a decreasing misalignment score from 68% for half of the data to 49% for twice the data. However, after 200% of the data, the performance starts to deteriorate again. At 300% of the data, the performance is approximately equivalent to that at 100% of the data.

6 Discussion

Overall, the experiments show that fine-tuning with conditioning does improve model alignment to human preferences. This is supported by the fact that the model M-TS performed better than the baseline model, M-B, in every task when conditioned. In particular, conditioning on two control tokens increases the alignment with the intersection of the two preferences. When conditioning on two tokens, the M-TS model performed considerably better than the other three models.

However, more work is still needed to improve the overall performance of conditioning on multiple tokens. The overall best performance of a model is M-T when conditioned on a single token with 23.5% misalignment. Whereas the best performance when conditioning on two tokens is around 50% misalignment (M-TS fine-tuned with 200% of the data). Therefore, conditioning on two tokens has around twice the misalignment of one token. Furthermore, the intention of this research is to extend this to more than two conditions. More research is needed to determine if misalignment continues to increase as the number of conditions increases.

The M-TS model does not perform as well as M-T and M-S when prompted with only one conditioning token. We hypothesise that this is because M-TS did not have any fine-tuning examples with only one control token. This has a similar justification to including a small percentage of fine-tuning data with no control tokens. Without including examples with one token in the fine-tuning data, prompting M-TS with only one token moves away from the distribution the model is fine-tuned on. Therefore, we would like to extend this research by including some examples of sentences with one token in the fine-tuning data to see if performance improves when conditioned on one token.

In Section 5.4, we observe an unexpected increase in the misalignment of M-TS as the number of tokens surpasses 1.6M, despite an initial improvement. This finding contradicts Korbak et al. (2023)'s study, where they reported a decrease in misalignment with increasing data. One explanation for this discrepancy is that we use a higher threshold for toxicity classification, which may make the classification more sensitive to noise. Future research should aim to explore this outcome in greater detail, and evaluate the impact of continuing to increase the amount of data in the fine-tuning.

A further limitation of our experiment is the quality and fluency of the generated sentences of the fine-tuned models. We can see in Table 3 that the generated sentences lack coherence and intelligibility. By comparing the generated sentences by models M-B to M-TS, we can infer this is not caused by conditioning. Therefore, we hypothesise it is caused by the fine-tuning process. This might be due to the limited size or restricted source of the training data. Another cause could be the inconsistency between the length of tokens in the pre-training of GPT-2 small (512 tokens), compared to the length of tokens it is then fine-tuned with (64). More research is needed to understand the cause and to increase the quality of the the generated text after fine-tuning. Using a capability metric such as the Kullback-Leibler divergence from a more advanced model (Korbak et al., 2023) would allow the quality of the generated text to be quantified.

Finally, it is important to note that the two human preferences examined - toxicity and sentiment - are strongly correlated (Brassard-Gourdeau and Khoury, 2019). Future research should investigate the impact of correlated preferences on the conditional fine-tuning of LMs.

7 Conclusion

This research investigated the effectiveness of conditioning on multiple control tokens to align LM generated text with multiple human preferences. This was achieved by fine-tuning GPT-2 small on data containing two control tokens (representing toxicity and sentiment). LM prompts were then conditioned on the control tokens. The results of the research show that conditioning of two control tokens considerably improves the model's performance in comparison to the baseline model. This is a promising approach to aligning LM generated text with human preference by generating fewer sentences contradicting the conditioning tokens.

Further research is needed to optimise conditioning on multiple tokens. This should include understanding more about the impact of increasing the amount of fine-tuning data as well as the impact of increasing the number of control tokens. Our findings demonstrate the potential for fine-tuned LMs to generate content that is free of bias or offensive language. This could be valuable in the development of safe LMs for public use.

Acknowledgments

We would like to thank the ongoing support and advice from our teaching assistant throughout the project.

References

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 610–623, New York, NY, USA. Association for Computing Machinery.

Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna M. Wallach. 2020. Language (technology) is power: A critical survey of "bias" in NLP. *CoRR*, abs/2005.14050.

Évelyne Brassard-Gourdeau and Rawad Khoury. 2019. Subversive toxicity detection using sentiment information. In *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying (TRAC-2) at the Annual Meeting of the Association for Computational Linguistics (ACL)*. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *CoRR*, abs/2005.14165.

Meng Cao, Mehdi Fatemi, Jackie Chi Kit Cheung, and Samira Shabanian. 2023. Systematic rectification of language models via dead-end analysis.

Jessica Ficler and Yoav Goldberg. 2017. Controlling linguistic style aspects in neural language generation. In *Proceedings of the Workshop on Stylistic Variation*, pages 94–104, Copenhagen, Denmark. Association for Computational Linguistics.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2021. The pile: An 800gb dataset of diverse text for language modeling. *CoRR*, abs/2101.00027.

Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A. Smith. 2020. RealToxicityPrompts: Evaluating neural toxic degeneration in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3356–3369, Online. Association for Computational Linguistics.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks.

Laura Hanu and Unitary team. 2020. Detoxify. Github. https://github.com/unitaryai/detoxify.

Dan Hendrycks, Xiaoyuan Liu, Eric Wallace, Adam Dziedzic, Rishabh Krishnan, and Dawn Song. 2020. Pretrained transformers improve out-of-distribution robustness.

Dirk Hovy and Shrimai Prabhumoye. 2021. Five sources of bias in natural language processing. *Language and Linguistics Compass*, 15(8):e12432.

Krystal Hu. 2023. Chatgpt sets record for fastest-growing user base - analyst note. *Reuters*.

C. Hutto and Eric Gilbert. 2014. Vader: A parsimonious rule-based model for sentiment analysis of social media text. *Proceedings of the International AAAI Conference on Web and Social Media*, 8(1):216–225.

Nitish Shirish Keskar, Bryan McCann, Lav R. Varshney, Caiming Xiong, and Richard Socher. 2019. Ctrl: A conditional transformer language model for controllable generation.

Tomasz Korbak, Hady Elsahar, German Kruszewski, and Marc Dymetman. 2022. Controlling conditional language models without catastrophic forgetting.

Tomasz Korbak, Kejian Shi, Angelica Chen, Rasika Bhalerao, Christopher L. Buckley, Jason Phang, Samuel R. Bowman, and Ethan Perez. 2023. Pretraining language models with human preferences.

Kris McGuffie and Alex Newhouse. 2020. The radicalization risks of gpt-3 and advanced neural language models.

Ines Montani, Matthew Honnibal, Matthew Honnibal, Sofie Van Landeghem, Adriane Boyd, Henning Peters, Paul O'Leary McCann, jim geovedi, Jim O'Regan, Maxim Samsonov, György Orosz, Daniël de Kok, Marcus Blättermann, Duygu Altinok, Søren Lind Kristiansen, Madeesh Kannan, Raphael Mitsch, Raphaël Bournhonesque, Edward, Lj Miranda, Peter Baumgartner, Richard Hudson, Explosion Bot, Roman, Leander Fiedler, Ryn Daniels, Wannaphong Phatthiyaphaibun, Grégory Howard, and Yohei Tamura. 2023. explosion/spaCy: v3.5.1: spancat for multi-class labeling, fixes for textcat+transformers and more.

OpenAI. 2023. Gpt-4 technical report.

Yoon A Park and Frank Rudzicz. 2022. Detoxifying language models with a toxic corpus.

John Pavlopoulos, Jeffrey Sorensen, Lucas Dixon, Nithum Thain, and Ion Androutsopoulos. 2020. Toxicity detection: Does context really matter?

Alec Radford and Karthik Narasimhan. 2018. Improving language understanding by generative pretraining. *OpenAI Blog*.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Maarten Sap, Dallas Card, Saadia Gabriel, Yejin Choi, and Noah A. Smith. 2019. The risk of racial bias in hate speech detection. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1668–1678, Florence, Italy. Association for Computational Linguistics.

Jérémy Scheurer, Jon Ander Campos, Tomasz Korbak, Jun Shern Chan, Angelica Chen, Kyunghyun Cho, and Ethan Perez. 2023. Training language models with language feedback at scale.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Controlling politeness in neural machine translation via side constraints. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 35–40, San Diego, California. Association for Computational Linguistics.

Johannes Welbl, Amelia Glaese, Jonathan Uesato, Sumanth Dathathri, John Mellor, Lisa Anne Hendricks, Kirsty Anderson, Pushmeet Kohli, Ben Coppin, and Po-Sen Huang. 2021. Challenges in detoxifying language models.

Albert Xu, Eshaan Pathak, Eric Wallace, Suchin Gururangan, Maarten Sap, and Dan Klein. 2021a. Detoxifying language models risks marginalizing minority voices.

Jing Xu, Da Ju, Margaret Li, Y-Lan Boureau, Jason Weston, and Emily Dinan. 2021b. Recipes for safety in open-domain chatbots.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2020. Fine-tuning language models from human preferences.