# Derivational Probing: Unveiling the Layer-wise Derivation of Syntactic Structures in Neural Language Models

**Taiga Someya**[♠] **Ryo Yoshida**[♠] **Hitomi Yanaka**[♠♡] **Yohei Oseki**[♠]
[♠]The University of Tokyo [♡]RIKEN
{taiga98-0809,yoshiryo0617,hyanaka,oseki}@g.ecc.u-tokyo.ac.jp

## Abstract

Recent work has demonstrated that neural language models encode syntactic structures in their internal *representations*, yet the *derivations* by which these structures are constructed across layers remain poorly understood. In this paper, we propose *Derivational Probing* to investigate how micro-syntactic structures (e.g., subject noun phrases) and macro-syntactic structures (e.g., the relationship between the root verbs and their direct dependents) are constructed as word embeddings propagate upward across layers. Our experiments on BERT reveal a clear bottom-up derivation: micro-syntactic structures emerge in lower layers and are gradually integrated into a coherent macro-syntactic structure in higher layers. Furthermore, a targeted evaluation on subject-verb number agreement shows that the timing of constructing macro-syntactic structures is critical for downstream performance, suggesting an optimal timing for integrating global syntactic information.

https://github.com/osekilab/derivational-probing

## 1 Introduction

Neural language models have achieved remarkable success across a wide range of natural language processing tasks. However, significant uncertainty remains regarding what these models truly learn and how they represent linguistic knowledge. This has spurred extensive research aimed at probing the linguistic capabilities of neural language models (Zhao et al., 2024; Chang and Bergen, 2024).

A prominent line of inquiry is *structural probing*, which directly analyzes word embeddings to uncover latent syntactic structures. For example, Hewitt and Manning (2019) demonstrated that the geometric organization of the word embedding space in BERT (Devlin et al., 2019) encodes syntactic distances defined over dependency parse trees, providing evidence that the model captures syntactic information. Yet, such work typically focuses on the static representations of the whole syntactic structures rather than the dynamic derivations by which these syntactic structures are built across layers. Understanding not just the resulting representations but also how they are built across layers is essential for a more comprehensive understanding and could also lead to better insights into how these representations are used.

Meanwhile, Tenney et al. (2019) introduced the *expected layer* metric and investigated how different layers in BERT encode different types of linguistic information (e.g., part-of-speech tagging, syntactic parsing, semantic role labeling, and coreference resolution), revealing that the model encodes linguistic abstractions in a manner reflecting a traditional NLP pipeline. However, their approach primarily relied on coarse-grained task accuracy measures, capturing only the overall effectiveness of each layer rather than examining the detailed, layer-wise construction of specific syntactic structures. Consequently, how the syntactic structures are built across layers remains underexplored.

In this paper, we fill this gap by proposing *Derivational Probing*—a method that integrates structural probing with the expected layer metric to probe derivation processes of syntactic structures in neural language models (Figure 1). Our proposed method allows us to investigate how *micro*-syntactic structures (e.g., subject and object noun phrases, prepositional phrases) and *macro*-syntactic structures (e.g., the relationship between the root verbs and their direct dependents) are constructed across layers.

Applying Derivational Probing to BERT (Devlin et al., 2019), our experiments reveal a clear bottom-up derivation, in which micro-syntactic structures emerge in lower layers and are gradually integrated into a coherent macro-syntactic structure in higher layers. Furthermore, our targeted analysis on a

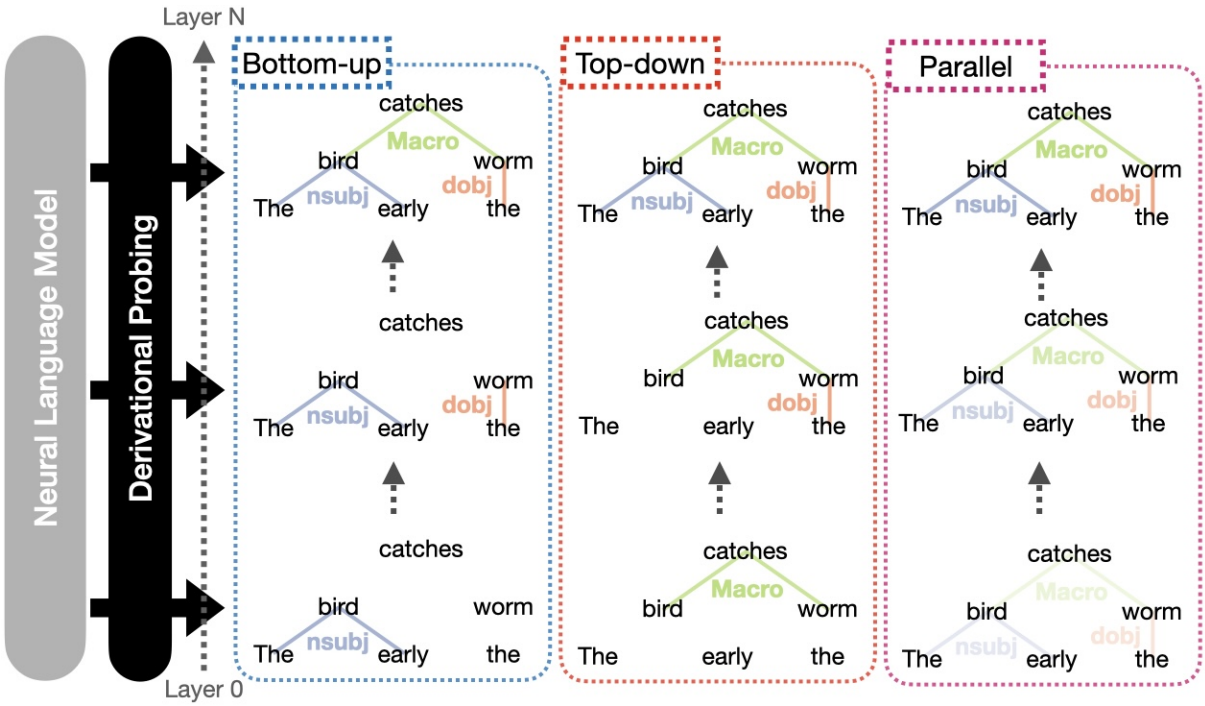


Figure 1: Derivational Probing investigates how syntactic structures are constructed across layers in neural language models. We illustrate three hypotheses for syntactic derivation. **Bottom-up:** Micro-syntactic structures, such as subject noun phrases or prepositional phrases, emerge in lower layers, and the macro-syntactic structure is formed in higher layers. **Top-down:** The macro-syntactic structure is formed in lower layers, with micro-syntactic structures refined in higher layers. **Parallel:** Micro- and macro-syntactic structures emerge in parallel.

subject-verb number agreement task shows that even when the final syntactic structure is correct, the specific layers at which the macro-syntactic structure is constructed significantly affect downstream performance. This suggests the existence of an optimal timing for integrating global syntactic information.

Overall, our findings offer new insights into the internal mechanisms by which neural language models construct syntactic structures and underscore the importance of examining derivation processes across layers to improve the interpretability of neural language models.

## 2 Related Work

Attention-based analyses (e.g., Clark et al., 2019; Vig and Belinkov, 2019) have demonstrated that certain transformer heads tend to align with dependency relations, providing evidence that Transformer language models capture linguistic dependency relations in their attention weights.

In contrast, Hewitt and Manning (2019) introduced a structural probe with a linear transformation from hidden representations into a space where Euclidean distances reflect dependency tree distances. This approach revealed that full syntactic trees are implicitly encoded in models such as BERT. Building on this, later work refined the approach by incorporating non-linear mappings (e.g., White et al., 2021), enforcing constraints such as orthogonality (Limisiewicz and Mareček, 2021), and using a controlled corpus to isolate the effect of syntax (Maudslay and Cotterell, 2021).

Other studies have refined structural probing by quantifying context-dependent syntactic signals in deeper layers—for example, conditional probing (Hewitt et al., 2021) and information gain metrics (Kunz and Kuhlmann, 2022)—but these methods focus on the performance of specific probing tasks (e.g., POS-tagging) rather than where the syntactic structures are constructed.

In contrast, our proposed method specifically tracks how each subgraph in the syntactic tree develops as information propagates through the network layers. By analyzing the evolution of individual syntactic components—from micro-syntactic structures to the assembly of the macro-syntactic structure—we offer a more granular perspective on the incremental construction of syntax, complementing and extending previous layer-wise analyses.

## 3 Technical Preliminaries

In this section, we review foundational methods from prior research: structural probing to assess the presence and quality of syntactic representations and the expected layer metric for quantifying how linguistic information gradually builds up across successive layers within language models.

### 3.1 Structural Probing

Hewitt and Manning (2019) introduced the *structural probe* as a method to evaluate whether contextual word representations encode syntactic information. Given a sentence $\mathbf{s} = w_1 \cdots w_t$, each token is represented by a $d$-dimentional contextual embedding $\mathbf{h}_i \in \mathbb{R}^d$ (e.g., the output embedding of a model like BERT). The goal of the structural probe is to find a linear transformation that maps these embeddings to a space where the Euclidean distances approximate the true syntactic distances between words.

Specifically, for any two words $w_i$ and $w_j$ in a sentence, we define the transformed distance as:

$$d_{\mathbf{B}}(\mathbf{h}_i, \mathbf{h}_j) = \|\mathbf{B}\mathbf{h}_i - \mathbf{B}\mathbf{h}_j\|_2, \tag{1}$$

where $\mathbf{B} \in \mathbb{R}^{d' \times d}$ is a learnable projection matrix. The true syntactic distance, $\Delta_{ij}$, is typically defined as the number of edges on the shortest path between $w_i$ and $w_j$ in the dependency parse tree of the sentence. The probe is trained by minimizing an objective that penalizes the discrepancy between the predicted distances and $\Delta_{ij}$:

$$\mathcal{L} = \frac{1}{|\mathbf{s}|^2} \sum_{i=1}^{|\mathbf{s}|} \sum_{j=i+1}^{|\mathbf{s}|} \left| \Delta_{ij} - d_{\mathbf{B}}(\mathbf{h}_i, \mathbf{h}_j) \right|. \tag{2}$$

This formulation encourages the linear transformation $\mathbf{B}$ to capture the syntactic structure encoded in the contextual representations, enabling the recovery of parse trees via Prim's (1957) algorithm, a greedy algorithm that constructs minimum spanning trees by iteratively adding the lowest-weight edge connecting a new node to the growing tree.

### 3.2 Expected Layer

The *expected layer* metric introduced by Tenney et al. (2019) was initially developed to identify the layers within BERT responsible for solving various linguistic tasks. Specifically, the metric was used to capture at which layers broad linguistic abilities (e.g., part-of-speech tagging, syntactic parsing, semantic role labeling) emerge. Tenney et al. defined *scaler-mixed embeddings* $\boldsymbol{m}_i^\ell \in \mathbb{R}^d$ as the weighted average of embeddings from the bottom layer up to layer $\ell$:

$$\boldsymbol{m}_i^\ell = \gamma \sum_{k=0}^{\ell} \mathbf{w}_k \, \mathbf{h}_i^k, \tag{3}$$

where $\mathbf{w} = \mathrm{softmax}(\mathbf{a})$ ($\mathbf{a} \in \mathbb{R}^{\ell+1}$) is learnable scalar mixing weights and $\gamma$ is a learnable scaling factor, following Peters et al. (2018).

By measuring performance at layer index $\ell$, denoted by $S(\ell)$, and tracking its improvements across layers, Tenney et al. defined the expected layer to reflect the layer at which the relevant linguistic task information is predominantly captured:

$$E[\ell] = \sum_{\ell=1}^{L} \frac{S(\ell) - S(\ell-1)}{\sum_{\ell=1}^{L}(S(\ell) - S(\ell-1))} \ell. \tag{4}$$

This is the weighted average of layer indices, where each layer's weight corresponds to its relative contribution to the overall performance improvement. It was initially proposed to broadly characterize the hierarchical progression of different linguistic capabilities within transformer models, rather than pinpointing the exact layers at which specific syntactic structures are built.

## 4 Derivational Probing

Building upon these prior techniques, we propose *Derivational Probing*, a novel method explicitly designed to investigate the dynamic construction of syntactic structures across the layers of neural language models.

Our approach effectively combines expected layer metric (Tenney et al., 2019) with the structural probing (Hewitt and Manning, 2019), enabling a detailed analysis of how syntactic information accumulates across model layers. Specifically, for each layer $\ell$, we use scalar-mixed embeddings as defined in Eq. (3) and compute pairwise distances:

$$d_{\mathbf{B}_\ell}(\boldsymbol{m}_i^\ell, \boldsymbol{m}_j^\ell) = \|\mathbf{B}_\ell \boldsymbol{m}_i^\ell - \mathbf{B}_\ell \boldsymbol{m}_j^\ell\|_2. \tag{5}$$

We then train the transformation matrix $\mathbf{B_k}$ to minimize discrepancies with true dependency parse distances, analogous to structural probing.

This integration allows us to calculate the expected layer for each syntactic subgraph (micro- and macro-syntactic structures defined in detail later) and perform a fine-grained, quantitative analysis of their construction across model layers. We

use the Unlabeled Undirected Attachment Score (UUAS) for each layer $\ell$ as $S(\ell)$, defined as the proportion of correctly predicted edges to the total number of edges in the reference dependency parse, without considering edge labels or direction.

To better understand the derivation strategy that models employ when constructing a syntactic tree, we introduce a distinction between *macro-syntactic structures* (the root verb with its direct dependents) and *micro-syntactic structures* (local components, such as subordinate phrases like `nsubj`) (Figure 2). This distinction is motivated by our interest in whether models construct syntactic trees top-down, bottom-up, or in a parallel fashion. To empirically evaluate which of these hypotheses is most plausible, we adopt the following methodological approach: For both micro-syntactic structures and macro-syntactic structures, we (1) construct the full parse tree using a minimum spanning tree algorithm, (2) extract the relevant edges (as highlighted in Figure 2), and (3) compute the UUAS by comparing these edges to the reference parse. By tracking UUAS improvements across layers, we calculate the expected layer $E[l]$ for each structure, revealing the layers at which different syntactic subgraphs are effectively constructed.

We next provide detailed descriptions of each hypothesis.

**Bottom-up derivation.** A bottom-up derivation first constructs micro-syntactic structures and subsequently integrates these into macro-syntactic structures, ultimately forming a complete dependency tree. We refer to this as a "bottom-up derivation" because it resembles the construction order of the arc-standard transition-based dependency parser (Nivre, 2004). Arc-standard parsing utilizes a stack-based transition system and constructs a dependency tree in a bottom-up manner: dependents must be fully processed and attached to their heads before those heads themselves are incorporated into macro-syntactic structures. Under this hypothesis, models initially identify micro-syntactic structures—such as the internal phrase structures of subjects and objects—in lower layers, which are then progressively combined into a coherent macro-syntactic hierarchy at higher layers.

**Top-down derivation.** A top-down derivation, in contrast, begins by establishing macro-syntactic structures and subsequently refines these by incorporating detailed micro-syntactic dependencies. We term this approach a "top-down derivation" because its construction order aligns closely with the head-driven transition-based parser proposed by Hayashi et al. (2012). Their algorithm explicitly predicts dependent nodes from head nodes, progressively building syntactic structures from head to dependent, thus genuinely following a top-down, predictive parsing order. Under this hypothesis, models prioritize the recognition of macro-syntactic structures before refining micro-syntactic struvtures.

**Parallel derivation.** Finally, an alternative hypothesis is that models construct micro- and macro-syntactic structures concurrently, with local dependencies and the global structure forming at roughly the same rate across layers. This hypothesis is less clearly aligned with traditional dependency parsing algorithms, as most classical approaches tend to favor either bottom-up or top-down derivations.

**Notes on the term "derivation".** Here, we explicitly use the term *derivation* (strategy) throughout this paper rather than "parsing strategy" to clearly distinguish two related but distinct concepts. While "parsing strategy" generally refers to methodological choices for *incrementally* constructing a parse tree (such as bottom-up or top-down), our use of "derivation" specifically captures an *atemporal* process describing how syntactic structures progressively emerge across the internal layers of a language model given the full sentence context, emphasizing *layer-wise* structural development rather than sequential, *left-to-right* incremental processing.

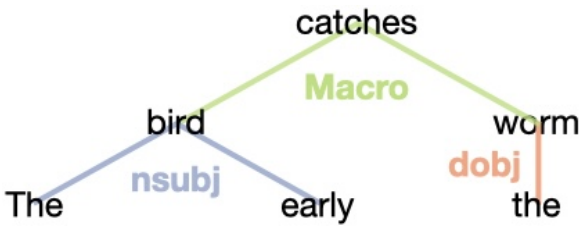


Figure 2: *Macro-syntactic structure* (`Marco`) and *micro-syntactic structures* (`nsubj` and `dobj`).

## 5 Experimental Setup

### 5.1 Data

We utilize the Wikitext-103 dataset (Merity et al., 2016) as our primary source of natural language, parsing each sentence with spaCy's dependency parser (EN_CORE_WEB_LG) (Honnibal et al., 2020). To focus on the language model's ability to

| Structure Set | Example Sentence |
|---|---|
| Marco, nsubj, dobj | The concert caused a major stir. |
| Marco, nsubj, prep | The match ended in a goalless draw. |
| Marco, nsubj, attr | Her parents were music professors. |
| Marco, nsubj, prep, dobj | The film received positive reviews from critics. |

Table 1: Example sentences for each primary structure set described in §5.1

construct syntactic structures in a clear-cut setting, we restrict our analysis to single-clause sentences by excluding those with relative clauses or clausal subjects. Additionally, we filter out sentences containing dependency relations such as "dep" (unclassified dependents) and punctuation marks other than sentence-final punctuation to minimize noise.

Following the definitions introduced in the previous section (§4), we group sentences based on dependency relations emanating from the root verb, thereby distinguishing between the overall (macro-syntactic; Marco) structure and subordinate (micro-syntactic) structures. We retain only those groups that represent more than 10% of the data, focusing our analysis on the predominant structure sets. This filtering results in four primary structure sets (See Table 1 for examples): (1) Marco with micro relations nsubj and dobj; (2) Marco with micro relations nsubj and prep; (3) Marco with micro relations nsubj and attr; and (4) Marco with micro relations nsubj, prep, and dobj.

From the resulting dataset, we randomly sample 50,000 sentences, partitioning them into 40,000 for training, 5,000 for validation, and 5,000 for testing.

### 5.2 Models

We employ two pre-trained language models: BERT-base[1] and BERT-large[2] (cased) (Devlin et al., 2019). BERT-base uses 12 layers, 12 heads, and a 768-dimensional hidden state, while BERT-large uses 24 layers, 16 heads, and a 1024-dimensional hidden state. These models provide a range of capacities, allowing us to investigate differences in how syntactic structures are constructed across models.

We focus specifically on BERT because our method is designed to examine the *atemporal*, layer-wise derivation of syntactic structures given entire sentences. In contrast, autoregressive language models such as GPT-2 process information incrementally in a left-to-right manner, and the *temporal*, token-wise derivation of syntactic structures cannot be probed via our method (cf. Eisape et al., 2022). Nevertheless, our method is still applicable to the word embeddings of autoregressive language models such as GPT-2, and we report GPT-2 results in App. A.

For each model, we probe all layers to determine the progression of syntactic information and compute the expected layer at which specific structures emerge. We conduct training with five different random seeds and report the average performance along with the standard deviation. Additional hyperparameters and training details are provided in App. B.

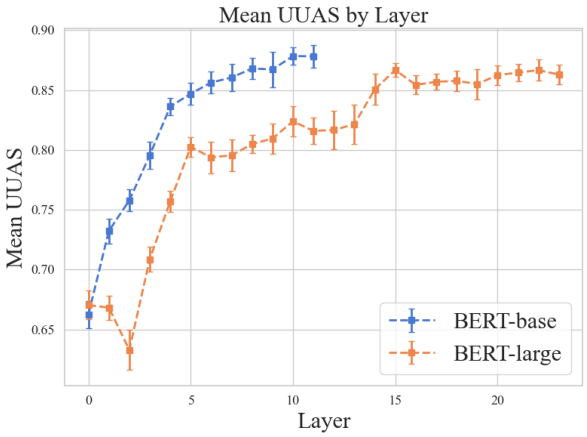


Figure 3: Global UUAS by each layer for each model. Error bars represent standard deviations across 5 random seeds.

## 6 Results

### 6.1 Overall UUAS Performance

As a sanity check to verify whether our models exhibit overall trends similar to those reported in previous studies, we conducted an experiment measuring the test set UUAS for overall sentence structures across layers for each model (Figure 3). BERT-base and BERT-large display similar trends,

[1] https://huggingface.co/google-bert/bert-base-cased

[2] https://huggingface.co/google-bert/bert-large-cased

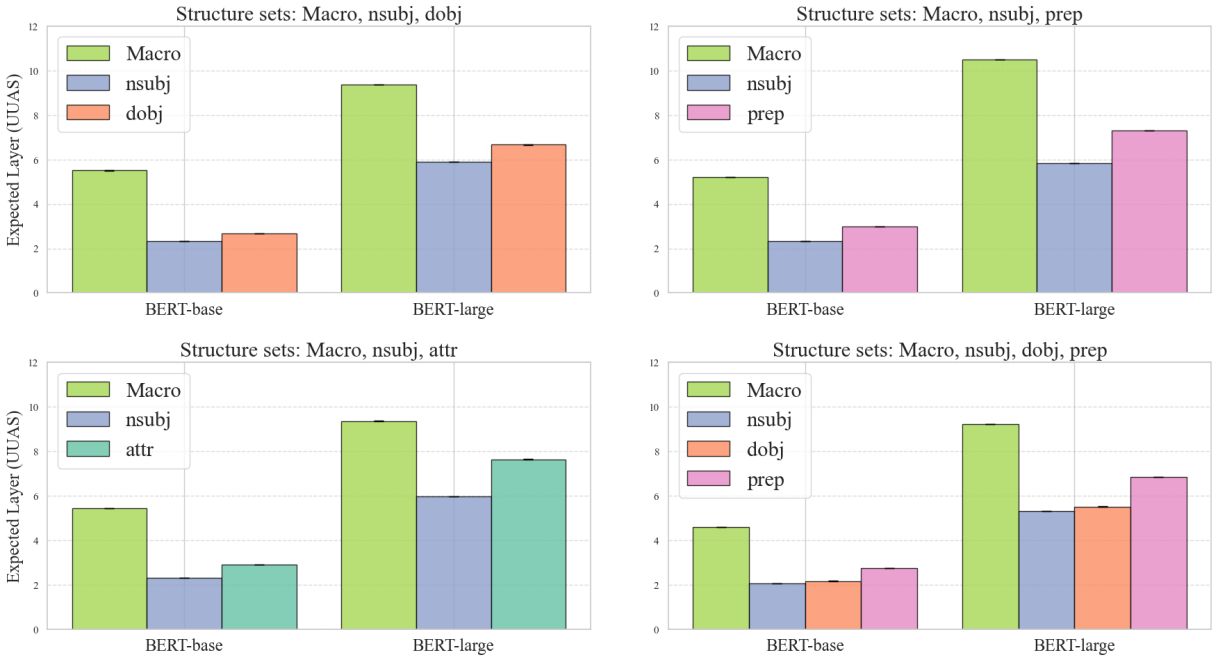


Figure 4: Expected layer for each model across different structure sets. Error bars represent standard deviation across 5 random seeds.

with the UUAS score saturating around the middle layers. BERT-large shows slightly slower improvement, likely reflecting its deeper architecture and larger capacity. These trends mostly align with previous findings (Hewitt and Manning, 2019) that neural language models tend to exhibit peak UUAS performance in their middle layers. However, unlike previous studies, we do not observe a decrease in average UUAS in later layers, which we attribute to our method of computing word embeddings as a weighted average from layer 0 to layer $\ell$ (Eq. (3)).

### 6.2 Expected Layer Across Structure Sets

Figure 4 summarizes the expected layers for each syntactic structure within the four primary structure sets (§5.1), for both BERT-base and BERT-large.

In both BERT-base and BERT-large, the macro-syntactic structure consistently exhibits the highest expected layer across all sets, whereas micro-syntactic structures such as `nsubj`, `dobj`, and `prep` tend to appear in lower layers. This suggests a *bottom-up* derivation process in which micro-syntactic structures (e.g., subject or object phrases) are constructed earlier, and these components are gradually integrated into a coherent macro-syntactic structure in later layers. This observation is consistent with prior work on BERT, which shows that local information (e.g., POS tags) is captured early, while more abstract global structures emerge later (cf. Tenney et al., 2019). Notably, this pattern holds for both BERT-base and BERT-large, although the overall expected layers are slightly higher in BERT-large—likely reflecting its deeper architecture and larger capacity.

## 7 Detailed Analysis: Subject-Verb Agreement Task

### 7.1 Experimental Setup

To investigate how the process and layers involved in syntactic structure construction relate to model performance, we conduct a detailed analysis on subject-verb agreement using sentences with intervening nouns ("attractors"), following the approach of Marvin and Linzen (2018) with some modification. We sampled 1,000 positive (grammatical) and 1,000 negative (ungrammatical) sentences. All of the sampled sentences have a subject followed by a prepositional phrase, a verb, and a direct object noun phrase. They thus are categorized into `Marco`, `nsubj`, `dobj` structure sets. In our modification of their templates, each of `Marco`, `nsubj`, `dobj` is required to contain more than one word.

a. The senators behind the brilliant architect <u>avoid</u> spicy dishes.
b. *The senators behind the brilliant architect <u>avoids</u> spicy dishes.

This ensures that we can extract meaningful subgraphs within each syntactic substructure.

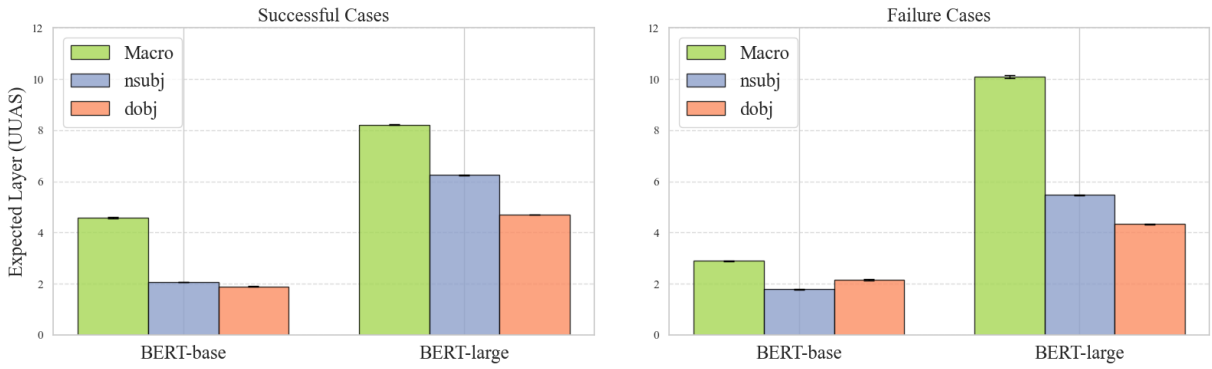


Figure 5: Expected layers for syntactic structures in successful and failed subject-verb agreement cases. Error bars show standard deviations across 5 random seeds.

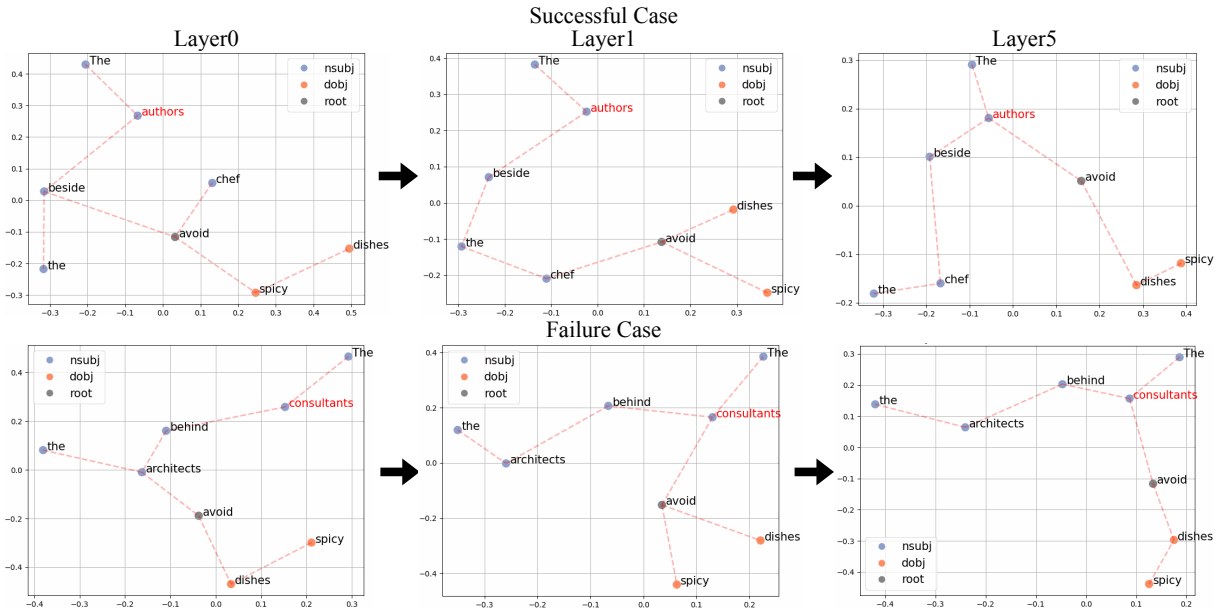


Figure 6: Derivation process visualizations for BERT-base on subject-verb agreement for a successful case ("The authors beside the chef avoid spicy dishes.") and a failure case ("The consultants behind the architects avoid spicy dishes."). Red highlights indicate the correct subject.

We first evaluate model performance on this task by computing pseudo-whole-sentence probabilities (Salazar et al., 2020). Specifically, we calculate the probability of each token by masking it one by one and then aggregate these token-level probabilities to derive an overall sentence probability. We expect the model to assign higher pseudo-probabilities to grammatical sentences compared to ungrammatical ones. We then analyze how the syntactic construction process differs between cases where the model performs well and those where it fails.

Furthermore, to visualize the evolution of syntactic structures across model layers, we employ Multidimensional Scaling (MDS). Specifically, we apply scikit-learn's MDS implementation (Pedregosa et al., 2011) with default parameters to word embeddings projected by our structural probe, allowing us to illustrate clearly how syntactic representations develop across different model layers.

### 7.2 Results

Overall, BERT-base correctly assigned higher pseudo-whole-sentence probabilities to grammatical sentences in 984 out of 1,000 examples, whereas BERT-large achieved correctness in 983 out of 1,000 cases. Despite their similar overall accuracies, we observe distinct patterns between BERT-base and BERT-large (Figure 5).

**BERT-base.** BERT-base frequently failed when macro-syntactic structures were established prematurely, potentially restricting the incorporation of essential micro-syntactic details. As illustrated in

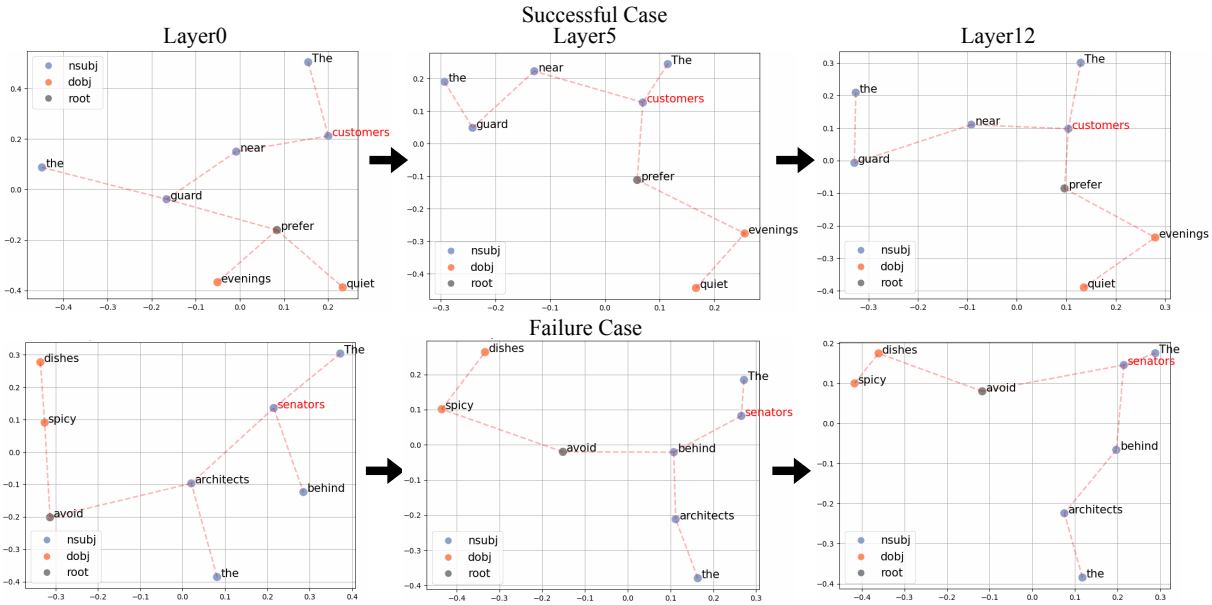


Figure 7: MDS visualizations of syntactic structure evolution in BERT-large for subject-verb agreement for a successful case ("The customers near the guard prefer quiet evenings.") and a failure case ("The senators behind the architects avoid spicy dishes."). Red highlights indicate the correct subject.

Figure 6, successful cases show a sequential pattern where BERT-base first constructs micro-syntactic structures within the subject phrase in early layers, subsequently aligning the subject (*authors*) with the verb (*avoid*) around layer 5 after stabilizing the internal subject dependencies. In contrast, failure cases reveal premature alignment of macro-syntactic structures, with the subject (*consultants*) prematurely linked to the verb (*avoid*) before fully establishing necessary micro-syntactic details. This premature commitment might have negatively impacted the overall syntactic representation, disrupting correct subject-verb agreement.

**BERT-large.** BERT-large exhibited higher expected layers for macro-syntactic structures in failure cases, suggesting delayed integration of macro-syntactic information. Figure 7 illustrates representative successful and unsuccessful cases for BERT-large. Successful predictions demonstrate early alignment of the subject (*customers*; highlighted in red) with the verb (*prefer*) around layer 5, facilitating accurate subject-verb agreement. Conversely, in unsuccessful cases, this alignment emerged considerably later (around layer 12), highlighting delayed macro-syntactic integration.

These analyses suggest an optimal intermediate range of layers for integrating macro-syntactic information. Forming macro-syntactic structures either prematurely or excessively late can negatively affect syntactic processing, highlighting the importance of appropriately timed integration for accurate predictions. These visualizations underscore how deviations from this optimal timing contribute to subject-verb agreement errors.

## 8 Discussion and Conclusion

In this paper, we introduced *Derivational Probing*—a method that integrates structural probing with an expected layer metric to trace the construction process of syntactic structures in neural language models. Our experiments revealed that BERT models tend to build micro-syntactic dependencies first and gradually assemble them into a coherent macro-structure.

BERT's bidirectional context supports a stepwise, bottom-up construction—starting with the formation of local, micro-syntactic structures and culminating in a fully integrated macro representation. These findings offer valuable insights into the internal mechanisms by which deep neural models construct syntactic trees and highlight the importance of examining layer-wise structural formation for improved model interpretability.

One promising direction for future research is to incorporate multilingual probes, which will help determine whether these syntactic structures generalize beyond English or are not mere artifacts of the particular training corpus. Another exciting direction would be to explore incremental parsing strategies in autoregressive language models as an

alternative to non-incremental derivation processes across layers (cf. Eisape et al., 2022), which could yield further insights into the syntactic knowledge of neural language models.

## Limitations

First, our experiments were conducted on only two neural language models (BERT-base and BERT-large). It remains unclear whether similar results would be obtained for larger models or other architectural variants. However, our method is applicable to any open neural model, making it feasible to extend this analysis to a broader range of models in future research.

Second, this study focused solely on English data. It is uncertain whether similar layer-wise syntactic structure construction patterns would be observed when applying our method to other languages. Nevertheless, our approach is language-agnostic, making cross-linguistic analysis an important direction for future work.

Furthermore, semantic cues may influence the results of syntactic probes. Our study does not fully account for these potential semantic confounds. Future research should consider methods to more rigorously isolate syntactic information, such as using Jabberwocky sentences as demonstrated by Maudslay and Cotterell (2021).

Lastly, our method relies on dependency parsing, primarily due to the use of the structural probe from Hewitt and Manning (2019), which analyzes distances between tokens in the embedding space. This approach is inherently tied to formalisms like dependency grammar that focus on relationships between terminal symbols (tokens). As a result, our method may not be directly applicable to other grammatical theories or parsing approaches that involve non-terminal symbols, such as constituency grammar. This limitation arises because analyzing distances between tokens does not capture the hierarchical structures represented by non-terminals. Future work could explore adapting our method or developing new probing techniques that can handle non-terminal representations to verify the generalizability of our findings.

## Ethical considerations

The training corpus is extracted from public web pages and thus could be socially biased, despite its popular use in the NLP community.

## Acknowledgments

This work was supported by JSPS KAKENHI Grant Numbers JP24H00087, JP24H00809 and JST PRESTO Grant Numbers JPMJPR21C2, JPMJPR21C8. TS was also supported by JST BOOST NAIS.

## References

Tyler A. Chang and Benjamin K. Bergen. 2024. Language Model Behavior: A Comprehensive Survey. *Computational Linguistics*, 50(1):293–350.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Tiwalayo Eisape, Vineet Gangireddy, Roger Levy, and Yoon Kim. 2022. Probing for incremental parse states in autoregressive language models. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2801–2813, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Katsuhiko Hayashi, Taro Watanabe, Masayuki Asahara, and Yuji Matsumoto. 2012. Head-driven transition-based parsing with top-down prediction. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 657–665, Jeju Island, Korea. Association for Computational Linguistics.

John Hewitt, Kawin Ethayarajh, Percy Liang, and Christopher Manning. 2021. Conditional probing: measuring usable information beyond a baseline. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1626–1639, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4129–4138, Minneapolis, Minnesota. Association for Computational Linguistics.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Jenny Kunz and Marco Kuhlmann. 2022. Where does linguistic information emerge in neural language models? measuring gains and contributions across layers. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4664–4676, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Tomasz Limisiewicz and David Mareček. 2021. Introducing orthogonal constraint in structural probes. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 428–442, Online. Association for Computational Linguistics.

Rebecca Marvin and Tal Linzen. 2018. Targeted syntactic evaluation of language models. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1202, Brussels, Belgium. Association for Computational Linguistics.

Rowan Hall Maudslay and Ryan Cotterell. 2021. Do syntactic probes probe syntax? experiments with jabberwocky probing. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 124–131, Online. Association for Computational Linguistics.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Pointer sentinel mixture models. *Preprint*, arXiv:1609.07843.

Joakim Nivre. 2004. Incrementality in deterministic dependency parsing. In *Proceedings of the Workshop on Incremental Parsing: Bringing Engineering and Cognition Together*, pages 50–57, Barcelona, Spain. Association for Computational Linguistics.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(85):2825–2830.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

R. C. Prim. 1957. Shortest connection networks and some generalizations. *Bell System Technical Journal*, 36(6):1389–1401.

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

Julian Salazar, Davis Liang, Toan Q. Nguyen, and Katrin Kirchhoff. 2020. Masked language model scoring. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2699–2712, Online. Association for Computational Linguistics.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.

Jesse Vig and Yonatan Belinkov. 2019. Analyzing the structure of attention in a transformer language model. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 63–76, Florence, Italy. Association for Computational Linguistics.

Jennifer C. White, Tiago Pimentel, Naomi Saphra, and Ryan Cotterell. 2021. A non-linear structural probe. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 132–138, Online. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. 2024. Explainability for large language models: A survey. *ACM Trans. Intell. Syst. Technol.*, 15(2).

## A The Experimental Results for GPT-2 Models

Figures 8 and 9 show the experimental results with the same experimental setup as §5, but conducted with GPT-2 small[3]/medium[4] (Radford et al., 2019). In contrast to BERT, GPT-2 (both small and medium) displays a more *parallel* derivation: the expected layer values for both micro-syntactic and macro-syntactic structures are closely aligned, suggesting parallel derivation rather than a bottom-up or top-down derivation.

## B Hyperparameters

Hyperparameters for our experiments are shown in Table 2. All models were trained and evaluated on 4× NVIDIA RTX A5000 (24GB). The total computational cost for all experiments in this paper is about 120 GPU hours.

| | |
|---|---|
| Optimizer | Adam |
| Learning rate | 1e-3 |
| Number of epochs | 40 |
| Learning rate scheduler | ReduceLROnPlateau |
| Batch size | 32 |

Table 2: Hyperparameters for our experiments

## C License of the Data and Tools

The licenses of the data and tools used in this paper are summarized in Table 3. We confirmed that all the data and the tools were used under their respective license terms.

| Data/tool | License |
|---|---|
| `spacy` (Honnibal et al., 2020) | MIT |
| `transformers` (Wolf et al., 2020) | Apache 2.0 |
| `Wikitext-103` (Merity et al., 2016) | CC-BY-SA 3.0 |

Table 3: License of the data and tools

[3] https://huggingface.co/openai-community/gpt2

[4] https://huggingface.co/openai-community/gpt2-medium

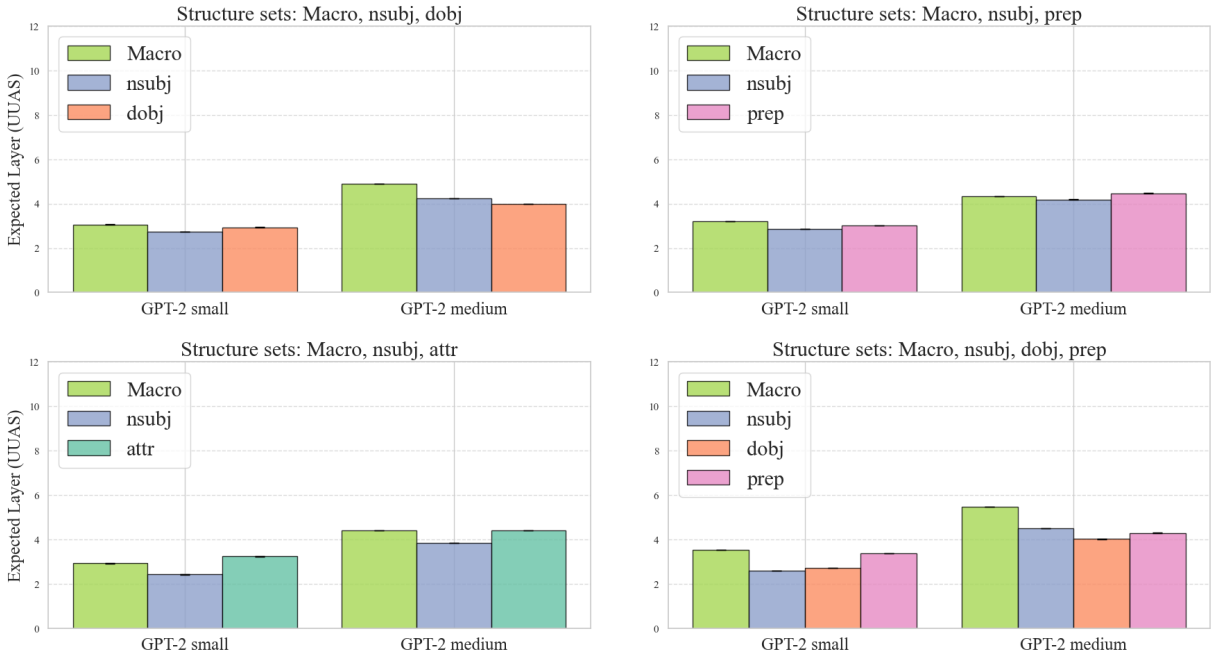


Figure 8: Expected layer for each GPT-2 model across different structure sets. Error bars represent standard deviation across 5 random seeds.

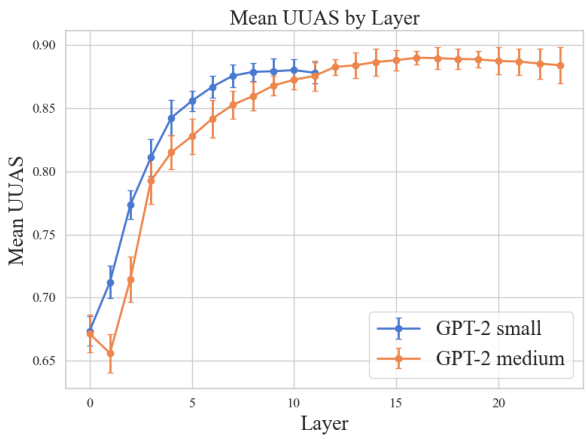


Figure 9: Global UUAS by each layer for each GPT-2 model. Error bars represent standard deviations across 5 random seeds.