Richard Ma

Dataset curation:

Input prompts for the Causal Language Models (Causal LMs) were generated from public domain 19th and 18th century novels originally written in English. These were: *Bleak house* (Charles Dickens, 1853), *Clarissa Harlowe; or the history of a young lady* (Samuel Richardson, 1748), *David Copperfield* (Dickens, 1853), *Middlemarch* (George Eliot, 1872), *Moby Dick* (Herman Melville, 1851), *The Mysteries of London* (George Reynolds, 1846), *The Pickwick Papers* (Dickens, 1837), *The Tenant of Wildfell Hall* (Anne Bronte, 1848), *The Way We Live Now* (Anthony Trollope, 1875), and *Varney the Vampire* (James Malcom Rymer, 1845). *Wuthering Heights* (Emily Bronte, 1847) was included in the initial raw dataset, but no prompts originating from it were ultimately processed by the Causal LMs due to resource limitations.

Every sentence in the works were considered for addition to the prompts dataset. Sentences are defined as strings separated by periods (.). Tokens within a sentence are distinguished as strings separated by spaces. The final half of a sentence's tokens are discarded to create a prompt. Prompts with fewer than three tokens are discarded. In prompts with more than five tokens, tokens other than the initial five are discarded. As such, all prompts in the prompt dataset have three, four or five tokens. This translates to three, four or five recognizable words in the vast majority of cases, although in some cases irregular formatting in the novel can lead to exceptions.

As an example, the phrase '*He desires to paint you the dreamiest, shadiest, quietest, most enchanting bit of romantic landscape in all the valley of the Saco*' (Melville 1851) is used to generate the prompt, '*He desires to paint you'*.

Four open-source Causal LMs were used to make predictions using the prompts: BLOOM (1) with 1.1 billion parameters, TII's Falcon 1 billion parameters (2), with Meta AI's OPT with 350 million parameters (3), and Distil GPT2 (4). Models were implemented in Python using the Hugging face framework (5). Greedy search was used in all cases. Autogressive Causal LM predictions after the first returned period token were discarded, thus forming complete sentences from the prompts. Because the goal is to train a classifier to identify which casual LM completed a prompt, a vertical bar (|) was used to separate the prompt and LM prediction to implicitly teach models that the prompt itself has no bearing on which LM made the prediction.

For example, the prediction made by Falcon on the prompt *'What is the chief'* is stored as *'What is the chief| advantage of owning a VPS.'*

Available computational resources were used continuously to run these casual LM predictions on the prompt dataset for a period of time. Due to resource limitations, not every prompt received a prediction, and different LMs made different amounts of predictions. With more resources, a future improvement would see predictions made for all prompts for all LMs. Resource limitations also informed the choice of the number of LMs.

Dataset separation into training, testing and validation sets was conducted by the following method: The first 1250 prompt-prediction pairs for each novel were added to the testing set, the next 1000 pairs added to the validation set, and all remaining pairs added to the training set. This resulted in an overall testing:training+validation split of **19.6:80.4**, and a validation:training split of **18.9:81.1**. This method was utilized in order to keep all predictions

made on a given prompt within a given dataset. For example, we may desire that the four different LM predictions on the prompt *'What is the chief'* remain within the training set, because one LM's prediction on this prompt in the testing set could result in prior knowledge bias for a model trained on other predictions for the same prompt. Because of this constraint, and because the different LMs made different numbers of predictions, inconsistent distributions can be seen in the various datasets shown in **Table 1**. This problem is a further motivation to improve uniformity of dataset distribution in future training, given more resources.

	Bloom	**Distil GPT2**	**OPT**	**Falcon**	**Total**
Testing	7,500	10,000	11,250	8,159	36,909
Validation	6,000	7,609	9,000	6,000	28,609
Training	18,482	37,483	46,781	19,794	122,540
Total	31,982	55,092	67,031	33,953	**188,058**

Table 1: The distribution of the total 188,058 prompt-prediction pairs among the four casual LMs and among the testing, validating, and training datasets.

Classifier Model Construction:
The best-trained model, a Keras neural network, has the architecture shown in Figure 1.

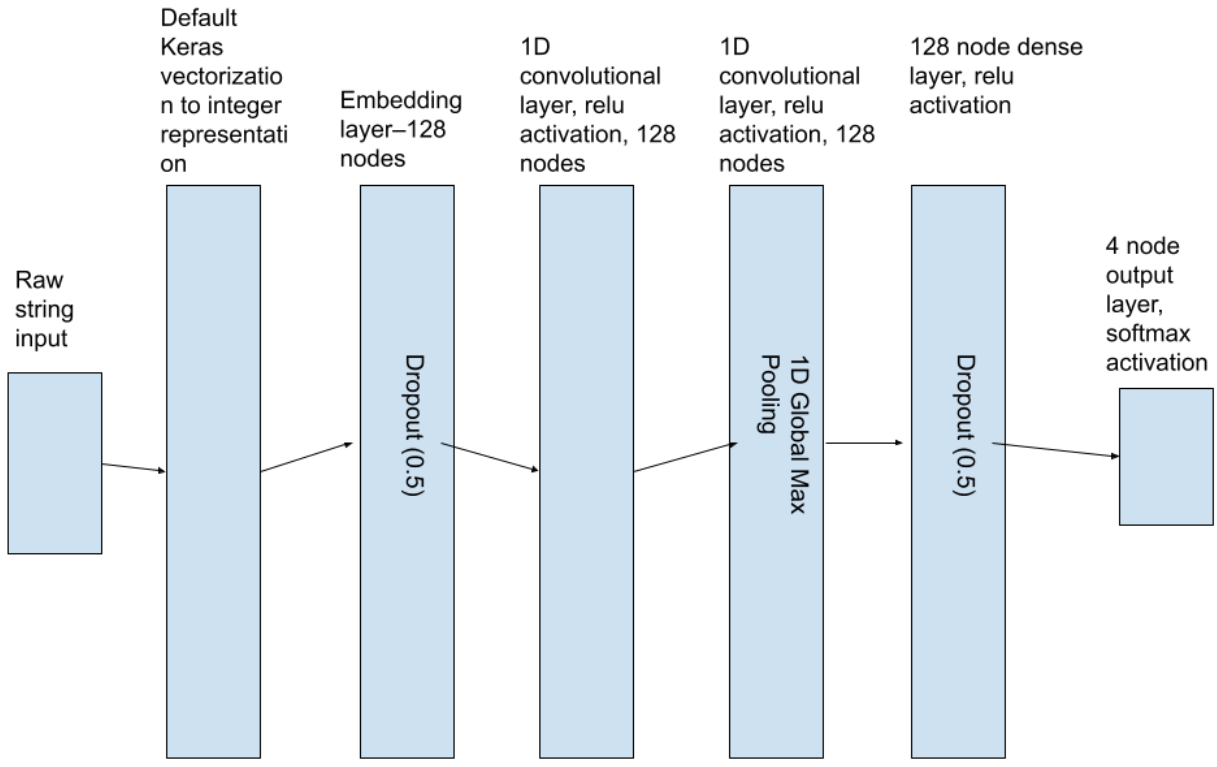

Figure 1: Schematic of neural network architecture used for classification of dataset

The data was split into batch sizes of 128 points. For fitting, ADAM optimizer was used with sparse categorical cross entropy loss function and accuracy metric. 3 training epochs were used. The starting framework for the model was inspired by a Keras tutorial example for binary text classification. (6)

Results and analysis:

The aforementioned best-performing model achieved a classification accuracy of 60.22% on the testing dataset of 36,909 points. In the immediate region of hyperparameter space around this best-performing model, there was little sensitivity noted to hyperparameters changes. The initial model tried (different from the best model by a batch size of 32) had a testing classification accuracy of 59.6%. In attempting increasing training epochs to 4 or 5, in adding additional convolutional layers and dense layers, and in changing batch size no model was noted to predict with below 59.23% accuracy on the testing set.

This value shows that there are fundamental and/or systematic differences in the responses of the four LLMs that are implicitly identifiable by the classifier, given that the testing dataset is separate from and shares no prompts in common with the training and validation sets. Without any knowledge, randomly guessing would yield at best a ~25.7% accuracy (numerically calculated) if the relative distributions of the labels in the testing set were known.

Ground-truth Bloom responses in the testing set were predicted by the classifier 55.9% of the time. The values are 61.41% for DistilGPT, 59.9% for OPT and 63.08% for Falcon ground truth responses. No large differences are shown in these prediction accuracies, indicating that these models may differ relatively equally from each other in terms of prompt completion style.

It had previously been hypothesized that OPT predictions would see the largest identification accuracy due to the notable higher frequency of vulgar language in its predictions. A future investigation could use a vulgar language dataset to evaluate the frequency of these inclusions and see if there is a statistically significant effect on prediction accuracy if they are included.

Discussion of related work:

The problem of detecting which Causal LM filled in a given prompt is closely related to a pressing issue–that of distinguishing between human-written text and machine generated text. A review paper by Wu et al. (7) discusses the issue. They introduce with the observation that humans are generally incapable of distinguishing text generated by modern models and can identify with at best only slightly more accuracy than that compared to randomly guessing. Similarly to how, given a large pool of data, a human is able to distinguish between distinct patterns in the different LLMs (such as a larger use of vulgar language and reference to modern politics and news by OPT) in our study, the authors in the Wu et al. paper highlight that at a large scale some recurring behaviors can be seen in LLMs. All things being equal, they tend to be clearer, more logical and less biased than human writers. Differences can be seen in clearness and bias in writing, relative prevalence of different types of speech, and in the breadth of vocabulary used.

Bibliography:

1. Le Scao, T., Fan, A., Akiki, C., Pavlick, E., Ilić, S., Hesslow, D., ... & Al-Shaibani, M. S. (2023). Bloom: A 176b-parameter open-access multilingual language model.
2. Malartic, Q., Chowdhury, N. R., Cojocaru, R., Farooq, M., Campesan, G., Djilali, Y. A. D., ... & Hacid, H. (2024). Falcon2-11B Technical Report. arXiv preprint arXiv:2407.14885.
3. Zhang, S., Roller, S., Goyal, N., Artetxe, M., Chen, M., Chen, S., ... & Zettlemoyer, L. (2022). Opt: Open pre-trained transformer language models. arXiv preprint arXiv:2205.01068.
4. Sanh, V. (2019). DistilBERT, A Distilled Version of BERT: Smaller, Faster, Cheaper and Lighter. arXiv preprint arXiv:1910.01108.
5. *Hugging face – the AI community building the future.* Hugging Face –. (n.d.). https://huggingface.co/
6. (Chollet, F., & Omernick, M. (n.d.). Keras documentation: Text classification from scratch. Keras. https://keras.io/examples/nlp/text_classification_from_scratch/
7. Wu, J., Yang, S., Zhan, R., Yuan, Y., Wong, D. F., & Chao, L. S. (2023). A survey on llm-gernerated text detection: Necessity, methods, and future directions. *arXiv preprint arXiv:2310.14724*.