An Exploratory, Probe-Based Study of GPT-2 Hidden States

A Logit-Lens Visualisation

Chamod Kalupahana

Independent Researcher

October 30, 2025

Keywords— mechanistic interpretability, logit lens, GPT-2, probing

Code & Colab: GitHub | Colab

Abstract

AI is increasingly shaping our world, yet we still refer to AI systems as black boxes, we provide qualitative, probe-based visualizations that can build intuition, not definitive claims. This experiment shows a method to represent the hidden internal states of the model as human-readable tokens using a logit lens.In one prompt, we found that OpenAI's GPT-2 model consistently internally *thought* that Donald Trump hates gays in a single layer's top-1 probe prediction. We believe this is due to the nature of GPT-2's WebText training dataset, which contains non-spam outbound links from Reddit, which is a progressive platform and contains substantial criticism against Donald Trump. However, we believe this is an anecdotal, probe-dependent association rather than evidence of the model's beliefs or dataset causality. Furthermore, this result is not fully reliable as there are many inaccuracies with logit lenses. Mainly that they output tokens from internal layers (especially early layers) are an approximation of the model distribution and should be treated as probe probabilities. Furthermore, there is possible miscalibration with the final softmax layer and optimistic bias from teacher forcing generation of tokens. Due to the limitations of this experiment, all findings here are qualitative illustrations, not statistical evidence. For replication and further research, we have ensured that the full source code is available here and can be run with a non-GPU Google Colab instance.

1 Introduction

1.1 State of AI in 2025

AI is one of the most growing, hyped and important fields of the last few years. Ever since the public release of OpenAI's ChatGPT in late 2022 [1], the field has been rapidly advancing. As of this writing, in late 2025, OpenAI's GPT-5 is public and several forms of agentic AI models are starting to be rolled out to the public [2] and we're expected to spend much more time and money with AI.

Despite the ever-growing usage of these AI models, we continuously refer to them as black boxes. We are repeatedly told that we cannot *look* inside these models, that they are just numbers which only makes sense to machines. If we want to trust these models, then any small step towards understanding their internals will benefit us greatly, for understanding their capabilities and grasping their consequences going forward. This is why investigating AI models with a logit lens is important.

1.2 What is a logit lens

Logit comes from statistics, simply the log of a probability. In AI models, the intermediate computations between layers are represented by hidden vector states, which are transformed by the weights and biases; these are what the model computes over and compares. They are only designed to be readable only to the model itself. However, we can transform these hidden states into a logit by applying the Equation 1 for each hidden state $z_i^{(\ell)}$.

$$\begin{aligned} &\textbf{Binary logit:} && \operatorname{logit}(p) = \log \frac{p}{1-p}. \\ &\textbf{Logit lens (layer } \ell\textbf{):} && \tilde{\mathbf{z}}^{(\ell)} = \mathbf{W}_\ell \mathbf{h}_\ell + \mathbf{b}_\ell \in \mathbb{R}^{|V|}. \\ &\textbf{Probabilities:} && p_i^{(\ell)} = \frac{\exp\left(\tilde{z}_i^{(\ell)}\right)}{\sum_{j \in V} \exp\left(\tilde{z}_j^{(\ell)}\right)}. \end{aligned} \tag{1}$$

The final part of Equation 1 is the softmax layer at the output end of the model is what converts these raw numbers into probabilities. A logit lens maps these probabilities to the vocabulary used by the model (output tokeniser), we can inspect what tokens would be outputted by the intermediate hidden states. This effectively converts the unreadable hidden states into a readable, analysable tokens and words.

1.3 Applying a logit lens to GPT

At the time of writing GPT-5 is publicly available, while it would be fascinating to apply a logit lens on it, it is more suitable to approach a much smaller, less complex model such as OpenAI's GPT-2 to understand the basics of what the model is thinking. Furthermore, the representations should be more limited in scope than GPT-5's, which would make the investigation easier.

1.4 Training Dataset of GPT-2

Since this experiment is based on investigating GPT-2 internal states, it helps to define the training dataset to contextualise our analysis. From [3], OpenAI trained GPT-2 on a dataset called WebText, which was constructed by OpenAI by scraping non-spam outbound links from Reddit posts. They categorised a post as non-spam if it had more than three upvotes and they omitted links to Wikipedia to avoid potential overfitting. Note that the dataset did not include the content from the Reddit posts themselves, just the content of the outbound links contained within each Reddit post. We will keep this in consideration when analysing the results from our logit lens.

1.5 Internal Layers of GPT-2

Crucially for the analysis of the logit lens output, it would also be helpful to define what the layers and model architecture do. [4] shows that starting layers are for extracting phrase and clause structure and the later layers are more focused on discourse level reasoning and more contextually refined token predictions. There is also work from [5] to show that the hidden states evolve incrementally throughout the model as they propagate through the layers. Using this understanding, we can build a simple understanding of the purpose of the layers in GPT-2 and how they incrementally change the output tokens which is shown in Figure 1

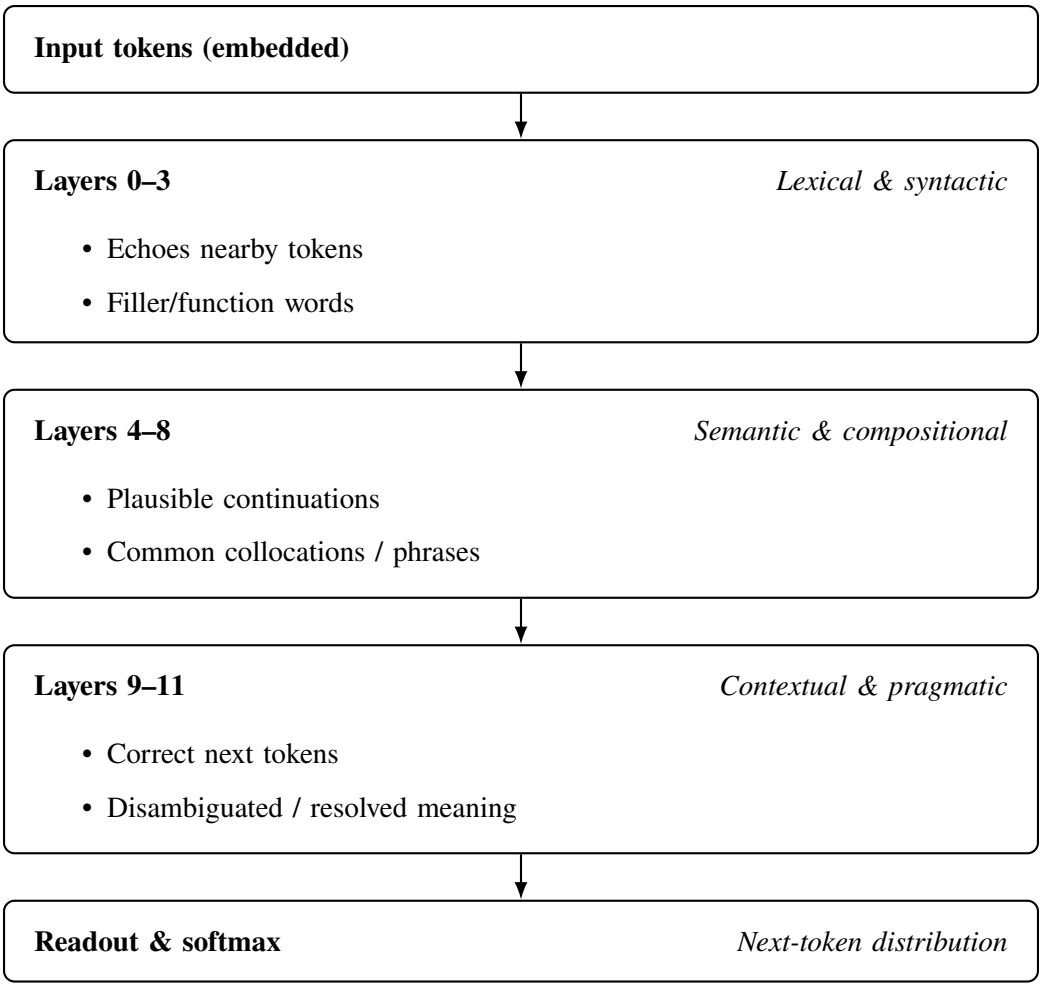

Figure 1: GPT-2 small schematic of evolving focus across Transformer layers, from local form to global meaning, culminating in the readout and softmax over the vocabulary.

2 Method

2.1 Interpreting Hidden States through the Logit Lens

As seen in Figure 1, consider the internal layers of the model $\ell_{1->12}$. For each prompt, tokens $x_{1:T}$ are encoded and passed once through the model (`output_hidden_states=True`). Each layer ℓ_n outputs hidden states as shown in Equation (2):

$$\boxed{h_\ell \in \mathbb{R}^{T\times d}.} \tag{2}$$

These hidden states are mapped into vocabulary space by a linear translator. The model outputs a matrix of logits with dimensions $[T \times |V|]$, where T denotes the number of tokens in the prompt and $|V|$ denotes the vocabulary size. This mapping produces layer-wise logits as in Equation (3):

$$\boxed{\tilde{\mathbf{Z}}_\ell = \mathbf{W}_\ell \mathbf{h}_\ell + \mathbf{b}_\ell \in \mathbb{R}^{T\times|V|}.} \tag{3}$$

Applying a softmax to these logits yields the per-layer token probabilities (Equation (4)):

$$p_\ell(t,i) = \frac{\exp(\tilde{Z}_\ell[t,i])}{\sum_{j\in V}\exp(\tilde{Z}_\ell[t,j])}. \tag{4}$$

These equations are adapted from earlier in Equation (1). No per-layer tuning or calibration was performed. From these distributions we record: (i) the *ground-truth probability* $p_\ell(t, x_{t+1})$, (ii) the *top-1 token* $\arg\max_i p_\ell(t,i)$, and (iii) the *top-k* predictions ($k = 5$) per position and layer.

We visualise three panels per prompt: a heatmap of ground-truth probabilities, a heatmap of top-1 probabilities, and a grid of top-1 tokens. The x-axes display the current token x_t (top) and gold next token x_{t+1} (bottom). All figures are generated with `Matplotlib` and exported at 300 dpi for clarity.

2.2 Prompt Engineering

Now given the method to extract hidden states from the model, we need to carefully define the tokens $x_{1:T}$ for the input prompt. Since each prompt requires its own forward pass to extract hidden states and its own compute, results and analysis, only 3 input prompts were initially defined for this experiment. The input prompts are shown in Table 1:

(a) `I would describe Brad as annoying.`	**(b)** `Apple release a new iPhone XR this year, which has a new camera!`	**(c)** `donald trump hates everything from what i hear`

Table 1: Input prompts for GPT-2 used in this experiment.

These prompts in Table 1 are ideal because they cover a variety of prompt cases. Prompt (a) is simple, testing the model's ability to predict sentence structure and what it would describe a hypothetical person, where the context is minimal. Prompt (b) discusses a factual statement about the technological company Apple releasing their annual product. In addition to predicting sentence structure, we are testing the model's ability to correctly the fetch the relevant information from within its training dataset. Finally, Prompt (b) discusses a political figure and celebrity describing what they hate. This is subjective so we are testing the model's perception of this figure and any internal biases it may have.

For the prompts, Prompt (b) and Prompt (c), we ensured that it was contained within the knowledge of GPT-2. Since GPT-2 was trained on WebText which was constructed in 2019, we can conclude that the model has no knowledge of events after 2019 [3]. Apple's iPhone XR release was in 2018 [6] and Donald Trump's initial presidency took place between 2016 and 2020 [7]. Therefore, these events should be contained within GPT-2's WebText training dataset and it should be *known* by the model.

During the experimentation of Prompt (c), we also created variations of Prompt (c) where the celebrity name was replaced while keeping the sentence otherwise identical. This was done to validate findings from Prompt (c), and the list of celebrity variations can be seen in Figure 2.

2.3 Google Colab Environment

This experiment accessed OpenAI's GPT-2 through the Hugging Face Transformers library, since OpenAI fully released GPT-2 as open-source and MIT-licensed code in November 2019 [8]. To ensure that the experiment was reproducible and reliable, all of the programming for this experiment was done and hosted on Google Colab, which also meant that we didn't have to bear the cost of compute. This experiment only used the free CPU (non-GPU) Python3 instance of Google Colab which typically provide 2 vCPUs and 12 GB RAM. The notebook for this experiment can be found here. We invite everyone to make forks and copies of the notebook for their own analysis and experimentation.

2.4 Tuned Lenses Package

While we did briefly investigate using the `transformer_utils` public library for a logit lens [9], at the time of writing, it seemed incompatible with Hugging Face Transformers and the GPT-2 model used in this experiment. Therefore, we opted for a custom written code for the logit lens as given in our colab notebook here.

3 Results and Analysis

In this section, we present the findings from our experiments and examine what the model predicted internally for different prompts.

3.1 Baseline Prompt

This subsection is to guide us through interpreting the graphs and it's meaning. Looking at Figure 3, we can see how the model's prediction progresses throughout it's layers. For the first token input [*I*], the model at layer 1 predicts [*'m*] creating the word [*I'm*] which we would expect in common English. However it quickly deviates to a comma token [,] and strongly predicts that for the next token. Looking at the ground token [*would*], we can see that neither of the model's internal predictions was correct, but they were common guesses.

We can see that this behaviour continues for each token in the prompt, with the model changing it's internal predictions as it progresses through the layers

Interestingly, we can see for the input token [*as*], the model is deciding what to describe [*Brad*] as. While we were expecting an interesting adjective, the model decides on describing him [*well*] or continuing the sentence. At least we can see that GPT-2 has no inner bias against (perhaps slightly towards) anyone called Brad, which is a relief.

Next, Figure 4 lets us investigate how accurately the model predicted the next token in the prompt. Figure 3 only shows the token for which the model had the highest probability for, whereas Figure 4 shows the probability of the ground truth token outputted by the model. We can see that the model failed to produce any meaningful non-zero probabilities for most of it's internal layer predictions. The exceptions are where the model predicts the token [*as*] for the input token [*Brad*], and ground truth token [.] for the input token [*annoying*], at layers 10. While the model doesn't finalise on these tokens for the output, both deviates at layer 11, we have captured that the model was at least *thinking* of the correct answer (for only 2 out of the 6 input tokens).

Figure 5 shows the distribution of probabilities for the model's top guess at each layer. This adds a lot of context to Figure 3, especially for the first token input [*I*].

Looking at Figure 3, we see that the model continually predicts the [,] token for most of the layers, which suggests that the model was fairly certain for its prediction. But looking at Figure 5, we can see that the model had a low probability for it's highest prediction. This actually shows that the model was uncertain about the next token. This demonstrates why it was important to generate this graph alongside Figure 3, it adds context and additional information for how the model is thinking.

3.2 Apple iPhone Prompt

Now moving onto the next prompt, we can see model's token predictions in Figure 6. Similarly to what we saw in 3, we see that the model struggles to predict a token that is relevant for the first input token, before improving its understanding from the 2nd input token onward.

The iPhone X from Apple was released in 2017 [10]. As a sanity check, we can conclude that the dataset that this version of GPT-2 was training was taken around that time. From [3], we know that the WebText dataset was created in 2019 therefore, this is consistent with (but does not establish) knowledge of 2017–2019 era phones.

The model wouldn't have actually predicted [*iPhone X*] for it's final layer output, it instead opted for a [,] token which also showcases the importance of using this logit lens to probe inside the model and discover insights like this.

Later on in the prompt, we can see that the model predicts [*Snapdragon*] for the input tokens [*which has a*]. This is a good guess by GPT-2 since Snapdragon commonly refers to smartphone CPUs however, it predicts the wrong brand of CPU used in this smartphone, since Apple used its own in-house A11 Bionic [10].

For the input token [*new*], it predicts common guesses for the next token such as [*bie*], making the word [*newbie*]. However, the model progresses through its layers, the predictions become far more contextual, understanding the prompt as a whole, and predicting tokens that make sense for a new phone to have such as [*battery*] or predicting a new [*generation*].

Looking at Figure 7, we can see that the model had a low probability for the ground tokens for most of the input tokens. Some exceptions are for the input token [*a*] where the model was confident about the ground token [*new*], especially at layers 6 and 7, before diverging at later layers to a comma token [,]

Figure 8 shows that the model varies a lot in its confidence for the next token, particularly for the input tokens [*release*] and [*has*], where the model has very strong confidence in its early layers before quickly diverging to different tokens (neither of which were the ground token).

3.3 Donald Trump Prompt

Figure 9 is particularly interesting, it starts similarly to the other tables we've produced, where it predicts [,] for the first input token. However, for the input token [*hates*], the model thinks of several predictions, including the token [*gays*] and [*Trump*], for the next token before diverging. The model then predicts [*gays*] later in the prompt, for the input tokens [*hates everything from*]. This suggests that the model internally thinks Donald Trump is likely to hate gays or himself. While this could be biases from Donald Trump himself, this could also involve specifically the dataset that GPT-2 was trained on, in the sense that it is common language for some minority to appear after the word [*hates*] in web articles. So perhaps the model has learned this behaviour. We will investigate this further when analysing Figure 2.

Figure 10 shows the model was very inaccurate for the probability of the ground token, similar to what we saw in Figure 4. The only exception was the for the input token [*hates*] where the model was relatively certain (with a probability ≈ 0.25) for the correct next token before diverging to [*gays*].

Looking at Figure 11, we can see that for the input token [*hates*], the model was uncertain about it's predictions as it progresses through the layers. This suggests that the model thinking of several predictions in Figure 9 wasn't due to the model being sure of those predictions, rather that it was sample of the predictions that the model was the least unsure about.

3.4 Investigating Other Celebrities

As an extension of Figure 9, we took the same input prompt but replaced the celebrity for each run. We can see that for layers L0 $\rightarrow$ L8, the model predicts very similar tokens for each prompt, especially since the model is building context. Layer L9 is what revealed the [*gays*] output for Donald Trump but surprisingly, we can see that the output token [*gays*] only occurs for Donald Trump. Most other celebrities get a prediction of [*everything*] or [*me*]. This suggests that actually the result that we got in Figure 9 wasn't due to seeing what GPT-2 predicts celebrities hate, but actually for seeing GPT-2 predicts what Donald Trump hates.

Top-1 Token Predictions Across Layers (Input Token: 'hates')

Celebrity	L0	L1	L2	L3	L4	L5	L6	L7	L8	L9	L10	L11
donald trump	hate	hate	fully	fully	hate	fully	fully	everything	everything	gays	Trump	the
elon musk	hate	hate	fully	fully	fully	fully	fully	everything	everything	everything	the	the
amelia earhart	hate	hate	fully	fully	fully	fully	fully	everything	everything	her	her	the
barack obama	hate	hate	fully	fully	fully	fully	fully	everything	everything	everything	his	the
serena williams	hate	fully	fully	fully	fully	fully	fully	fully	fully	me	me	the
kayne west	hate	hate	fully	fully	fully	fully	fully	everything	everything	me	me	the
taylor swift	hate	hate	fully	fully	fully	fully	fully	everything	everything	everything	me	,
keanu reeves	hate	fully	fully	fully	fully	fully	fully	fully	fully	his	him	the

Figure 2: GPT-2 logit lens table: Snapshot of Figure 9 showing the output token for the input token of [*hates*] at each layer for a variety of celebrities. Each cell represents the model's highest prediction for the next token

4 Discussion

4.1 Why does Donald Trump hate gays?

Looking at the results from Figure 9, it's unlikely that OpenAI opted for the model to learn what Donald Trump hates specifically. Looking at the training dataset from [3], we can see that it is comprised of outbound links from non-spam Reddit posts. Reddit is a modern progressive platform which contains substantial criticism and opposition to Donald Trump, especially towards the end of his presidency in 2019. Even though the dataset is not public, we can suggest an unlikely possible reason is that the websites that GPT-2 was trained also contained significant criticism towards Donald Trump and perhaps his views on homosexuals and the LGBT community.

Considering the massive volume of content and posts on Reddit, even at the height of controversy of Donald Trump's presidency, we can expect that only a small number of non-spam posts and outbound links contained anything to do with Donald Trump and even less on his views on homosexuals. Therefore, this suggests an unlikely way that a small volume of content 'poisoning' the general reasoning and knowledge of the trained model.

However, it is more likely that this could be due to an outlier. The one prompt about Donald Trump yielded a 'gays' continuation at a single intermediate layer. We do not claim causality; such outputs can reflect prompt phrasing, tokenization, the specific probe, or training-data correlations. Establishing the source would require corpus analysis and cross-model replication, which we leave to future work.

4.2 Diverging Confidence in Figure 7

Looking back at Figure 7 for the input token of [,], we can see that it predicts the next ground truth token with the most confidence around layers 5 and 6, before diverging to an incorrect token [*and*]. From the context of meaning of the layers in [4], we can see that the layers 5 and 6 are primarily focused on extracting phrase and clause structure whereas the later layers are more concerned with discourse level reasoning. This suggests that the ground truth token [*which*] was only likely when looking at the short phrasing, on the order of a few words, rather than at the order of the whole sentence.

4.3 Inaccuracies of Logit Lens

However, it is important to discuss the shortcomings of logit lenses. Firstly, the lens is not equivalent to the model's true distribution. The lens used in this experiment is a learned translator from layer states to vocabulary logits, rather it is an approximation of what the model would predict at the point. This has a greater effect on the early to middle layers of the model, since they were not trained to be predictive, but rather intermediate computations to be used in the next layer. This suggests that the tokens and probabilities that

we've been extracting from the model are probe probabilities, not the model's actual next-token probabilities [11].

Secondly, even a good logit lens translator can be mis-scaled with the softmax layers which yields wrong probabilities [12] which may have affected the output probabilities in the probability heatmaps such as in Figure 7.

Thirdly, for the generation of the logit lens token grids, such as Figure 3, the generation of predictions is different from what we can expect at model inference for a novel generation. At each token prediction, even if the model predicts the wrong token for the next token, that token isn't used for the input or context tokens for predicting the next token. Instead, the ground-truth token is used for the input token for the next prediction. This is known as teacher forcing generation, and it means that the model runs into an optimistic bias when predicting text and generating novel text, since errors of wrong output would propagate into the context and cause errors for each next token prediction when the context is not being forced into the ground truth tokens by the *teacher* [13]. Whilst the main limitation is due to probe mismatch and potential mis-scaling, we can still suggest that the tokens predictions in token grids have been overstated for how likely they are meaning they are overly optimistic.

Finally, we come against the constraints of data visualisation. In our heatmap tables, such as Figure 10, since we represent the probabilities in terms of colours, a few outliers can set the max and min of the colour scale which causes all other probabilities to appear washed out. This removes fine details and uncertainty from the scale which makes our analysis more difficult. Furthermore, we did not normalise per-figure colour scales in ways that make cross-prompt comparisons valid.

Taking this knowledge back to the results from Figure 9 suggests that these results are much less reliable than we initially concluded. Especially since the token [*gays*] only appears in one layer's prediction, the inaccuracies of logit lenses suggests that this is more likely to be a outlier than expected, reliable outputs. However, it is particularly unlikely that the results were only reproducible for Donald Trump and no other celebrities in Figure 2 being completely down to randomness and error so we can expect combination of the two factors to explain these results.

Furthermore, it also removes the ability to analyse the token predictions patterns in Figure 7 as we did before. In fact, the behaviour of the prediction diverging in can be explained by [4] when the outputs of the early layers of the model are less transparent than the later layers. We can also expect the mis-scaling with softmax layers from [12] to have an effect on these results.

4.4 Ethical Considerations

This work is an exploratory, probe-based visualization of a pretrained language model's intermediate representations. The goal is to build intuition, not to diagnose the model's *beliefs*, make causal claims about training data, or draw normative conclusions about any individual or group.

5 Conclusion

Logit Lenses are indeed a very useful tool for initally inspecting the inside *thoughts* of AI models, they provide approximate, layer-wise readouts that can aid intuition. We have shown that they are an excellent starting point for understanding how the model thinks and how tokens propagate through its layers. By just looking at the final outputs of the model, we miss out on the crucial data on *how* the model arrived at its results. This can show if the model was initially going to predict the correct answer before it decided against it as we saw in Figure 7 or what the model has learnt about a particular subject *subconsciously* as we saw in Figure 9. We refrain from causal or normative interpretations; our results are qualitative and probe-dependent.

However, we must be careful when analysing and drawing conclusions from these results, since there are

many challenges for ensuring that logit lenses extract meaningful, accurate data from within the model. Even excluding any errors and inaccuracies from the logit lens used in this experiment, there needs to be further research to ensure that the results of logit lenses are verifiable and reproducible. Hopefully the logit lens being publicly available will both help in verifying these results and providing a starting point to build a more accurate open-source logit lens for GPT-2 and other models.

If considering how to take this particular experiment further, we would expand the variety of input prompts, touching on many more topics to learn more about what GPT-2 has learned. It would be interesting to generate an expanded dataset of Figure 2 and to see if the token [*gays*] appears for any other celebrities, particularly controversial celebrities. In order to investigate further on what GPT-2 thinks about the LGBT community, it would be helpful to include more prompts covering this topic when testing.

References

[1] OpenAI. Chatgpt release. *1*, 2022. URL https://openai.com/index/chatgpt/.

[2] OpenAI. Gpt-5 release. *1*, 2025. URL https://openai.com/index/introducing-gpt-5/.

[3] Alec Radford. Language models are unsupervised multitask learners. *1*, 2019. URL https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf.

[4] Kawin Ethayarajh. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings. *1*, 2019. URL https://arxiv.org/abs/1909.00512.

[5] Javier Ferrando, Gerard I. Gállego, Ioannis Tsiamas, and Marta R. Costa-jussà. Explaining how transformers use context to build predictions, 2023. URL https://arxiv.org/abs/2305.12535.

[6] Wikipedia. iphone xr. *1*, 2025. URL https://en.wikipedia.org/wiki/IPhone_XR.

[7] Wikipedia. Donald trump. *1*, 2025. URL https://en.wikipedia.org/wiki/Donald_Trump.

[8] Hugging Face. Gpt-2. Transformers Library. URL https://huggingface.co/docs/transformers/en/model_doc/gpt2.

[9] PyPi. transformer-utils. Large autoregressive language modeling helpers. URL https://pypi.org/project/transformer-utils/.

[10] Wikipedia. iphone x. *1*, 2025. URL https://en.wikipedia.org/wiki/IPhone_X.

[11] Alexander Yom Din, Taelin Karidi, Leshem Choshen, and Mor Geva. Jump to conclusions: Short-cutting transformers with linear transformations, 2024. URL https://arxiv.org/abs/2303.09435.

[12] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015. URL https://arxiv.org/abs/1503.02531.

[13] Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. Sequence level training with recurrent neural networks, 2016. URL https://arxiv.org/abs/1511.06732.

A Appendix: Results Figures

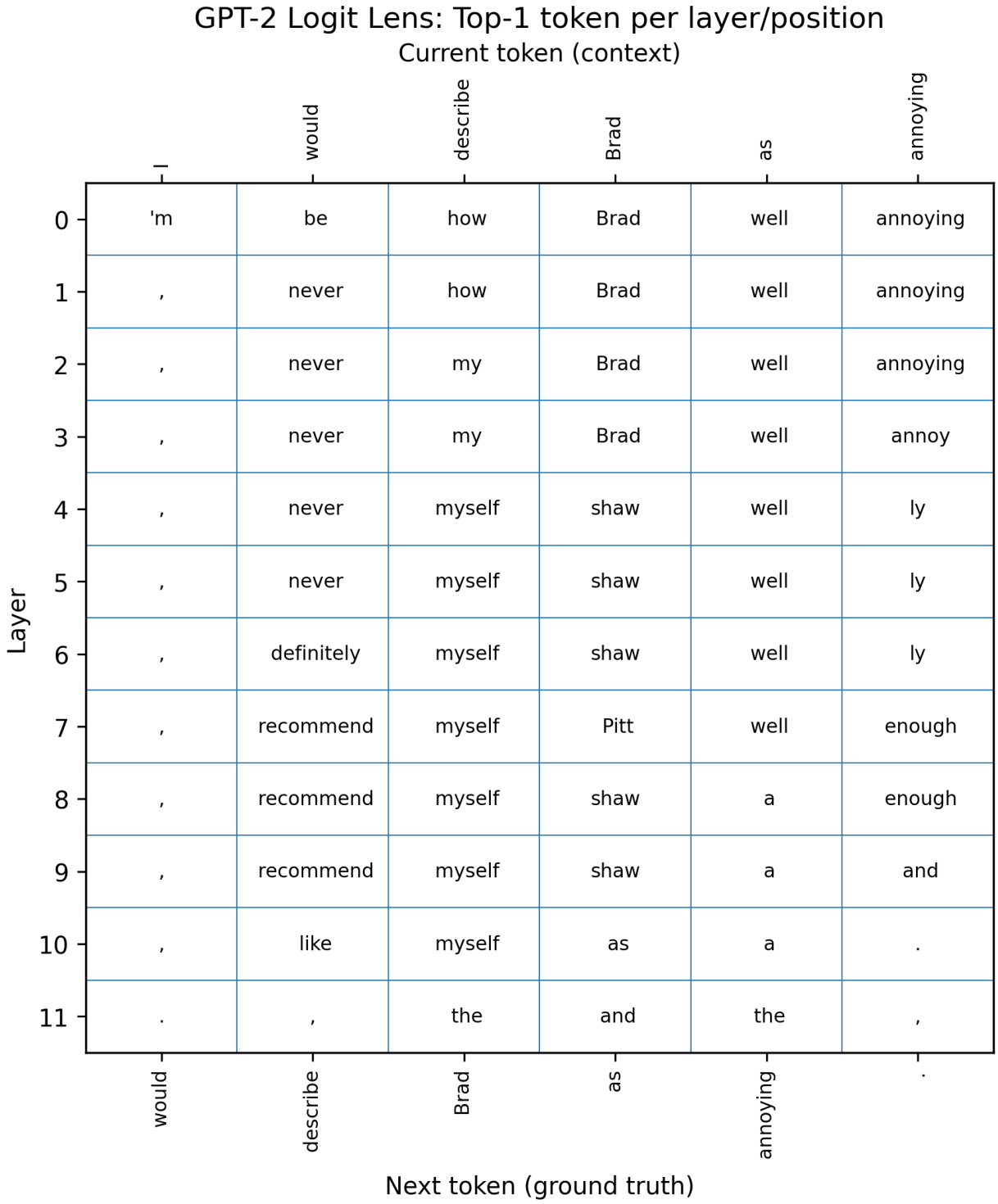

Figure 3: GPT-2 logit lens token grid: The top tokens are the input into the model and the bottom tokens are the ground truth for the next token. Each cell represents the model's prediction for the next token for the given input token and previous token. These are illustrative snapshots from one prompt; they are not aggregate statistics and should not be over-interpreted. Prompt: *I would describe Brad as annoying.*

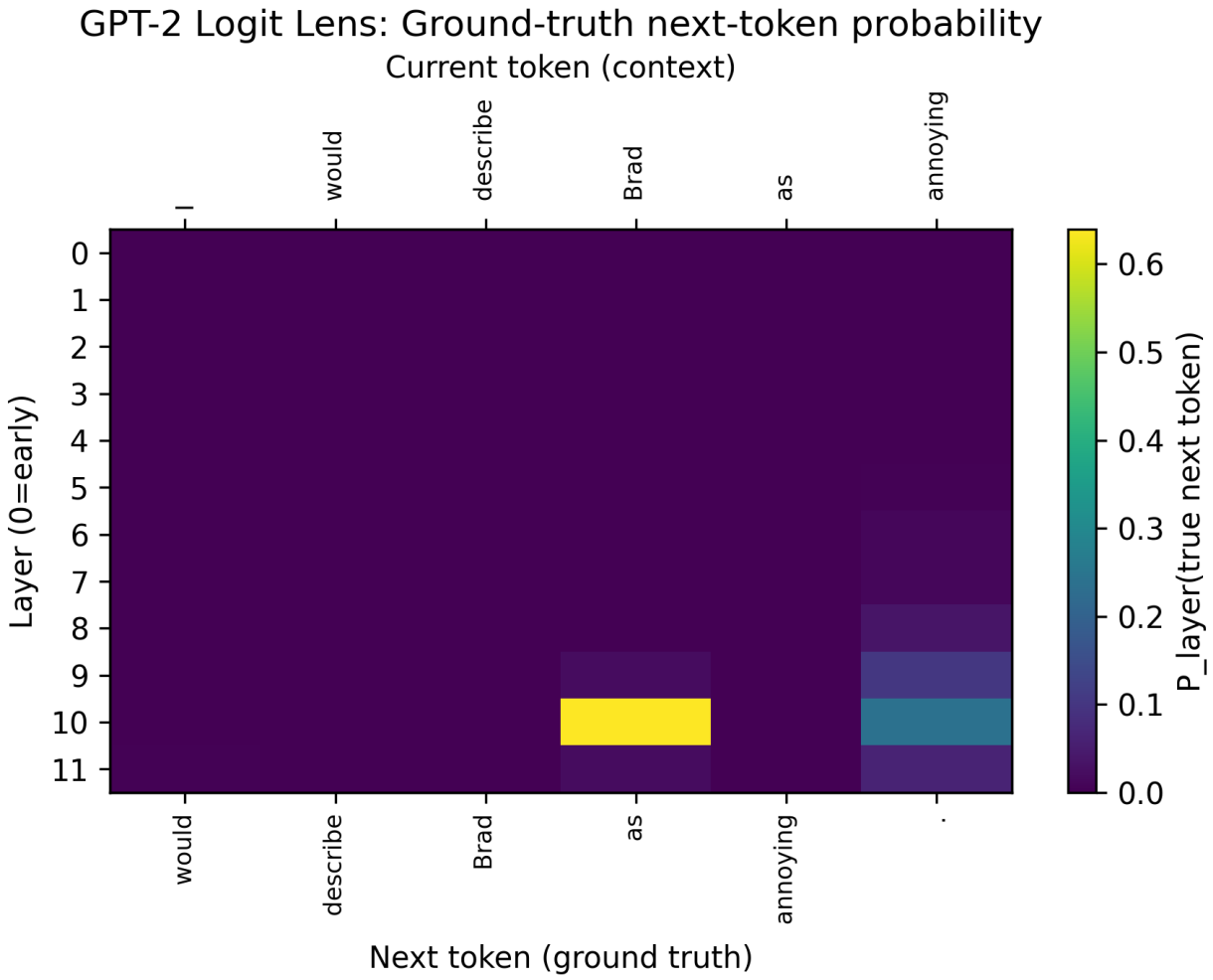

Figure 4: GPT-2 logit lens ground truth heatmap: The top tokens are the input into the model and the bottom tokens are the ground truth for the next token. Each cell represents the model output probability of the ground token, ignoring all other tokens. High probability (yellow) shows that the model predicts the ground truth token correctly while low probability (purple) indicate weak or incorrect predictions.

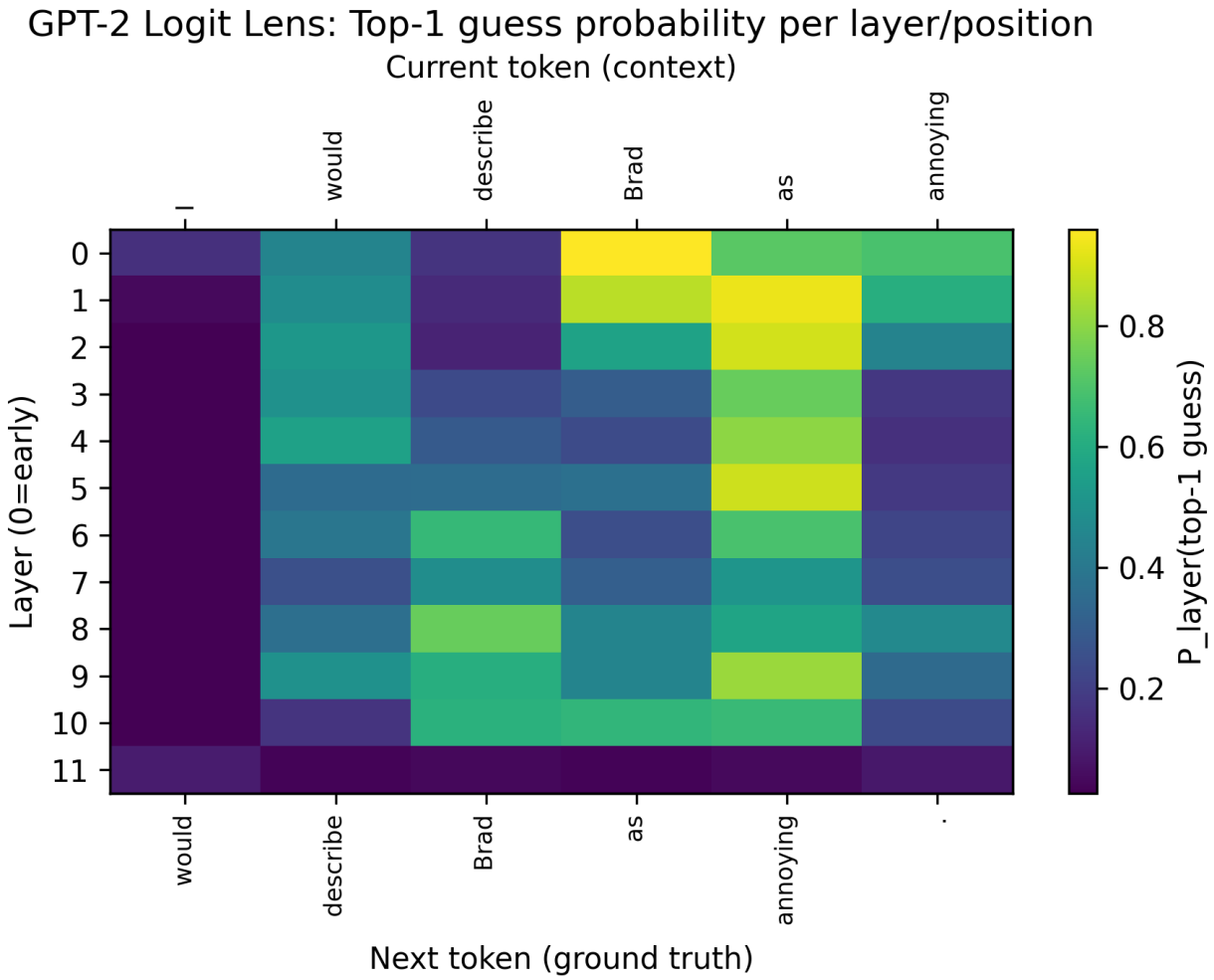

Figure 5: GPT-2 logit lens top guess probability heatmap: The top tokens are the input into the model and the bottom tokens are the ground truth for the next token. Each cell represents the model output probability of the model's most likely token for each layer. High probability (yellow) shows that the model has high certainty while low probability (purple) indicate weak certainty.

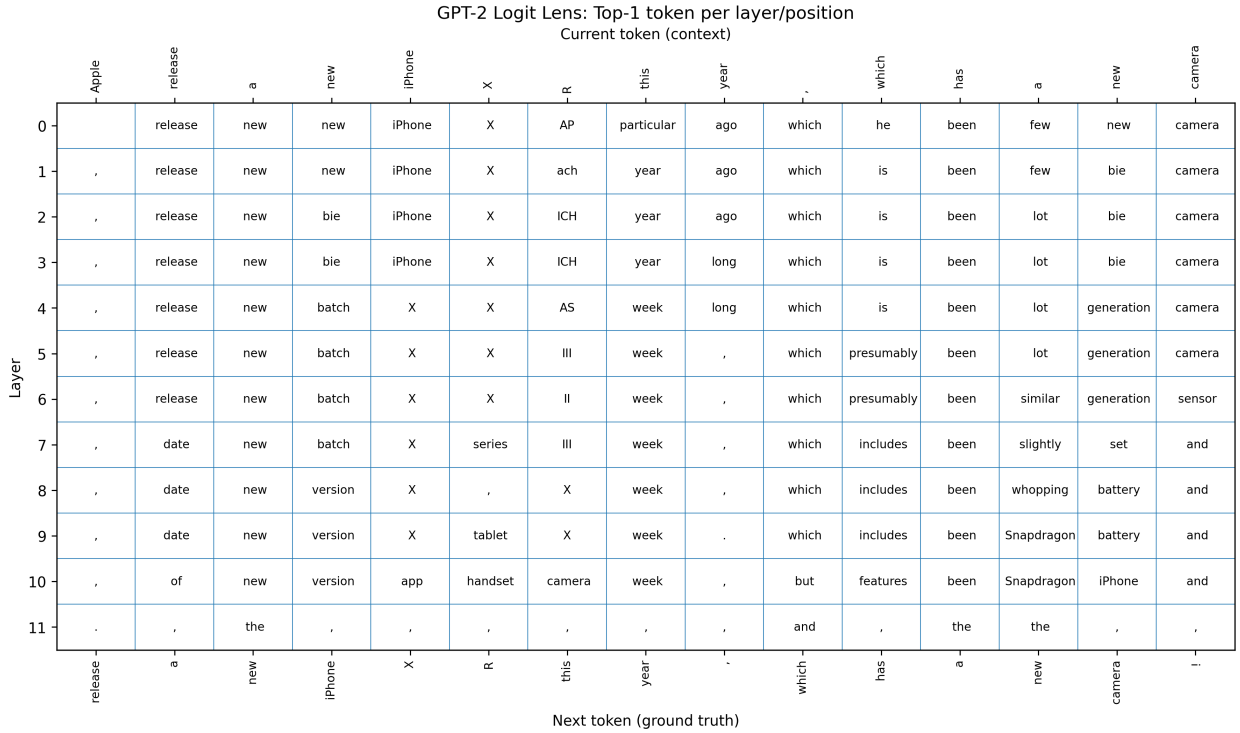

Figure 6: GPT-2 logit lens token grid: The top tokens are the input into the model and the bottom tokens are the ground truth for the next token. Each cell represents the model's prediction for the next token for the given input token and previous token. These are illustrative snapshots from one prompt; they are not aggregate statistics and should not be over-interpreted. Prompt: *Apple release a new iPhone XR this year, which has a new camera!*

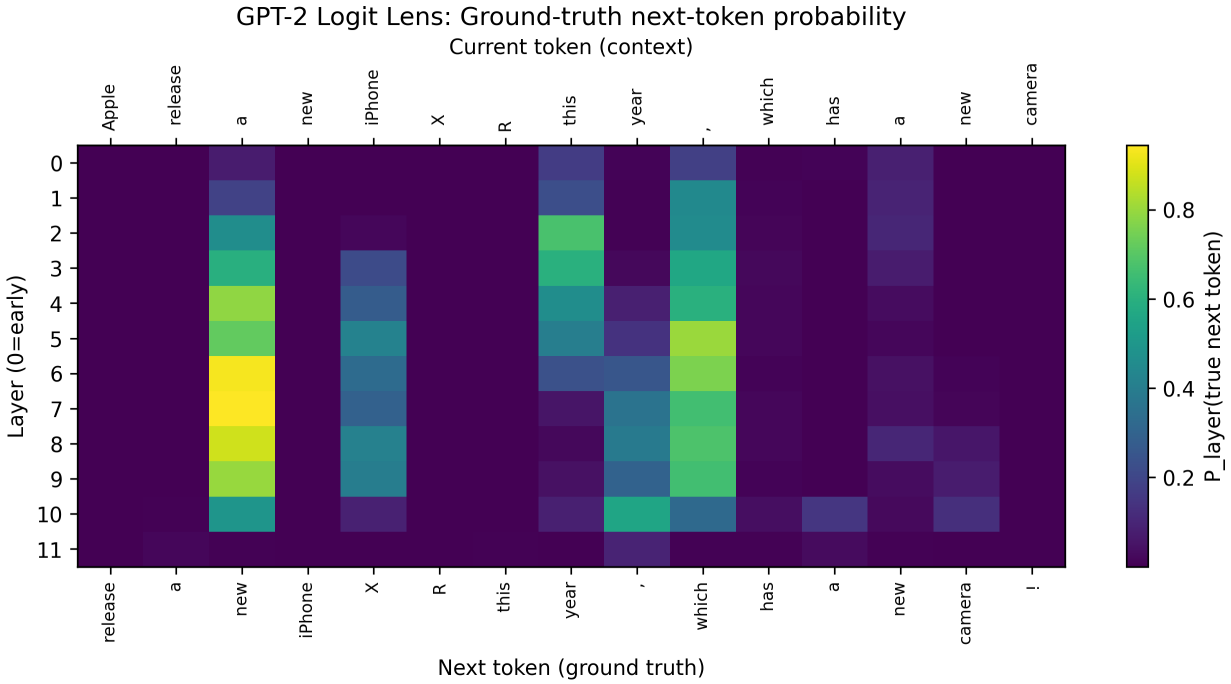

Figure 7: GPT-2 logit lens ground truth heatmap: The top tokens are the input into the model and the bottom tokens are the ground truth for the next token. Each cell represents the model output probability of the ground token, ignoring all other tokens. High probability (yellow) shows that the model predicts the ground truth token correctly while low probability (purple) indicate weak or incorrect predictions.

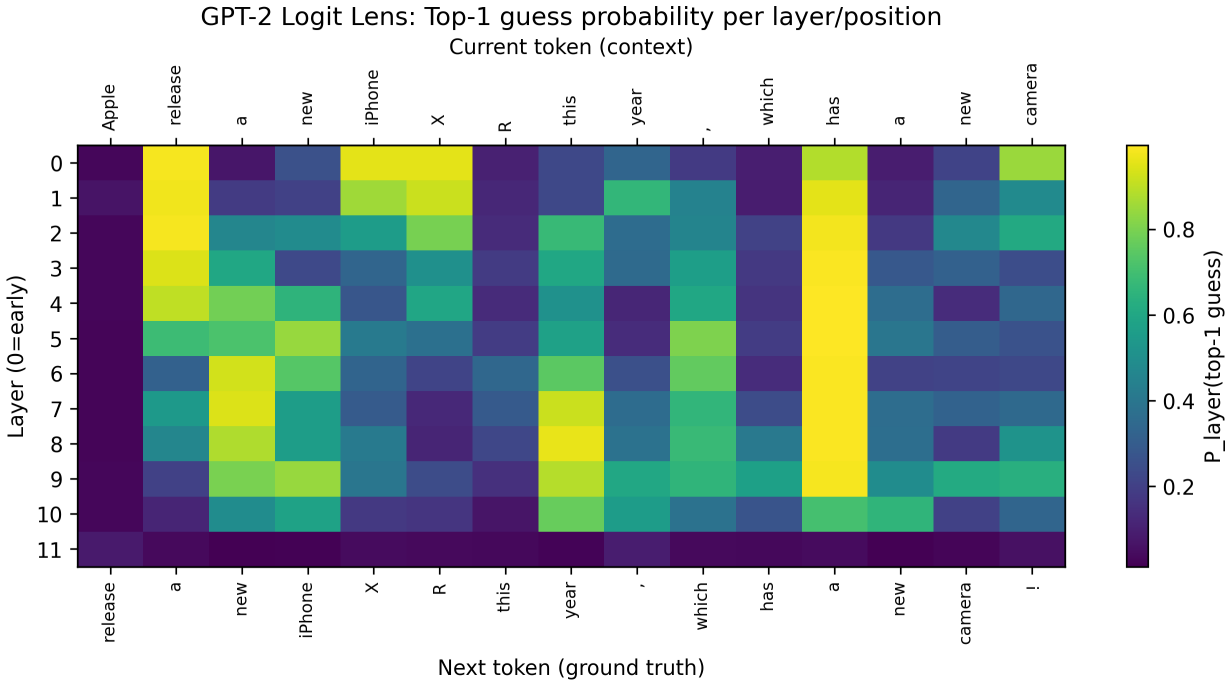

Figure 8: GPT-2 logit lens top guess probability heatmap: The top tokens are the input into the model and the bottom tokens are the ground truth for the next token. Each cell represents the model output probability of the model's most likely token for each layer. High probability (yellow) shows that the model has high certainty while low probability (purple) indicate weak certainty.

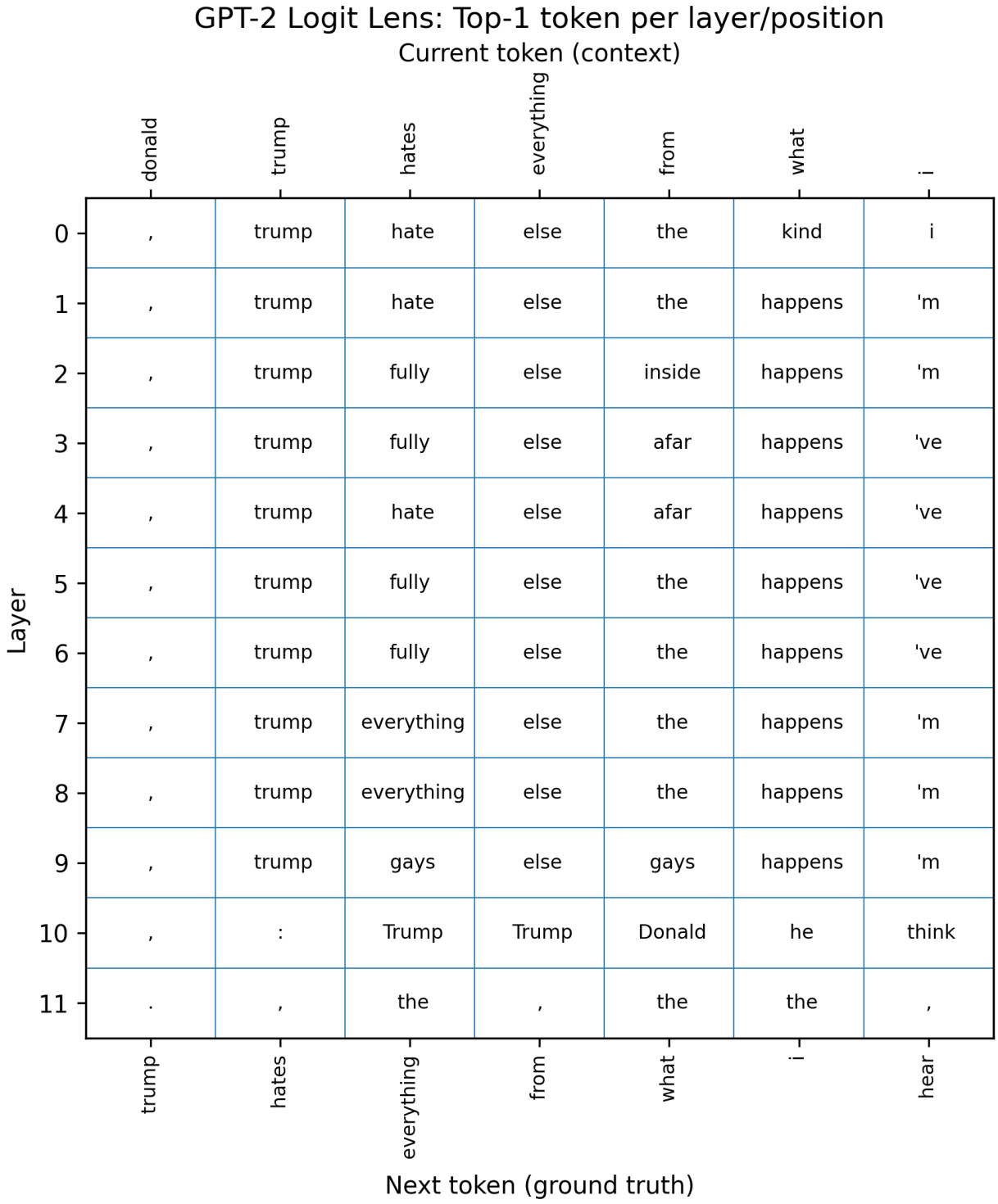

Figure 9: GPT-2 logit lens token grid: The top tokens are the input into the model and the bottom tokens are the ground truth for the next token. Each cell represents the model's prediction for the next token for the given input token and previous token. These are illustrative snapshots from one prompt; they are not aggregate statistics and should not be over-interpreted. Prompt: *donald trump hates everything from what i hear*

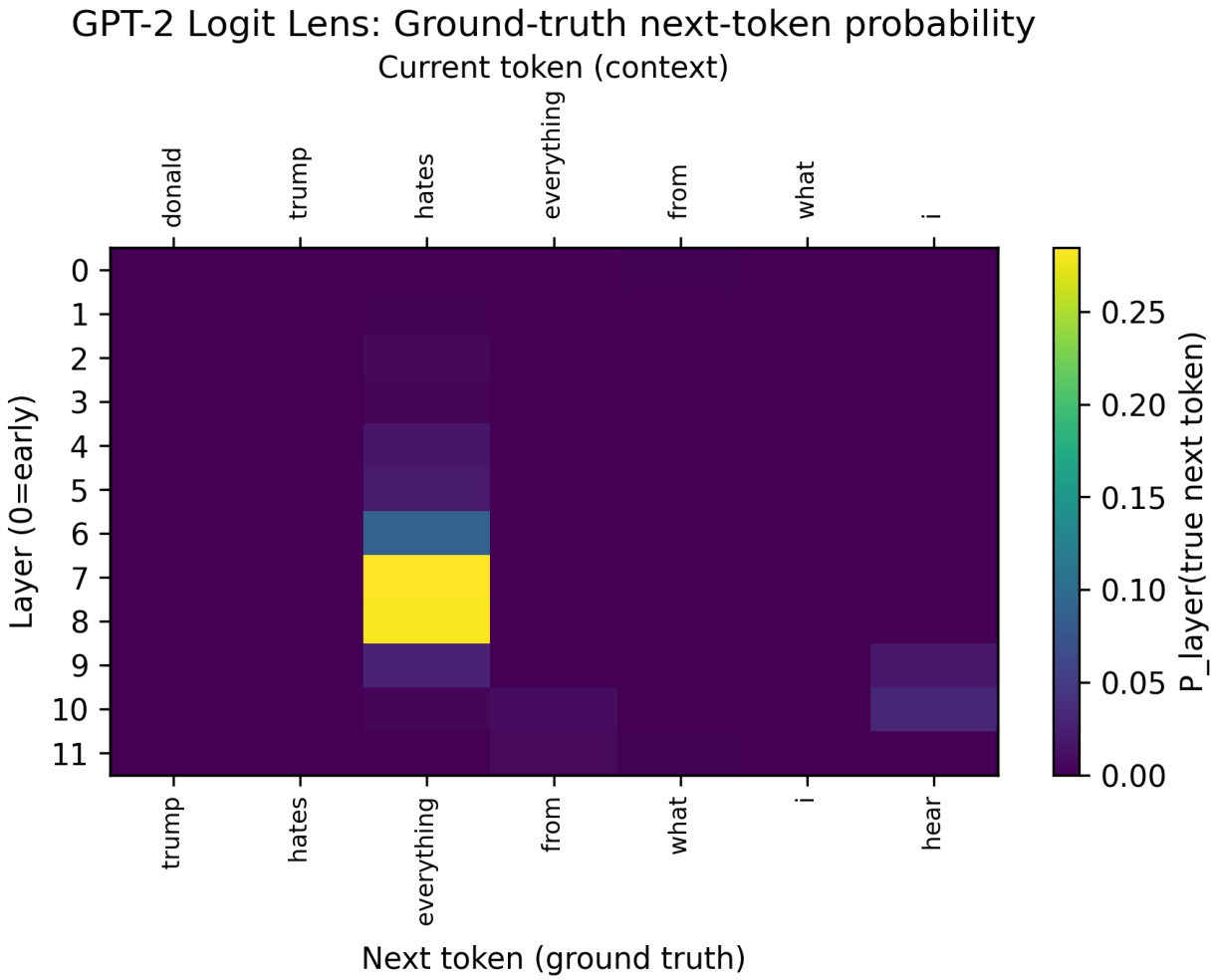

Figure 10: GPT-2 logit lens ground truth heatmap: The top tokens are the input into the model and the bottom tokens are the ground truth for the next token. Each cell represents the model output probability of the ground token, ignoring all other tokens. High probability (yellow) shows that the model predicts the ground truth token correctly while low probability (purple) indicate weak or incorrect predictions.

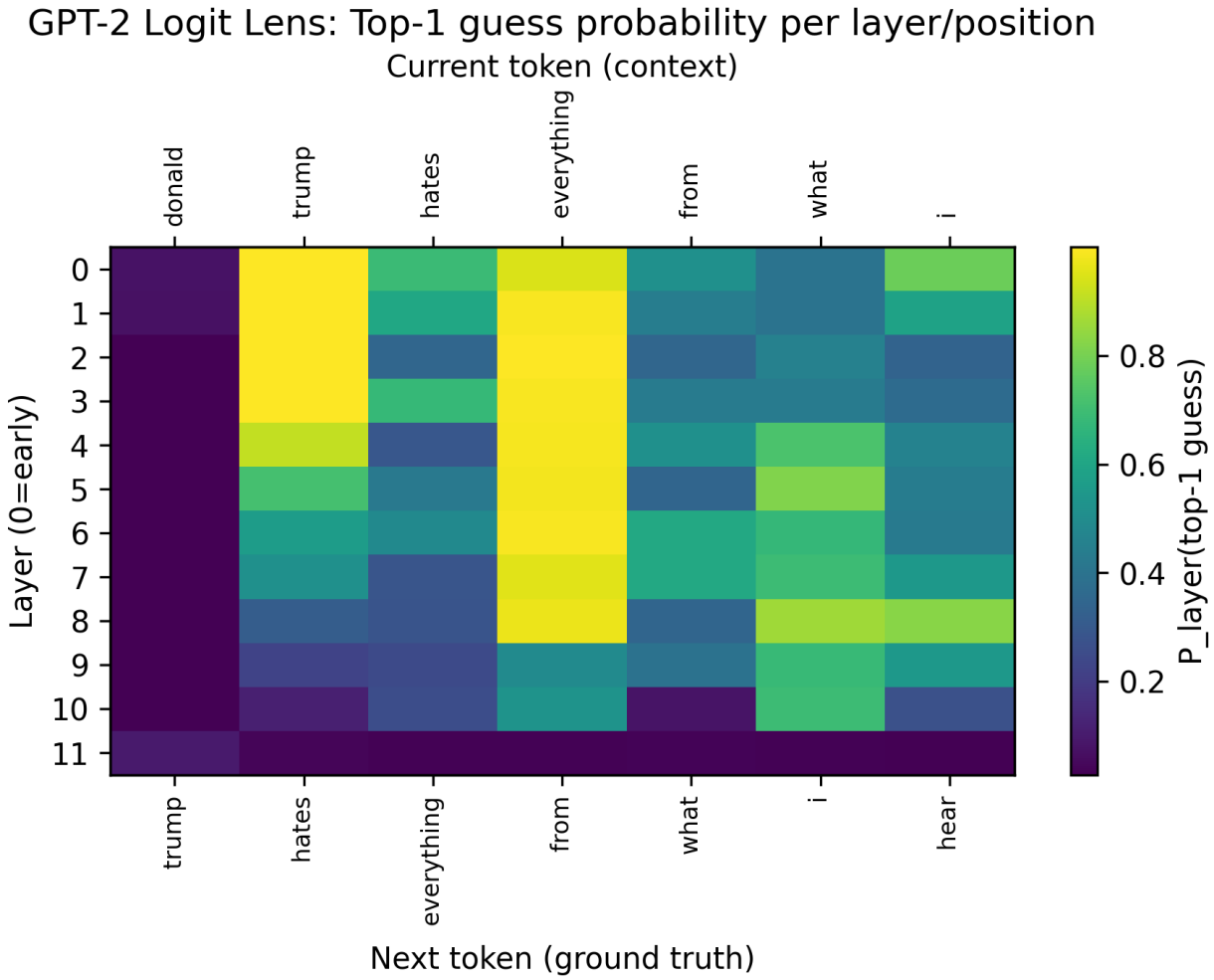

Figure 11: GPT-2 logit lens top guess probability heatmap: The top tokens are the input into the model and the bottom tokens are the ground truth for the next token. Each cell represents the model output probability of the model's most likely token for each layer. High probability (yellow) shows that the model has high certainty while low probability (purple) indicate weak certainty.