AttentionRadViz: Token-Anchored Attention Profile Visualization for Transformer Head Analysis

Jared Wilber

NVIDIA

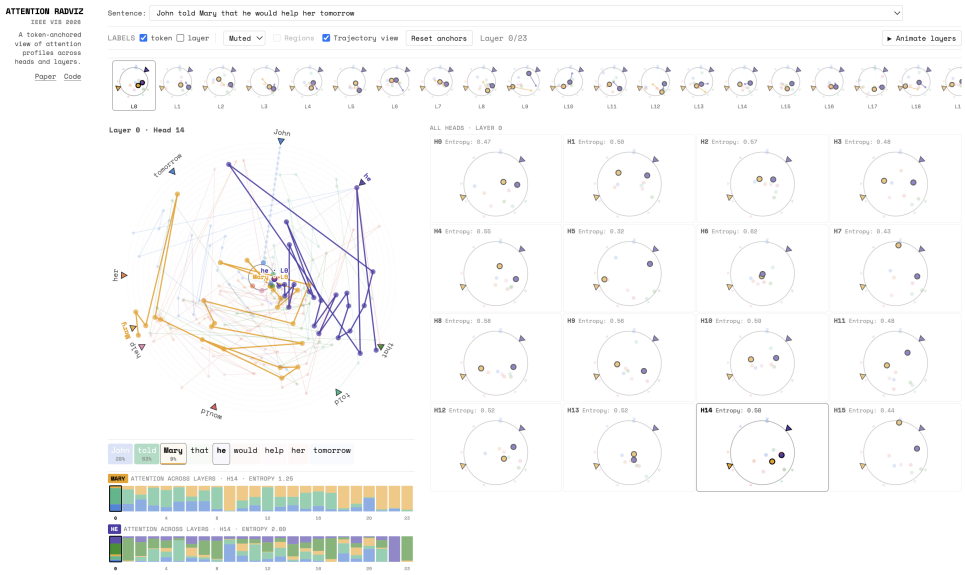

Figure 1: Interactive AttentionRadViz interface for comparing attention structure across heads and layers. The main view projects token attention profiles into a token-anchored RadViz layout with draggable anchors and trajectory trails. In this example, *her*/*Mary* form a visually distinct pattern from *he*/*John*, exposing candidate coreference-related profile structure. The stacked bar summarizes selected-token attention across layers, the right panel shows all heads in the current layer, and the top strip includes layer-wise navigation and additional UI options.

Abstract

While attention heatmaps are the standard interface for inspecting multi-head attention, they primarily expose pairwise token-to-token weights rather than supporting comparison of how multiple tokens attend over the full context or how those attention profiles evolve across layers. We present AttentionRadViz, a visual analytics tool that maps each token's attention distribution to a point in a RadViz layout whose anchors are the input tokens. In this token-anchored projection, spatial proximity indicates similarity between tokens as attenders, while anchor positions preserve a semantic reference to the input sequence. The system combines single-head inspection with layer-by-layer small multiples and radial token trajectories, helping researchers compare attention profiles across tokens, follow how those profiles change across depth, and triage heads by profile structure, readily distinguishing structured heads from collapsed or sink-like ones. We demonstrate AttentionRadViz on Pythia-410M attention exports with focused examples of interactive coreference-profile inspection, attention-sink detection, and transient cross-layer profile emergence. We also report a projection sanity check showing that RadViz preserves local attention-profile neighborhoods while retaining interpretable token anchors. AttentionRadViz is released as an open-source library.

Index Terms: Transformer interpretability, attention visualization, visual analytics, RadViz, machine learning visualization.

1 Introduction

Multi-head attention is commonly inspected through heatmaps, which provide direct access to pairwise token-to-token weights. Heatmaps are useful for local questions such as where one token attends in a particular layer and head. Comparison tasks that arise in interpretability workflows: determining whether several tokens attend in similar ways, tracking how token attention profiles evolve across depth, and deciding which heads merit closer inspection among many candidates, all require mentally integrating information across many rows, layers, and heads in a heatmap.

This limitation matters because transformer head analysis often starts as a search problem. Researchers scan for recognizable behaviors such as sink heads, previous-token heads, induction-like heads, or context-sensitive heads [9, 7]. Existing attention visualizations preserve exact weights, but they do not directly reveal row-level similarity structure among full attention profiles. To our knowledge, prior attention visualizations do not expose attention rows as comparable objects within a shared semantic coordinate system. Two tokens may behave similarly as attenders without that relationship being visually salient. The difficulty increases when the goal is not only to inspect one layer, but to understand how these attention profiles change across the network.

We address this gap with AttentionRadViz, a token-anchored visualization for comparing attention profiles within and across layers. For a fixed head and layer, each token's attention row is treated

as a distribution over the input sequence and projected into a RadViz layout whose anchors are the input tokens. Tokens with similar attention distributions appear near one another, making inter-token similarity visible. The interface complements this encoding with layer-by-layer small multiples, radial token trajectory overlays, and head-ranking scores for exploration.

We focus on four analysis tasks: comparing full attention profiles across tokens, following profile evolution across layers, identifying sink-like or otherwise uninformative heads, and surfacing candidate token relationships that merit mechanistic follow-up. AttentionRadViz is designed to support these tasks as an exploratory interface, not to establish causal explanations by itself.

Our contributions are:

- a token-anchored RadViz encoding for transformer attention rows that emphasizes similarity among attention profiles and provides a stable reference frame across layers;
- coordinated views for head-specific inspection, layer-by-layer comparison, and radial token trajectories within a shared anchor frame;
- focused use-case examples showing profile grouping, sink-head collapse, and transient cross-layer behavior;
- an open-source JavaScript library and notebook-native anywidget release [5] with reproducible experiment scripts, available at https://github.com/jwilber/AttentionRadViz.

An interactive demo page with the examples from this paper is available at https://attentionradviz.github.io.

2 Related Work

Attention visualization is a common interface for transformer inspection. BertViz visualizes attention at multiple scales through bipartite weight diagrams and remains useful for local inspection of token-to-token weights [9]. Yeh et al.'s AttentionViz [10] projects a joint embedding of query and key vectors, exposing global attention patterns across many input sequences. AttentionViz embeds learned Q/K vectors aggregated across examples in a model-level geometric space, whereas AttentionRadViz projects per-prompt attention rows into a token-anchored layout, surfacing within-head profile similarity and its evolution across layers. Attention rollout and attention flow aggregate attention across layers to estimate information flow to input tokens [1]. A recent line of pathway-style VIS systems examines how information moves through the network: the LM Transparency Tool traces token influence and information flow from the output back through heads and neurons [8], while Transformer Explainer walks through a text-generative model's generation dynamics [3]. These systems trace representation evolution and token influence across the network; AttentionRadViz is narrower and complementary, focusing on head-level attention-profile similarity within a fixed semantic frame.

Our visual encoding builds on RadViz, a multivariate projection technique that places dimensions as anchors on a circle and positions items according to weighted pull from each anchor [4]. We adapt the standard weighted-sum projection to transformer attention. Unlike optimized variants such as RadViz Deluxe [6], we keep anchors in token sequence order to preserve a semantic reference frame. Anchors are not abstract dimensions, but the tokens that receive attention, and their positions remain comparable across layers.

3 AttentionRadViz

Let a prompt contain N tokens. For layer l and head h, token i has attention distribution $a_i^{(l,h)} \in \Delta^{N-1}$, where $a_{ij}^{(l,h)}$ is the attention weight from token i to token j. Standard heatmaps visualize

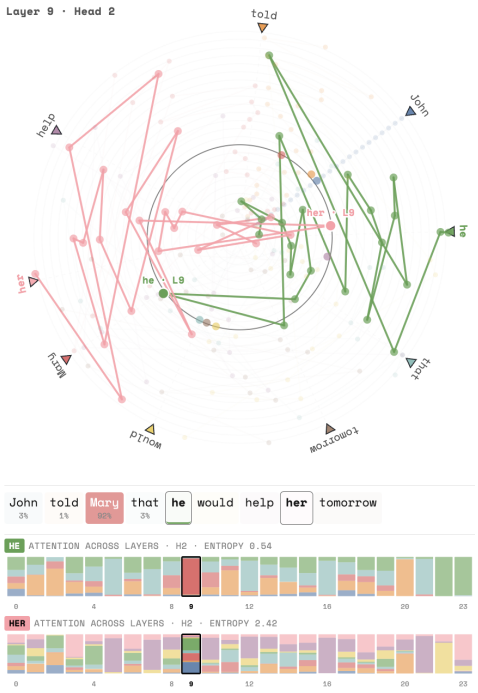

Figure 2: Cross-layer evolution of a coreference pairing. We select the two pronouns, *he* and *her*, and trace their attention profiles across the layers of head H2 in the token-anchored frame; the anchors are draggable, and here we have dragged them to better separate the *John*/*he* and *Mary*/*her* groups. The stacked bars below give each pronoun's full attention distribution layer by layer, colored by target token. Reading across depth shows that the pairing is not monotonic: layer 9 (highlighted) is the one layer where it crosses, *he* leans toward *Mary* and *her* toward *John*, while across the remaining layers, and most strongly by the final layer, the head places its greatest weight on the expected *he*/*John* and *her*/*Mary* pairs. Both the trajectory view and the stacked bars make this depth-dependent flip visible at a glance, which a single-layer heatmap would miss.

these distributions row by row. AttentionRadViz treats each row as a multivariate object.

We place one anchor for each input token on the boundary of a circle in sequence order. Given anchor positions $u_1, \ldots, u_N \in \mathbb{R}^2$, the projected position of token i is

$$x_i^{(l,h)} = \sum_{j=1}^{N} a_{ij}^{(l,h)} u_j, \tag{1}$$

since each attention row sums to one. A token is pulled toward the anchors it attends to most strongly. Tokens with similar attention distributions occupy nearby positions, while concentrated attention pushes points toward the boundary. Spatial proximity in the projection is an exploratory visual cue: two tokens near one another have similar attention rows under this weighted-sum layout, but proximity is not evidence of semantic relatedness or causal interaction. Figure 3 shows the recurring structures this makes visible at a glance, alongside the standard heatmap of the same head: repeated entity mentions cluster together, subject and predicate tokens separate, and sink heads collapse toward a single anchor.

The encoding applies to any attention matrix whose rows are distributions over a token set, including encoder self-attention and cross-attention, where the anchors would be the key-side tokens. Our interface conventions (sequence-ordered anchors and causally

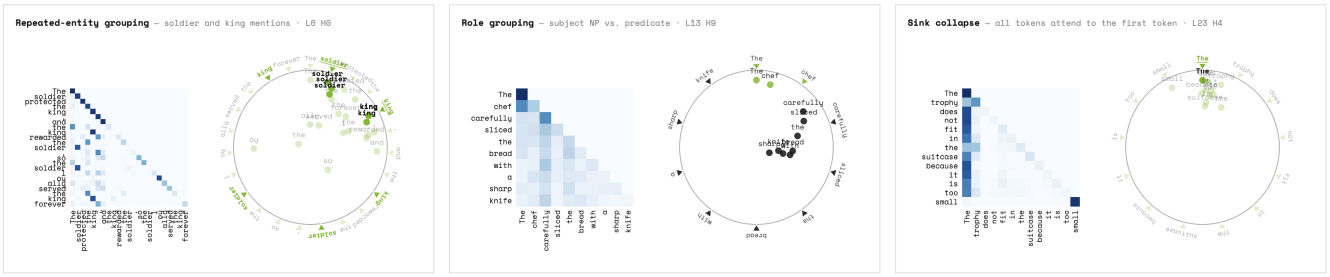

Figure 3: What the token-anchored projection reveals, shown for three heads (attention heatmap left, AttentionRadViz right within each pair). *Repeated-entity grouping* (L6 H0): the repeated mentions of *soldier* and *king* form tight clusters, indicating that these tokens share attention profiles. *Role grouping* (L13 H9): the subject noun phrase *The chef* (green) separates from the predicate tokens (black). *Sink collapse* (L23 H4): most tokens attend to the first token *The* (mean 0.66 attention from non-self rows; the argmax target for 83% of them), collapsing the projection toward the *The* anchor.

masked rows) and all reported experiments target decoder-only causal LMs. Using sequence order preserves a natural reference frame for these models and keeps anchor semantics stable across layers. The notebook widget also lets researchers move anchors interactively to separate overlapping groups and test alternate spatial hypotheses.

The interface supports two core cross-layer views for a selected head. Small multiples show one RadViz plot per sampled layer while keeping anchors fixed, so researchers can scan how profile structure changes over depth. A radial trajectory overlay keeps the same anchor frame but connects selected token positions across layers (Figure 2), making depth-dependent migration visible without scanning separate plots. Within each small multiple, a short trail also links every selected token back to its position in the preceding layer (Figure 5), so per-layer movement is legible without leaving the grid. The implementation also includes an optional difference mode that highlights adjacent-layer displacement, but we treat it as a secondary exploration aid rather than a primary paper claim.

The token-anchored frame is what makes cross-layer comparison interpretable. Projections such as t-SNE, UMAP, or PCA on attention rows produce axes that lack a semantic referent or shift as the projection is re-fit per layer. Fixing RadViz anchors to the input sequence pins the coordinate system across depth, giving trajectories and small multiples a stable reference frame.

To reduce manual inspection, we compute an informative-head score from projected spread while penalizing sink-like and self-focused heads. Let p_i be the RadViz position of token i. We define spread $= (\mathrm{var}(p_x) + \mathrm{var}(p_y))\,\mathrm{mean}_i \|p_i\|_2$ and score a head as $\mathrm{spread}(1-b)(1-s)$, where b is the first-token/BOS attention score and s is mean self-attention. The score is a triage heuristic, not a measure of causal importance. It prioritizes visually differentiated heads while down-weighting trivial first-token and self-focused layouts.

The all-heads grid provides the primary entry point for head triage (Figure 4). For a selected layer, each head is rendered as a small RadViz plot annotated with mean row entropy, allowing researchers to scan the full set of heads at a glance rather than inspect them sequentially. Collapsed or sink-like layouts are immediately distinguishable from heads with structured profile separation, supporting the workflow of narrowing from the full head space to candidates worth inspecting in cross-layer views.

4 Evaluation and Use Cases

We evaluated AttentionRadViz on Pythia-410M [2] using prompts spanning simple syntax, coreference, induction-like repetition, garden-path structure, relative clauses, semantic roles, conjunction, and long-range dependencies. The reproducible experiment bundle analyzes 3,072 head instances for projection quality and 8,832 prompt/layer/head instances for use-case candidate selection.

4.1 Profile-Neighborhood Sanity Check

We compare RadViz with PCA using pairwise Jensen-Shannon distances between attention rows as the reference geometry. Across evaluated heads, mean Spearman correlation between attention-space distances and 2D projected distances is 0.816 for RadViz and 0.832 for PCA; median values are 0.853 and 0.867. RadViz is therefore slightly weaker on global rank preservation, which is expected because PCA optimizes low-dimensional approximation while RadViz preserves semantic anchors.

We compute nearest-neighbor overlap at $k = 2$, measuring whether each token's two nearest neighbors in attention-profile space are preserved in the 2D projection. At $k = 2$, AttentionRadViz achieves higher nearest-neighbor overlap than PCA, with a mean score of 0.759 versus 0.699, and outperforms PCA in 57.6% of evaluated heads. Thus, AttentionRadViz provides comparable or better local attention-profile neighborhood preservation while retaining the interpretability of fixed token anchors.

4.2 Use-Case Examples

Following the use-case style of prior attention visualization tools [9], we demonstrate AttentionRadViz through three focused examples selected to cover the main tasks above. To avoid purely handpicked cases, we scored every loaded prompt, layer, and head for sink/collapse and structured-head behavior; scored repeated-token prompts for profile grouping; and scored configured coreference, long-range, and Winograd templates for candidate token relationships and temporal-emergence behavior. The release bundle includes the candidate-search outputs used to select the clearer row-similarity, collapse, and cross-layer examples foregrounded here.

Interactive attention-structure inspection. Figure 1 shows how AttentionRadViz combines a detailed head view with compact context for heads and layers. In the selected head, trajectories for *her* and *Mary* occupy a different part of the projection than *he* and *John*. We read this as a hypothesis rather than a finding: the proximity suggests a candidate gendered/coreferential relationship in the attention profiles that merits mechanistic follow-up, not a demonstrated one. Because trails can be shown across layers, the view also lets researchers check whether such candidate relationships appear stable, diffuse, or reorganized across depth.

Grouping structure and sink-like heads. Figure 3 contrasts structured heads with a degenerate one in the same view. Its right panel is a sink-head candidate: most query rows attend to *The*, collapsing most tokens toward the *The* anchor with low spread. Its other two panels instead show structured heads whose projections

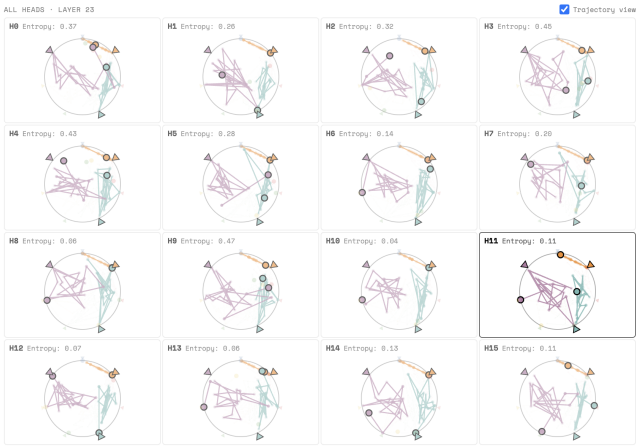

Figure 4: Small multiples of attention profiles across heads. In the core AttentionRadViz interface, the right panel renders every head of the selected layer as its own RadViz plot, so a researcher can compare and triage all heads at a glance, with structured heads standing out from collapsed or sink-like ones, rather than opening each head in turn. Here we enable trajectory trails and track the two pronouns *he* and *her* (in two contrasting colors) across layers, so each panel shows how that head treats the pair over depth: some heads keep them on distinct paths, while others overlap them.

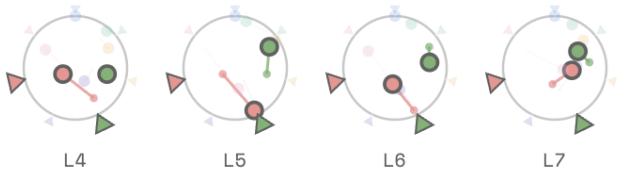

Figure 5: Layer-to-layer movement in the small multiples. A zoom into four consecutive layer panels (L4–L7) for one head: each selected token carries a short trail back to its position in the previous layer, so the trail's length and direction show how far and in which way its attention profile shifts from one layer to the next. These per-depth panels are the by-layer facet (fixed head, one panel per layer), complementing the by-head all-heads grid (Figure 4); together they let a researcher pivot between comparing heads at one depth and following one head across depth.

separate tokens into interpretable groups: repeated mentions of the same entity, and a subject noun phrase distinct from its predicate. Reading the pairs side by side shows that the projection distinguishes uninformative or degenerate heads from structured ones as clearly as it exposes the grouping itself, while the accompanying heatmaps retain the exact pairwise weights.

Tracing profile evolution across layers. Figure 2 shows that a single static layer can hide how a pairing shifts with depth. Tracing the pronouns *he* and *her* across one head, the expected *he*/*John* and *her*/*Mary* pairing dominates across most layers and is strongest at the final layer, yet layer 9 stands out as the one depth where the pairing flips, with *he* toward *Mary* and *her* toward *John*. AttentionRadViz makes such non-monotonic, depth-dependent behavior visible at a glance, which a single selected layer would miss.

5 Discussion and Limitations

AttentionRadViz is intended as a complementary view of attention, not a replacement for pairwise heatmaps. Heatmaps are still preferable when the goal is to inspect exact token-to-token weights. AttentionRadViz instead emphasizes a different set of questions: which tokens have similar attention profiles, how those profiles change across layers, and which heads exhibit interpretable structure versus collapse or dispersion.

AttentionRadViz demonstrates practical value for interpretability researchers through focused use-case examples, a reproducible notebook workflow, and an open-source release. It does not establish causal mechanisms; stronger mechanistic claims would require interventions such as activation patching or ablation.

Design choices and open questions. A single per-token color ties each token's point, anchor, trail, and stacked-bar entry together across the coordinated views, so a token is easy to follow from one panel to the next; the cost is reduced palette distinctness as N grows, which we mitigate by desaturating unselected tokens and limiting trails to the current selection. We keep the plain weighted-sum projection because attention rows are already distributions, making each position a parameter-free convex combination of anchors, and we retain sequence-ordered anchors for a stable cross-layer frame, trading the tighter separation an optimized layout might offer for comparability across depth. Two questions remain open: draggable anchors aid disambiguation but complicate reproducibility of visual findings, and the many-to-one projection would benefit from an explicit ambiguity cue, such as on-demand access to the underlying attention row.

Scaling to longer contexts. The token-anchored circle places one anchor per token, so beyond roughly 50–100 tokens anchor labels collide and individual pull vectors become ambiguous; the tool is aimed at short diagnostic prompts. Several mitigations extend its reach: *token grouping*, merging adjacent tokens into phrase- or sentence-level anchors; *anchor filtering*, showing only anchors whose aggregate received attention exceeds a threshold; and *selected-token views*, projecting only user-chosen query rows. These trade some resolution for legibility and are natural directions for supporting longer prompts.

Several limitations remain. Our examples focus on single-head analysis in one model family, leaving open questions about multi-head aggregation and comparisons across architectures. Extending our interface conventions to encoder-only and cross-attention settings, where anchors are the key-side tokens and no causal ordering applies, is likewise left to future work. Finally, while the interface is designed for exploratory use, formal user studies would be needed to quantify whether it improves researcher speed, accuracy, or insight generation relative to existing attention visualizations.

6 Conclusion

We presented AttentionRadViz, an interactive tool for visualizing transformer attention rows in a token-anchored RadViz layout. The tool supports head-specific and cross-layer inspection of full attention profiles, complementing heatmaps that emphasize exact pairwise weights. We demonstrated AttentionRadViz on Pythia-410M attention exports through use cases for interactive coreference-profile inspection, identifying sink-like heads, and tracing transient layer behavior. Quantitatively, AttentionRadViz preserved local attention-row neighborhoods better than PCA while retaining interpretable token anchors. Together, these results suggest that AttentionRadViz can help researchers triage attention heads and identify candidate token relationships for closer mechanistic analysis.

References

[1] S. Abnar and W. Zuidema. Quantifying attention flow in transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4190–4197. Association for Computational Linguistics, Online, 2020. doi: 10.18653/v1/2020.acl-main.385 2

[2] S. Biderman, H. Schoelkopf, Q. G. Anthony, H. Bradley, K. O'Brien, E. Hallahan, M. A. Khan, S. Purohit, U. S. Prashanth, E. Raff,

A. Skowron, L. Sutawika, and O. van der Wal. Pythia: A suite for analyzing large language models across training and scaling. In *Proceedings of the 40th International Conference on Machine Learning*, vol. 202, pp. 2397–2430. PMLR, 2023. 3

[3] A. Cho, G. C. Kim, A. Karpekov, A. Helbling, Z. J. Wang, S. Lee, B. Hoover, and D. H. Chau. Transformer explainer: Interactive learning of text-generative models. IEEE VIS 2024 Posters; arXiv:2408.04619, 2024. 2

[4] P. Hoffman, G. Grinstein, K. Marx, I. Grosse, and E. Stanley. DNA visual and analytic data mining. In *Proceedings of IEEE Visualization*, pp. 437–441. IEEE, Los Alamitos, CA, 1997. doi: 10.1109/VISUAL.1997.663916 2

[5] T. Manz, N. Abdennur, and N. Gehlenborg. anywidget: reusable widgets for interactive analysis and visualization in computational notebooks. *Journal of Open Source Software*, 9(102):6939, 2024. doi: 10.21105/joss.06939 2

[6] L. Nováková and O. Štěpánková. Visualization of trends using RadViz. In *Foundations of Intelligent Systems (ISMIS)*, vol. 5722 of *Lecture Notes in Computer Science*, pp. 56–65. Springer, Berlin, Heidelberg, 2009. doi: 10.1007/978-3-642-04125-9_9 2

[7] C. Olsson, N. Elhage, N. Nanda, N. Joseph, N. DasSarma, T. Henighan, B. Mann, A. Askell, Y. Bai, A. Chen, T. Conerly, D. Drain, D. Ganguli, Z. Hatfield-Dodds, D. Hernandez, S. Johnston, A. Jones, J. Kernion, L. Lovitt, K. Ndousse, D. Amodei, T. Brown, J. Clark, J. Kaplan, S. McCandlish, and C. Olah. In-context learning and induction heads. `https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/`, 2022. 1

[8] I. Tufanov, K. Hambardzumyan, J. Ferrando, and E. Voita. LM transparency tool: Interactive tool for analyzing transformer language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pp. 51–60. Association for Computational Linguistics, Bangkok, Thailand, Aug. 2024. doi: 10.18653/v1/2024.acl-demos.6 2

[9] J. Vig. A multiscale visualization of attention in the transformer model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pp. 37–42. Association for Computational Linguistics, Florence, Italy, 2019. doi: 10.18653/v1/P19-3007 1, 2, 3

[10] C. Yeh, Y. Chen, A. Wu, C. Chen, F. Viégas, and M. Wattenberg. AttentionViz: A global view of transformer attention. *IEEE Transactions on Visualization and Computer Graphics*, 30(1):262–272, 2024. doi: 10.1109/TVCG.2023.3327163 2