Finding Circuit for Singular To Plural Transformation with GPT2-XL

Varchita Lalwani

August 2024

1 Introduction

The transformation of singular nouns to their plural forms in English can follow various patterns, ranging from simple suffix additions to more complex vowel changes. This study focuses on leveraging the GPT-2 XL model to understand the intricacies of these transformations. The model's ability to predict plural forms is assessed by classifying singular nouns into four distinct categories: nouns ending with "s," nouns ending with "es," nouns with internal vowel changes, and nouns that remain the same in both singular and plural forms. By employing a series of ablation studies and attention analyses, the aim is to identify the internal circuits within the model that are responsible for these specific transformation tasks. The research also explores the impact of different sentence templates and the role of key structural components, such as layer normalization and attention heads, in enhancing or degrading the model's prediction accuracy.

2 Highlights of the Results:

1. **Distinct Pluralization Patterns Analyzed:** The study categorized singular nouns into four types based on their pluralization rules: nouns ending in "s," nouns ending in "es," nouns undergoing vowel changes, and nouns that remain the same. The main focus is on plurals ending with 's' and the obtained circuit is checked for generalization of other types of plurals.

2. **Layer-Specific Ablation Findings:** The ablation studies demonstrated that layers 6 to 11 of the GPT-2 XL model are crucial for maintaining high prediction accuracy in singular-to-plural transformations. Ablating these layers significantly reduced the model's ability to predict correct plural forms.

3. **Critical Role of Attention Heads and Layer Normalization:** Attention heads within layers 6 to 11 were identified as critical for processing specific tokens related to pluralization tasks. The study found that removing layer normalization from these layers resulted in a significant collapse of the model's predictive capabilities, highlighting its essential role in maintaining model stability and accuracy.

4. **Importance of Sentence Structure:** The experiments showed that the structure of the input sentence plays a vital role in guiding the model's predictions. Variations in sentence templates and the presence or absence of specific words (such as "The," "of," and "is") significantly impacted the model's ability to correctly predict plural forms. The inclusion of a beginning-of-sentence (BOS) token generally improved accuracy by providing a clearer starting context for the model.

5. **Attention Rollout Analysis:** Attention rollout revealed that layers 6 to 11 primarily maintain individual token information rather than facilitating interactions between different tokens. This insight suggests that these layers are more focused on preserving the identity of each token, which is crucial for accurate pluralization.

6. **Redundancy and Robustness in the Model:** The model exhibited a degree of redundancy and robustness, as ablating a single layer or small groups of layers did not drastically affect the output.

However, as more layers were ablated, the number of correct predictions decreased, indicating that while some layers have overlapping functions, others are more critical for specific tasks.

7. **Head-Specific Ablation Finding for Layers 6-11:** In layers 6-11, the focus was on maintaining individual token information. The next step involved identifying which attention heads were crucial by averaging attention values over token information. This process helped pinpoint the most important heads for each token, which are essential for forming the model’s internal circuits.

8. **Generalization Across Different Plural Forms:** The study found that ablating layers 6 to 11 reduced prediction accuracy across all categories of pluralization rules, suggesting that these layers are broadly important for handling a variety of plural forms beyond just those ending in ”s.”

3 Method for Sentence Generation Using GPT-2

The sentences are generated following a specific template and are divided into four distinct categories based on the pluralization pattern of the nouns. The base template used for generating sentences is: ”The plural of [singular noun] is”, [plural] needs to be predicted by GPT2. In this template, [singular noun] is the noun in its singular form. Categories of Pluralization Rules. The plural forms of nouns in English can vary widely, and the generated sentences are grouped into four categories based on the specific rules that apply to the noun’s transformation from singular to plural. Data generation is present in **singular_plural_dataset_building.ipynb** notebook

Nouns Ending with ”s”: In this category, the plural form of the noun is created by simply adding an ”s” to the end of the singular form. For example:
Singular: house → Plural: houses

Nouns Ending with ”es”: This category includes nouns that require adding ”es” to the singular form to make them plural. For example:
Singular: box → Plural: boxes

Nouns Ending with a Vowel Change: Some nouns change their internal vowel when converting from singular to plural. For example:
Singular: man → Plural: men Nouns That Remain the Same in Singular and Plural: Certain nouns do not

change at all when moving from singular to plural. The noun looks identical in both forms. For example:
Singular: sheep → Plural: sheep

There are total 127 singular words where plural will end with ’s’ in this dataset.
singular_s = [’cat’, ’dog’, ’book’, ’chair’,’house’, ’car’, ’apple’, ’pen’, ’desk’, ’cup’, ’hat’, ’tree’, ’flower’, ’computer’, ’window’, ’river’, ’ocean’, ’friend’, ’neighbor’, ’teacher’, ’doctor’, ’actor’, ’artist’, ’athlete’, ’bird’, ’toy’, ’key’, ’taxi’, ’piano’, ’guitar’, ’violin’, ’language’, ’song’,’ball’, ’game’, ’movie’, ’market’, ’hotel’, ’classroom’, ’plate’, ’knife’, ’fork’, ’spoon’, ’ring’, ’shirt’, ’skirt’, ’shoe’, ’hand’, ’head’, ’eye’, ’ear’, ’nose’, ’mouth’, ’finger’, ’toe’, ’knee’, ’elbow’, ’shoulder’, ’leg’, ’arm’, ’wing’, ’root’, ’planet’, ’star’, ’universe’, ’cloud’, ’sun’, ’moon’, ’road’, ’paper’, ’page’, ’laptop’, ’pencil’, ’top’, ’cover’, ’table’, ’pot’ ’scale’, ’bottle’, ’sticker’, ’wire’, ’fruit’, ’train’, ’plane’, ’arrow’, ’fan’, ’charger’, ’human’, ’peanut’, ’nut’, ’bolt’, ’almond’, ’male’, ’food’, ’trip’, ’mountain’, ’line’, ’bulb’, ’filter’, ’tire’, ’rose’, ’cap’, ’day’,’rope’, ’chain’, ’word’, ’click’, ’number’, ’image’, ’set’, ’model’, ’circuit’, ’clip’, ’stage’, ’ribbon’, ’board’, ’feather’, ’color’, ’phone’, ’printer’, ’plant’, ’notice’, ’vice’, ’place’, ’race’, ’choice’, ’spice’, ’service’]

There are total 41 singular words where plural will end with 'es' in this dataset.
singular_es = ['hero', 'echo', 'potato', 'tomato', 'box', 'church', 'class', 'church', 'watch', 'bus', 'dish', 'match', 'brush', 'buzz', 'dress', 'glass', 'fox', 'cross', 'tax', 'branch', 'bunch', 'wish', 'crash', 'gas', 'pass', 'guess', 'flash', 'press', 'mass', 'focus', 'lens', 'kiss', 'circus', 'nexus', 'city', 'country', 'company', 'leaf', 'library', 'galaxy', 'greenfly']

There are total 64 singular words where plural will remain same in this dataset.
singular_same = ['aircraft', 'eyeglasses', 'news', 'mews', 'sheep', 'alms', 'gallows', 'monkfish', 'shellfish', 'barracks', 'goldfish', 'moose', 'shorts', 'binoculars', 'grapefruit', 'shrimp', 'bison', 'offspring', 'smithereens', 'bourgeois', 'grouse', 'pants', 'spacecraft', 'breadfruit', 'haddock','species', 'cattle', 'halibut', 'patois', 'squid', 'chalk', 'headquaters', 'pilers', 'staff', 'chassis', 'hovercraft', 'police', 'starfruit', 'chinos', 'swine', 'cod', 'insignia', 'premises', 'tongs', 'corps', 'jackfruit', 'pyjamas', 'crossroads', 'jeans', 'reindeer', 'trout', 'deer', 'kennels', 'rendezvous', 'knickers', 'salmon', 'tweezers', 'dungarees', 'leggings', 'scissors', 'wheat', 'elk', 'series', 'you']

There are total 31 singular words where plural will change to vowel in this dataset.
singular_vowel = ['man', 'woman', 'child', 'tooth', 'foot', 'goose', 'mouse', 'louse', 'person', 'ox', 'die', 'penny', 'cactus', 'focus', 'fungus', 'thesis', 'analysis', 'basis', 'crisis', 'hypothesis', 'parenthesis', 'synthesis', 'ellipsis', 'phenomenon', 'criterion', 'datum', 'medium', 'index', 'matrix', 'vertex', 'stratum']

4 Accuracy with GPT2 models

The table displays the total number of correct predictions for each category of plural types out of the total data points in each category across four different GPT-2 models without any fine-tuning and with prepend_bos = True. For the **GPT-2 Small** model, the majority of the predictions were characters like 'a', spaces, and quotation marks. For the **GPT-2 Medium** model, most of the predictions consisted of the same singular words as the input. Based on these observations, I decided to focus on exploring the patterns in the **GPT-2 XL** model. The accuracy is present in **singular_plural_prediction.ipynb** notebook

GPT2	es	s	vowel	same
Small	0	0	0	0
Medium	0	0	0	0
Large	12	60	7	19
XL	19	80	10	22

Table 1: Pluralization performance of different GPT-2 models

5 Knockouts

I am focusing my analysis on identifying circuits within the model that are specifically responsible for handling singular nouns whose plural forms end in 's'. To achieve this, I am using **mean ablation**, targeting the nouns tokens as these are the elements that vary across all sentences in the dataset.

Knockouts refer to a technique where specific components of a model, such as neurons, attention heads, or even entire layers, are systematically disabled (or "knocked out") to study their contribution to the model's overall behavior. This method helps researchers understand how different parts of a neural network contribute to specific tasks or outputs.

The primary goal of using knockouts is to identify the internal ”circuits” within a neural network that correspond to specific functions or behaviors. For example, in a language model like GPT-2, researchers might use knockouts to determine which neurons or attention heads are responsible for understanding grammar, handling specific vocabulary, or even maintaining coherence in text generation.

5.1 Mean Ablation on GPT2-XL

1. The dataset covers all four categories of pluralization rules: ending in ’s’, ending in ’es’, vowel changes and remain the same in plural and singular forms.

2. I ran the GPT2-XL model across all the sentences in the dataset and extracted the model weights specifically for the noun tokens in each sentence.

3. After obtaining the weights for the noun tokens, I computed the mean of these weights across all sentences in the dataset. The averaging was done for all 48 layers of the model and for each of the 25 attention heads per layer.

4. As a result, I have mean ablation arrays that captures the average weight contributions of the noun tokens across all layers and attention heads. The code for this is in **get_mean_for_knockouts.ipynb** notebook.

As described above mean ablation is done to understand the role of different layers in GPT2-XL. By systematically ablating (removing or disabling) layers, either individually or in groups, we can observe how the model’s predictions are affected. This allows us to identify which layers are critical for maintaining accurate predictions and which layers have less influence.

Methodology

1. **Layer Ablation Process:** Started with 48 layers in the model, each having 25 arrays (heads). The ablation process is performed on these layers to analyze their impact on the model’s prediction capability. The ablation is performed in an incremental manner. Initially each layer is ablated one at a time. The number of layers ablated together is gradually increased up to 8 layers at a time. For each combination, the ablated layers are chosen in non-overlapping intervals to ensure a comprehensive analysis of all potential interactions.

2. **Tracking Common Predictions:** For each set of ablated layers, the number of common predictions between the ablated model and non-ablated (original) model is tracked. Say, if the original model has 80 predictions and after ablating a certain combination of layers, the ablated model has 78 common predictions. The code for this is in **singular_plural_prediction _ablation_study_multiple_layers.ipynb** notebook

3. Table 2 presents results for different numbers of layers ablated (referred to as Gap). For each Gap, the number of common predictions and the specific layers that were ablated is mentioned. This shows which combinations of layers, when ablated, result in fewer changes to the model’s predictions.

Key Observations:

1. **Minimal Impact with Small Gaps:** When a single layer (Gap - 1) is ablated, there are no significant changes in the model’s predictions. The number of common predictions remains high (78, 79, or 80 out of 80), suggesting that the ablation of a single layer does not drastically affect the model’s output. This indicates a level of redundancy or robustness in the model, where single layer disruptions are easily compensated for by remaining layers.

2. **Critical Impact with Larger Gaps:** As the number of ablated layers increases (gaps - 4, 5, 6, etc.), the number of common predictions decreases more noticeably. I decided to look for the common predictions less than 75. As the gap increases the common predictions fall below 75. Say, when 6 layers are ablated together (6-11), the common predictions drop to 31. This suggests that these specific layers are more critical to model's predictions capability. While common predictions are between 30 and 61 for gaps 4, 5, 6, 7 and 8, but the least common predictions are 31 for gap - 6, hence I decided to look into layers 6-11.

3. **Layer Insensitivity Beyond Certain Depths:** The table also reveals that ablating layers beyond a certain depth (layers greater than 34) does not significantly impact the model's predictions. Even when multiple layers are ablated simultaneously, the model still retains a certain number of common predictions. This observation suggests that deeper layers (those beyond layer 34) have a less critical role in the specific task or dataset being analyzed, or they contribute to more specialized, less generalizable functions within the model.

	78	**79**	**80**		
Gap - 1	78	79	80		
Layers-ablated	11, 12	0, 17, 5, 4, 16			
Gap - 2	77	79	76		
Layers-ablated	16-17, 12-13	18-19, 20-21	4-5, 6-7, 10-11		
Gap - 3	75	77	78	79	
Layers-ablated	12-14	9-11, 6-8	15-17	27-29, 18-20, 3-5, 24-26	
Gap - 4	61	70	77	78	79
Layers-ablated	8-11	4-7	24-27, 16-19	20-23, 12-15	0-3
Gap - 5	48	68	78	79	
Layers-ablated	5-9	10-14	20-24, 25-29	15-19, 0-4, 30-34	
Gap - 6	31	76	77	79	
Layers-ablated	6-11	24-29	12-17	18-29, 0-5, 30-35	
Gap - 7	38	77	76	78	79
Layers-ablated	7-13	14-20	28-34	21-27	0-6
Gap - 8	34	76	79		
Layers-ablated	8-15	24-31	16-23, 0-7		

Table 2: Results of Mean Ablation Study on Model Predictions

6 Analysis of Attention Rollout in layers 6 to 11

After identifying that layers 6 to 11 play a significant role in the singular-to-plural conversion task, I performed an in-depth analysis using **attention rollout**. This helps understand how information flows across multiple layers and which tokens the model focuses on during processing.

Attention rollout is a method used to visualize the cummulative attention between tokens, particularly in transformer models. It aggregates attention scores across multiple layers, providing a more holistic view of how attention is distributed throughout the network. This can reveal both the direct and indirect pathways through which information is propagated across layers. **Methodology**

1. **Store Attention Maps:** For layers 6 to 11, I extracted the attention maps of every attention head. Each attention map is a matrix where each element represents the attention weight from one token to another in the input sequence.

2. **Initialize the Rollout matrix:** Start with an identity matrix of size equal to the number of tokens. This matrix represents that, initially, each token only attends to itself with full attention.

3. **Aggregate Attention Scores:** For each layer (from 6 to 11), compute the mean attention matrix across all heads. Multiply these attention matrices layer by layer, aggregating the attention across all the specified layers. This multiplication step accumulates the direct and indirect attention paths, capturing both immediate and propagated influences between tokens.

4. **Compute Cummulative Attention:** The final result is a cumulative attention matrix that reflects how much attention each token pays to every other token across the entire range of layers 6 to 11.

5. **Plot the Attention Rollout:** Visualize the cumulative attention matrix using a heatmap. In this plot, the x-axis and y-axis represent the tokens in the input sequence, and each cell shows the aggregated attention weight from one token to another across layers 6 to 11.

Key Observations from the Attention Rollout Plot:

1. **Diagonal Dominance:** The Figure 1 reveals a strong diagonal pattern where the attention values are significantly higher along the diagonal (i.e., cells where the x and y coordinates are the same). This indicates that the model consistently focuses on each token itself rather than shifting its focus to other tokens. Layers 6 to 11 are primarily preserving individual token information, suggesting that these layers are maintaining the context or identity of each token rather than facilitating substantial interactions between different tokens.

2. **Off-Diagonal Attention:** The cells off the diagonal generally exhibit lower attention values, showing that there is less cross-token attention. Some off-diagonal cells have moderate attention values, indicating that there are some interactions between adjacent or closely related tokens (e.g., from token 1 to token 2, from token 2 to token 3, from token 3 to token 4, i.e., following tokens.). These interactions suggest that while the model primarily focuses on individual tokens, there is some degree of dependency or influence between adjacent tokens.

3. **Focused Attention on Specific Token Pairs:** Certain areas in the plot show focused attention on specific token pairs, such as token 2 focusing on token 3 or vice versa. The code for this is in **attention_rollout.ipynb** notebook.

7 Further Analysis: Averaging and Thresholding of Attention Value

Building upon the insights gained from the attention rollout, the next step involves a more granular analysis of the attention patterns for specific tokens that were highlighted along the diagonal in the cumulative attention plot. This step aims to quantify the average attention distribution for each token across all data points and subsequently investigate the impact of attention heads and layers through a targeted ablation study.

Averaging Attention Values for Diagonal Tokens

1. **Identification of Diagonal Tokens:** From the attention rollout plot, certain tokens (1,2,3,4,5) along the diagonal were identified as receiving consistently high attention across layers 6 to 11. These diagonal tokens suggest that the model is focusing on these tokens individually, preserving their information through multiple layers.

2. **Computing Average Attention Values:** To understand the overall attention dynamics, the next step is to compute the average attention values for each of these diagonal tokens. This average is computed over all data points in the dataset, allowing us to capture a generalizable attention pattern for each token. For each token, the attention values from all attention heads across the relevant layers (6 to 11) are averaged. The result of this computation is an array of dimensions 25 (heads) by 6 (layers), where each entry represents the mean attention value of a specific head and layer for a given token.

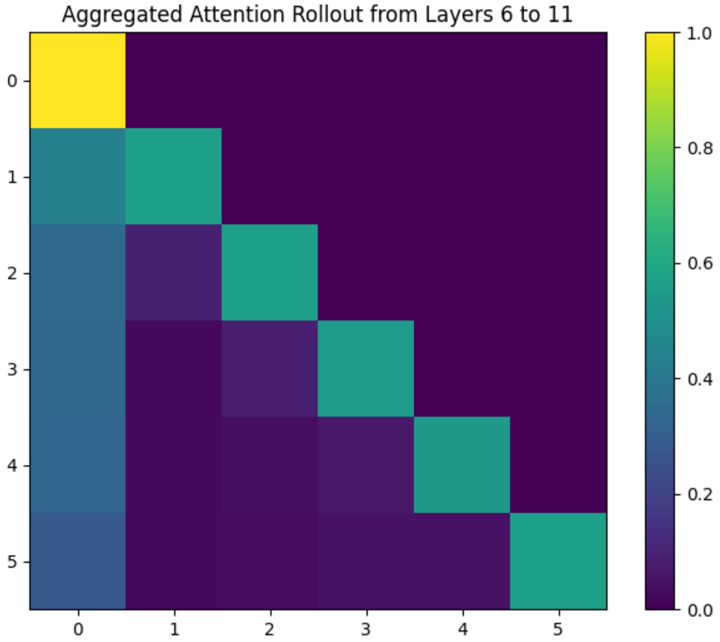


Figure 1: Attention Rollout Plot for Layers 6 to 11

3. **Visualization of Averaged Attention:** The resulting averages are visualized in a series of plots, one for each token (e.g., tokens 1, 2, 3, 4, and 5), as shown in Figure **??**. These plots illustrate the average attention distributions across the 25 attention heads and 6 layers, providing a detailed view of which heads and layers are most active or attentive for each token.

Thresholding Attention Values into Windows

1. **Defining Attention Threshold Windows:** To further analyze the attention distribution, each token's average attention values are segmented into five distinct threshold windows:

   (a) Window 1: 0.0 - 0.2

   (b) Window 2: 0.2 - 0.4

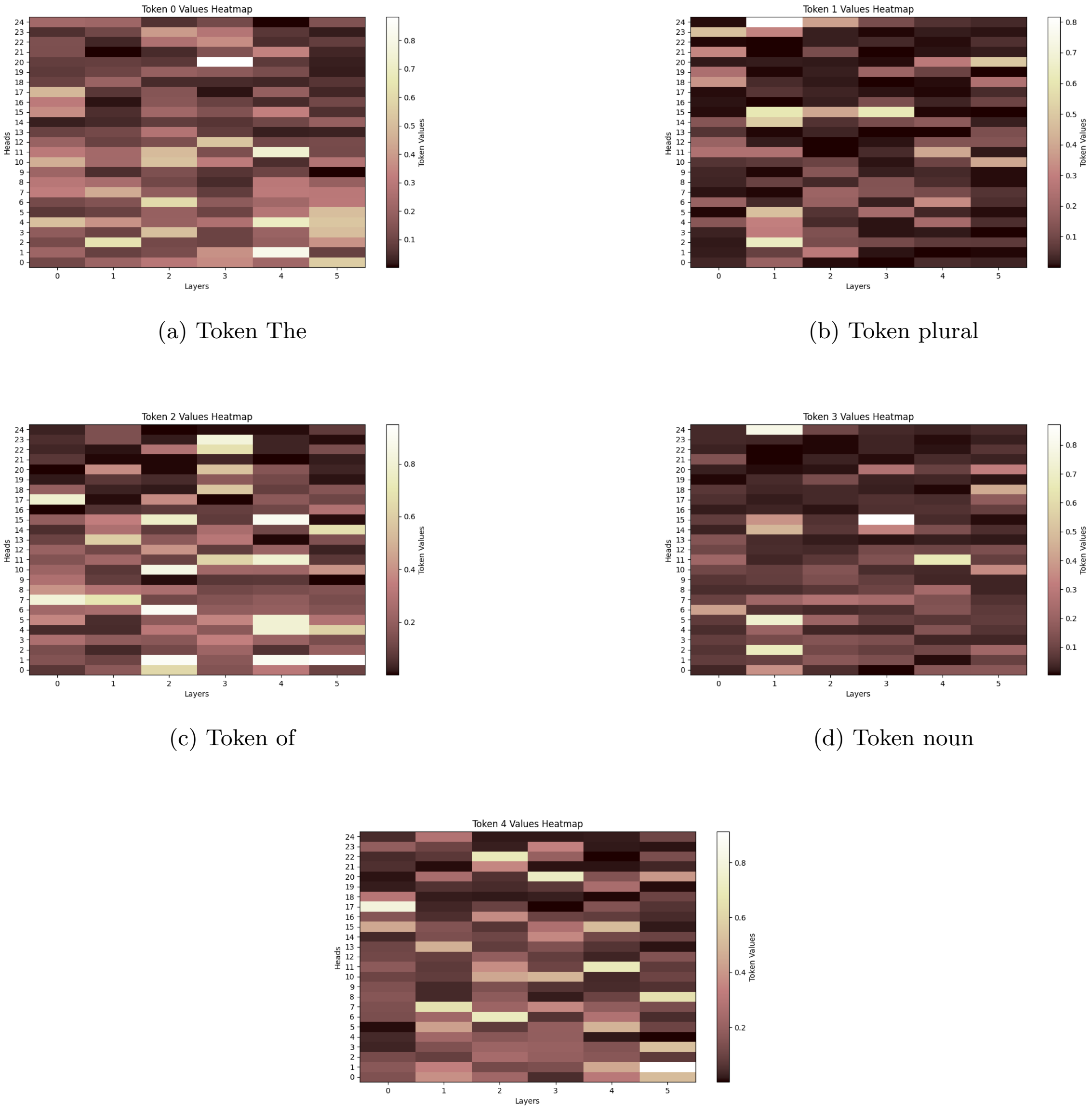
   (c) Window 3: 0.4 - 0.6

   (d) Window 4: 0.6 - 0.8

   (e) Window 5: 0.8 - 1.0

   These thresholds are chosen to categorize the attention values into varying levels of intensity, from low (near-zero attention) to high (near-total attention). This categorization helps identify how different attention heads and layers contribute to the model's focus on specific tokens.

2. **Head-Layer Ablation for Each Window Setting:** For each token and each defined window, I performed a head-layer ablation study. This involves selectively ablating (removing or disabling) attention heads and under corresponding layers that fall within each threshold range to observe how these ablations affect the model's performance and predictions. The purpose of this targeted ablation is to determine the importance of different heads and layers depending on their attention intensity levels. By examining the impact of ablating heads and layers in each threshold window, it is possible to identify which attention components are crucial for preserving key token information and which components can be removed without significantly affecting the model's output. The token wise and layer wise heads are shown in the 3



(a) Token The

(b) Token plural

(c) Token of

(d) Token noun

(e) Token is

Figure 2: Averaged Attention Values for Tokens 1, 2, 3, 4, and 5 Across Layers 6 to 11

3. **Output:** For each token, heads extracted from layers 6-11 were ablated collectively. It was observed that as the window setting progressed from Window 5 (0.8 - 1.0) to Window 1 (0.0 - 0.2), the number of common predictions decreased significantly, **from 31 in Window 5 to just 10 in Window 1**. This decline highlights the critical role of these heads in the attention mechanism within the specified layers.

4. **Generalizability Check:** To assess the generalizability of the identified circuit, additional experiments were conducted on plurals with specific endings, including those ending in 'es', 'same', and vowels. Ablation of layers 6-11 for these token categories resulted in a substantial reduction in prediction accuracy, reinforcing the importance of the ablated heads across different plural forms. The detailed results of these experiments are summarized in the 4.

	Layer 6	**Layer 7**	**Layer 8**	**Layer 9**	**Layer 10**	**Layer 11**
Token 1 (The)	0, 2, 3, 5, 6, 12, 13, 14, 18, 19, 20, 21, 22, 23	1, 3, 5, 6, 9, 12, 13, 14, 15, 16, 17, 19, 20, 21, 22, 23	1, 2, 4, 5, 7, 8, 9, 12, 14, 16, 17, 18, 19, 20, 21, 24	1, 8, 9, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24	2, 3, 5, 6, 7, 8, 9, 11, 13, 14, 15, 16, 17, 18, 19, 21, 24	2, 9, 10, 12, 13, 14, 16, 17, 18, 19, 20, 22, 23, 24
Token 2 (plural)	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 13, 14, 15, 17, 20, 22, 24	0, 1, 6, 7, 8, 9, 10, 12, 13, 16, 17, 18, 19, 20, 21	0, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16, 17, 18, 19, 20, 21, 22, 23	0, 1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16, 17, 18, 19, 20, 21, 22, 23, 24	0, 1, 2, 3, 5, 7, 8, 9, 10, 12, 13, 14, 15, 16, 17, 18, 19, 21, 22, 23, 24	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 12, 13, 14, 15, 16, 17, 19, 21, 22, 23, 24
Token 3 (of)	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 13, 14, 15, 16, 17, 20, 22, 24	0, 1, 6, 7, 8, 9, 10, 12, 13, 16, 17, 18, 19, 21, 22	0, 2, 3, 4, 5, 6, 9, 10, 11, 12, 13, 14, 16, 17, 18, 19, 20, 21, 22, 23	0, 1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16, 17, 18, 19, 20, 21, 22, 23, 24	0, 1, 2, 3, 5, 7, 8, 9, 10, 12, 13, 14, 15, 16, 17, 18, 19, 21, 22, 23, 24	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 12, 13, 14, 15, 16, 17, 19, 21, 22, 23, 24
Token 4 (noun)	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 12, 13, 14, 15, 16, 17, 19, 21, 22, 23, 24	1, 3, 6, 8, 9, 10, 11, 12, 13, 16, 17, 18, 19, 20, 21, 22, 23	0, 1, 2, 3, 4, 6, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24	0, 1, 2, 3, 4, 5, 6, 8, 9, 10, 11, 12, 13, 16, 17, 18, 19, 21, 22, 23, 24	0, 1, 2, 3, 4, 5, 6, 7, 9, 10, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24	0, 1, 3, 4, 5, 6, 7, 8, 9, 11, 12, 13, 14, 15, 16, 17, 19, 21, 22, 23, 24
Token 5 (is)	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16, 19, 20, 21, 22, 23, 24	2, 3, 8, 9, 10, 11, 12, 14, 15, 16, 17, 18, 19, 21, 22, 23	1, 4, 5, 8, 9, 12, 13, 14, 15, 17, 18, 19, 20, 23, 24	0, 1, 2, 4, 5, 6, 8, 9, 11, 12, 13, 16, 17, 18, 19, 21, 22, 24	2, 3, 4, 7, 8, 9, 10, 12, 13, 14, 16, 17, 18, 20, 21, 22, 23, 24	2, 4, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 21, 22, 23, 24

Table 3: Important attention heads for each token across layers 6 to 11

	es	**vowel**	**Same**
Ablation- 6-11 layers	10	5	19
No ablation	19	10	22

Table 4: Comparison of different ablation layers and conditions.

8 Removing Layer Normalization in Layers 6 to 11

Layer normalization normalizes the input across each layer, stabilizing the learning process and improving convergence during training. By maintaining a normalized range of values, layer normalization helps the model handle diverse input data more effectively and ensures that the gradient descent optimization process remains stable.

The purpose of this experiment was to assess the importance of layer normalization in layers 6 to 11, which were previously identified as being crucial for maintaining token-level information during the singular-to-plural conversion task. By disabling layer normalization in these specific layers, I aimed to determine its role in preserving the model's prediction accuracy and robustness.

1. **Model Modification:** modified the architecture of the GPT-2 XL model by removing the layer normalization step specifically in layers 6 to 11. All other aspects of the model, including the attention mechanisms and feed-forward networks, were left unchanged.

2. **Evaluation:** The modified model was then evaluated on the same dataset used for previous experiments, focusing on its ability to correctly predict the plural forms of singular nouns. The metric for evaluation remained consistent with previous experiments, where the number of correct predictions out of the total possible predictions was recorded.

Upon removing layer normalization from layers 6 to 11, the model's prediction accuracy dropped dramatically to 40. In other words, the model failed to make any correct predictions for the plural forms of singular nouns when layer normalization was disabled in these layers.

The experiment demonstrates that layer normalization is not merely a peripheral component but is central to the model's ability to maintain high performance in tasks requiring fine-grained token-level manipulations, such as singular-to-plural transformations. Removing layer normalization from layers 6 to 11 causes a total collapse in the model's predictive capability, underlining its critical role in preserving the stability and effectiveness of neural network operations within these layers.

9 Analysis of Sentence Templates

9.1 Standard Template: "The plural of noun is"

1. **prepend_bos** = True: This is my standard template setup. The model's accuracy with this configuration serves as a baseline for comparison, i.e., 80/127 correct predictions on plural ending with

’s’.

2. **prepend_bos** = False: Without the BOS (beginning of sentence) token, the model produces 52 correct predictions. The presence of the BOS token seems to slightly aid in accuracy.

9.2 Variation 1: Removal of ”The”

prepend_bos as True:

1. Outcome: 0 correct predictions. The majority of predictions are ”a”.
2. Interpretation: The removal of ”The” likely disrupts the model’s understanding of the sentence structure, leading it to default to simpler, more frequent words like ”a.” This suggests that ”The” plays a crucial role in guiding the model toward understanding the sentence as a specific structure requiring a plural form.

prepend_bos as False:

1. Outcome: Only 1 correct prediction, with the majority of predictions being repetitions of the singular noun.
2. Interpretation: The absence of both ”The” and the BOS token severely hinders the model’s ability to predict the correct plural form. The model seems to rely more on the structural cues provided by ”The” when the BOS token is not used.

9.3 Variation 2: Removal of ”of” with prepend_bos = True

1. Outcome: 0 correct predictions. Predictions are scattered, including ”a,” ”used,” and ”the.”
2. Interpretation: The word ”of” is essential for connecting ”The plural” with the noun. Its removal disrupts this connection, causing the model to generate unrelated words. This indicates that ”of” is a key component in the model’s understanding of the phrase.

9.4 Variation 2: Removal of ”a” with prepend_bos = True

1. Outcome: 0 correct predictions, with all predictions being ”is.”
2. Interpretation: This suggests that ”a” is crucial for the model to understand the structure leading to the noun. Without ”a,” the model defaults to ”is,” indicating it fails to recognize the need for a noun and instead completes the sentence with what it expects to be the end of the phrase.

9.5 Alternative Template: ”The plural form of noun is”

prepend_bos as True:

1. Outcome: 69 correct predictions.
2. Interpretation: Adding the word ”form” appears to work in a descent way but not leading to a higher number of correct predictions. This suggests that the model is not adding any better context to the intended meaning of the phrase with this explicit clarification.

prepend_bos as False:

1. Outcome: 50 correct predictions, a decrease compared to when prepend_bos is True.
2. Interpretation: The absence of the BOS token again reduces accuracy.

9.6 Alternative Template: "The plural version of noun is"

prepend_bos as True:

1. Outcome: 32 correct predictions.
2. Interpretation: The word "version" seems to confuse the model slightly more than "form," leading to fewer correct predictions. The model might be interpreting "version" less clearly as a cue for pluralization.

prepend_bos as False:

1. Outcome: 6 correct predictions, a significant drop.
2. Interpretation: Without the BOS token, the model struggles even more with this template, showing the critical role of sentence structure and preparatory context in generating accurate predictions.

These experiments demonstrate the importance of both sentence structure and the prepend_bos parameter in guiding the model's predictions. The standard template performs moderately well, but variations, particularly those that remove key structural words, significantly impact the model's ability to generate correct plural forms. The inclusion of words like "form" and "version" introduce some ambiguity. The prepend_bos token generally enhances accuracy, likely by providing a clearer starting context for the model. The code for this experiment is in **data_different_version_singular_plural.ipynb** notebook.

10 Future Work

1. **Analysis of Common Predictions:** Investigate the similarities among words where predictions remain consistent. Understanding commonalities in words where predictions are correct could provide insights into the underlying mechanisms of the model.
2. **Generalizability to Other Plural Forms:** Explore the performance of the identified heads on other plural forms, such as those ending in 'es', 'same', and vowels, to validate the generalizability of the results.
3. **Deeper Circuit Analysis:** Conduct a more in-depth analysis of the identified attention circuits. This includes examining their roles and interactions within the model to better understand their contributions to predictions.
4. **Impact of Increased Predictions Post-Ablation:** Examine instances where the number of total predictions increased following ablation. Understanding why and how ablation affects prediction count can offer insights into model behavior and attention mechanisms.
5. **Information Flow Analysis:** Investigate how information flows through the attention layers and heads. This includes analyzing how ablation affects information transmission and overall model performance.
6. **Importance of Other Layers:** Extend the analysis to include layers beyond 6-11. Assessing the importance of attention heads in other layers could provide a more comprehensive understanding of their role in the model's performance.

11 Conclusion

This study provides valuable insights into the behavior of the GPT-2 XL model in handling singular-to-plural transformations across different categories. The findings indicate that certain layers, particularly layers 6 to 11, play a critical role in preserving token-level information and maintaining prediction accuracy. The use of mean ablation and attention rollout techniques has revealed the importance of specific attention heads and layer normalizations in processing different types of pluralization rules. Furthermore, the experiments underscore the significance of sentence structure and the inclusion of specific words in guiding the model towards accurate predictions. Future work will focus on extending these findings to other forms of pluralization and conducting deeper analyses to better understand the model's internal circuits and their contributions to natural language processing tasks.