Are You Even Listening? Quantifying Prompt, User, and Assistant Attention Dynamics

Columbia COMS4705 Project Milestone
Keywords: *Mechanistic Interpretability, TransformerLens, Attention Dynamics, LLama-2-7b, Prompt Reliance*

Benny Attar
Department of Computer Science
Columbia University
ba2621@columbia.edu

Avi Maslow
Department of Computer Science
Columbia University
am6495@columbia.edu

Arsh Misra
Department of Computer Science
Columbia University
am6490@columbia.edu

Abstract

LLMs output can be composed of three different parts of a user journey: the system prompt (ie. "Answer in a list"), the query ("What are the best books on astrophysics?), and the assistant prior outputs (chat history). We explore and quantify how LLMs distribute attention across these three segments and quantify as a probability mass how much they each contributed to a given output. Using the TransformerLens [1] library and exploring LLaMA-2-7B [2], we have preliminary findings that in the case where a system prompt is given, it's attention mass overwhelmingly influences the model's next-token generation. To build steerable AI systems and ensure we understand their outputs, quantifying these segments is of the upmost importance.

1 Key Information to include

Our mentor is Prof. John Hewitt. We have met once as a team and once individually (Arsh). We have no external collaborators and are not sharing the project with any other courses.

2 Approach

For each dialogue turn we split the token positions into three disjoint sets: the system instruction tokens P, the user question tokens U, and the previous chat history tokens A. For the model's answer tokens $a \in A$, the transformer produces attention distributions over all earlier tokens b. Each layer and each attention head has its own attention matrix. To obtain a single interpretable measure of how much the model attends from a to b, we aggregate (sum or average) these values across all layers and heads into $W_{a,b}$. We then decompose the total attention mass used in generating the answer into three buckets:

$$\mathrm{PAM} = \frac{\sum_{a \in A} \sum_{p \in P} W_{a,p}}{Z}, \quad \mathrm{QAM} = \frac{\sum_{a \in A} \sum_{u \in U} W_{a,u}}{Z}, \quad \mathrm{SAM} = \frac{\sum_{a \in A} \sum_{a' < a} W_{a,a'}}{Z}$$

where PAM is the Prompt Attention Mass, QAM is the Question Attention Mass, and SAM is the Self-Context Attention Mass (ie. assistant chats). Here Z is the total valid attention mass, ensuring $\mathrm{PAM} + \mathrm{QAM} + \mathrm{SAM} = 1$. Our approach builds on prior work in quantifying attention flow [**?**].

We built a 300 example evaluation dataset from Alpaca [3], FLAN [4], and ShareGPT [5], filtering for unique system prompts by means of constraints in each (see Appendix Figure 1). Using the TransformerLens library [1], we instrumented LLaMA-2-7B with attention hooks to extract raw attention distributions at every token position during the forward pass. This generated 272,384 data points (266 examples $\times$ 32 layers $\times$ 32 heads). No fine-tuning was performed. After obtaining attention matrices for each head in every layer, we compute PAM/QAM/SAM scores by summing the aggregated attention weights $W_{a,b}$ over the appropriate token groups, then dividing by total attention mass Z. Due to LLaMA-2-7B's 4096-token limit, we dropped examples with overly long segments, resulting in a final dataset of 266 examples. (schema in Appendix Tables 1-3).

3 Experiments

3.1 Data

Alpaca has short prompts (mean=19.3 tokens), variable queries (mean=25.1 tokens), and minimal assistant context (mean=1.0 token); FLAN has very long prompts (mean=273.7 tokens) with minimal queries and assistant context (mean=1.0 each); ShareGPT has moderate prompts (mean=54.6 tokens), long queries (mean=255.8 tokens), and extensive multi-turn assistant context (mean=888.5 tokens). These instruction-tuned datasets exhibit different roles of scaling and instruction patterns [6].

3.2 Evaluation Method and Baselines

We evaluate attention dynamics using: (1) PAM/QAM/SAM scores as defined in Section 2; (2) Layer-wise analysis tracking how attention evolves across 32 layers; (3) Head specialization identifying heads with PAM > 0.6 as "prompt-specialists"; (4) Statistical significance testing via one-way ANOVA and Pearson correlation; (5) Length-attention correlations testing whether longer segments receive proportionally more attention.

3.3 Results

Overall Attention Distribution (Appendix Figure 2, Table 4): Alpaca and FLAN show extremely high PAM (0.824 and 0.906 respectively), indicating overwhelming attention to system prompts. ShareGPT exhibits much higher SAM (0.274), reflecting multi-turn conversational structure. ANOVA confirms significant differences in distributions across datasets for all metrics ($p < 0.0001$).

Layer-wise Evolution (Appendix Figure 3): In Alpaca and FLAN, PAM peaks early (Layers 3-4) and remains high throughout. ShareGPT shows SAM dominating early layers (Layer 1: 0.665), then PAM increasing in deeper layers (Layer 30: 0.575), suggesting the model first attends to dialogue history before incorporating prompt information.

Head Specialization (Appendix Figure 4, Table 5): We identified 911 prompt-specialist heads but only 1 user-specialist and 1 self-specialist head. Top prompt-specialists concentrate in late layers (27-30) with PAM > 0.90, suggesting instruction-following is mediated by a small subset of highly specialized heads.

Correlations (Appendix Figures 5-7, 8): Attention mass correlates significantly with segment length. Alpaca shows moderate PAM-length ($r = 0.30, p = 0.003$) and strong QAM-length correlation ($r = 0.76, p < 0.0001$). ShareGPT exhibits strong correlations across all metrics, especially SAM vs assistant length ($r = 0.78, p < 0.0001$), raising concerns about length bias. PAM and QAM are strongly negatively correlated (Alpaca: $r = -0.89$; FLAN: $r = -0.91$; ShareGPT: $r = -0.50$), indicating attention to prompts comes at the expense of queries.

4 Future work

As for the remainder of the project, our work shifts from infrastructure to analysis and causality. We move from correlation to causality via PCL-style ablations, a shift we introduced during planning once we realized attention alone was not sufficient to explain behavior. By zeroing Prompt $\rightarrow$ Assistant attention edges during decoding and measuring the change in output quality and constraint adherence, we can determine whether attention to the prompt actually causes instruction-following. We will compare LLaMA-2-Chat to base models and include randomized/shuffled prompts as controls. If we have time, we will conduct the same expirement on the same data on another open source model, potentially Mistral 7B. Then, we will integrate all components—attention metrics, behavior labels, and causal effects—into the final report, analyzing how PAM, PCL, and instruction-following interact. Our plan has narrowed from the original proposal to focus on explicit-constraint subsets and expanded to incorporate causal testing, both changes made because early-week work revealed they were essential for producing interpretable results.

References

[1] Neel Nanda and Joseph Bloom. Transformerlens. 2022.

[2] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023.

[3] Rohan Taori et al. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`, 2023.

[4] Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V. Le, Barret Zoph, Jason Wei, and Adam Roberts. The flan collection: Designing data and methods for effective instruction tuning, 2023.

[5] ShareGPT Dataset. Community-generated conversational data for instruction tuning. `https://sharegpt.com`, 2023.

[6] Changjiang Gao, Shujian Huang, Jixing Li, and Jiajun Chen. Roles of scaling and instruction tuning in language perception: Model vs. human attention. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023.

5 Appendix

5.1 Constraint Distribution

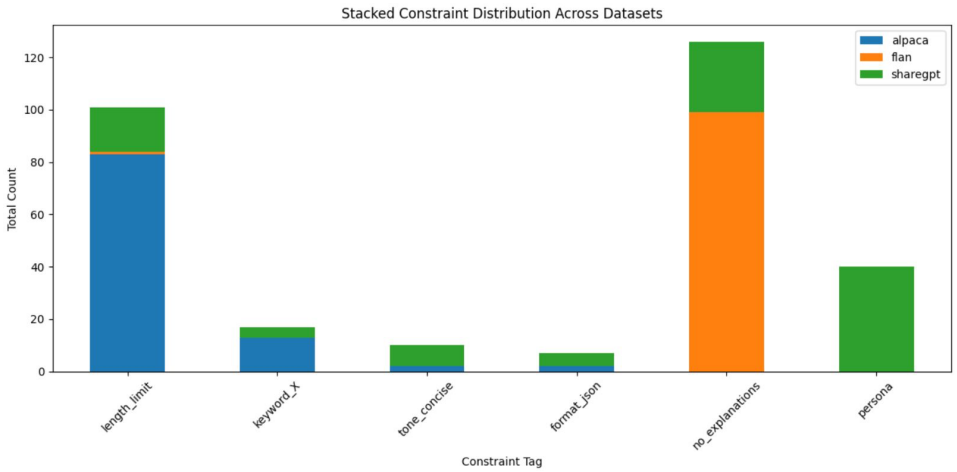

Figure 1: Constraint Distribution Across Our Datasets

5.2 JSONL Output Schema

Field	Type	Description	Example value
`id`	string	Sample identifier	`"alpaca:22052"`
`dataset`	string	Source dataset name	`"alpaca"`
`constraint_tags`	list[string]	Optional constraint labels	`[keyword_x]`
`seq_len`	int	Total token length of the dialogue turn	22
`p_len`	int	# system/prompt tokens	19
`u_len`	int	# user-question tokens	1
`a_len`	int	# answer tokens	1
`layers`	list[layer]	Per-layer attention statistics	see Table 2

Table 1: Schema of a single JSONL record in the attention-dynamics dataset.

Field	Type	Description	Example value
`layer`	int	Layer index (0-based)	`0`
`PAM`	float	Layer-averaged Prompt Attention Mass	`0.6838`
`QAM`	float	Layer-averaged Question Attention Mass	`0.0976`
`SAM`	float	Layer-averaged Self-Context Attention Mass	`0.1226`
`heads`	list[head]	Per-head attention statistics in this layer	see Table 3

Table 2: Schema of a `layers` entry (example shown for layer 0).

Field	Type	Description	Example value
`head`	int	Head index (0-based)	`0`
`PAM`	float	Prompt Attention Mass for this head	`0.5002`
`QAM`	float	Question Attention Mass for this head	`0.1630`
`SAM`	float	Self-Context Attention Mass for this head	`0.1520`

Table 3: Schema of a `heads` entry (example shown for head 0 of layer 0).

5.3 Exploratory Data Analysis: Figures and Tables

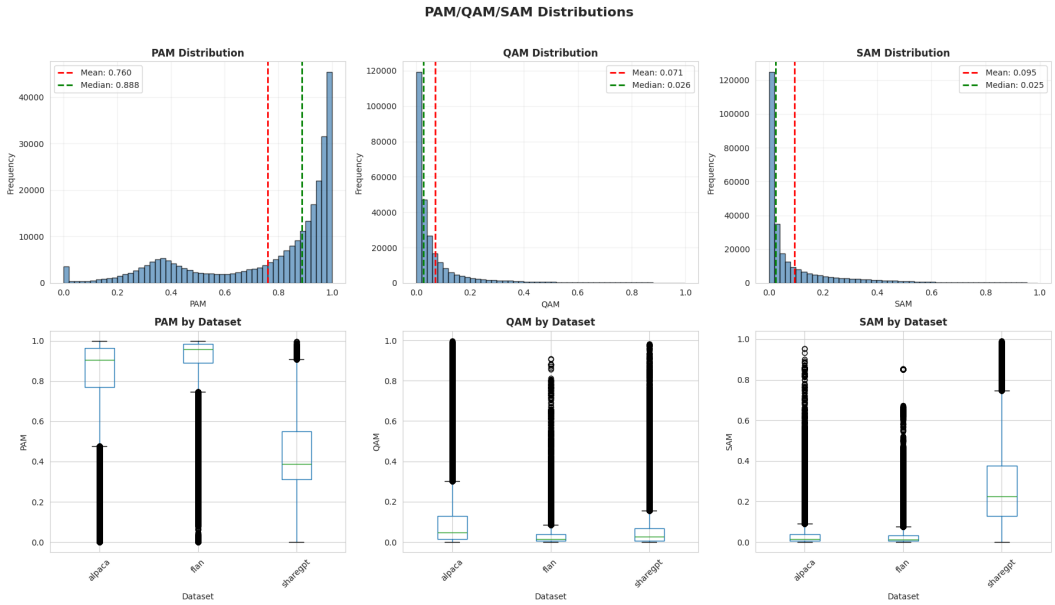

Figure 2: Distribution of PAM, QAM, and SAM across all datasets. Top row shows overall histograms; bottom row shows dataset-specific box plots. Note the high PAM in Alpaca and FLAN vs. high SAM in ShareGPT.

Dataset	PAM (mean ± std)	QAM (mean ± std)	SAM (mean ± std)
Alpaca	0.824 ± 0.205	0.112 ± 0.162	0.038 ± 0.072
FLAN	0.906 ± 0.138	0.035 ± 0.061	0.032 ± 0.061
ShareGPT	0.441 ± 0.226	0.062 ± 0.106	0.274 ± 0.193
Overall	0.760 ± 0.257	0.071 ± 0.131	0.095 ± 0.131

Table 4: Summary statistics for PAM, QAM, and SAM by dataset. Alpaca and FLAN exhibit very high prompt attention, while ShareGPT shows substantial self-attention.

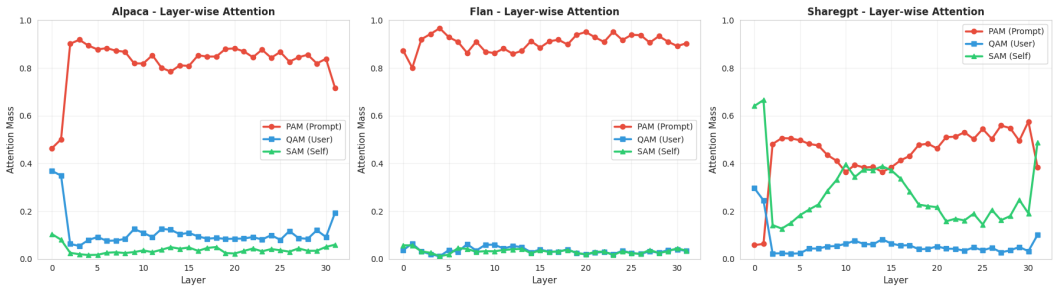

Figure 3: Layer-wise attention patterns for each dataset. Alpaca and FLAN maintain high PAM throughout, while ShareGPT shows a striking transition from high SAM in early layers to moderate PAM in later layers.

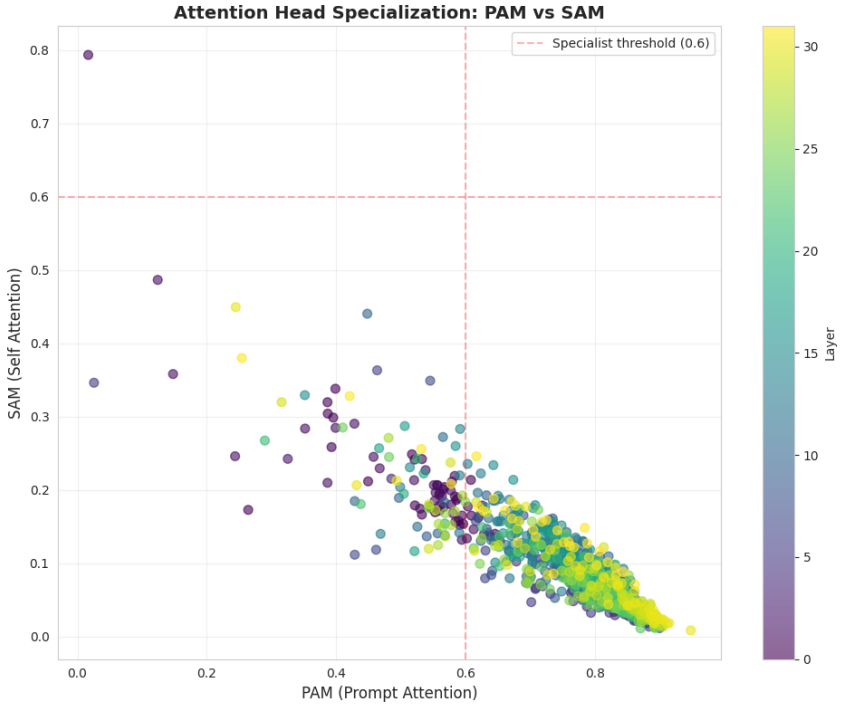

Figure 4: Scatter plot of PAM vs SAM for all 1,024 attention heads across 32 layers. Most heads cluster in the high-PAM region, with very few specializing in SAM or QAM. Color indicates layer depth.

Layer	Head	PAM	QAM
30	13	0.948	0.004
30	17	0.915	0.005
30	20	0.911	0.019
30	3	0.907	0.012
27	31	0.905	0.011

Table 5: Top 5 prompt-specialist attention heads by PAM score.

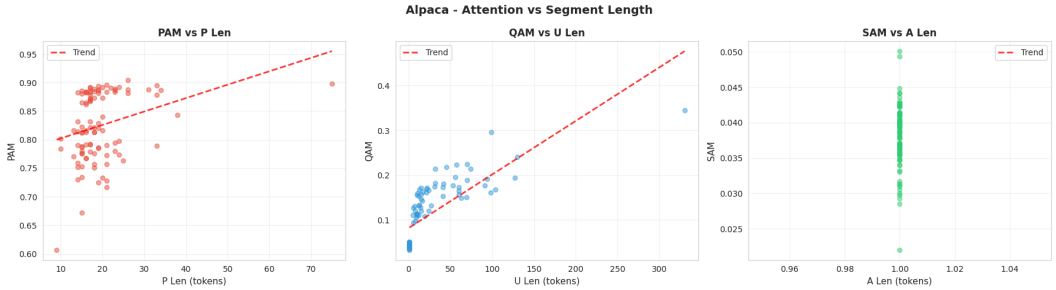

Figure 5: Alpaca: PAM vs prompt length ($r = 0.30, p = 0.003$); QAM vs user length ($r = 0.76, p < 0.0001$). Longer segments receive more attention.

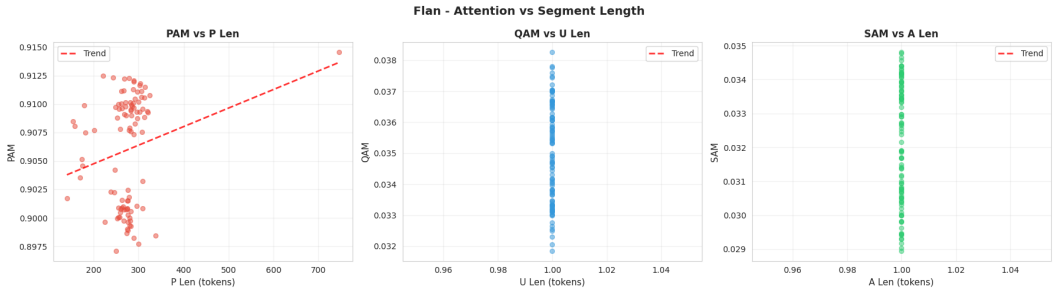

Figure 6: FLAN: PAM vs prompt length ($r = 0.21, p = 0.041$). User and assistant lengths are constant (1 token each), so no correlation can be computed.

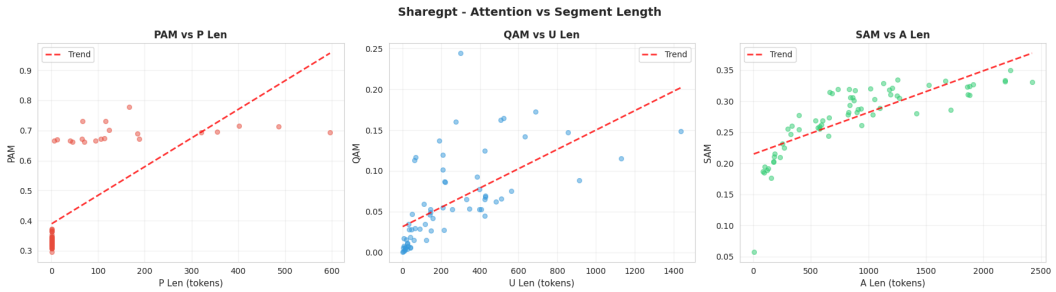

Figure 7: ShareGPT: Strong positive correlations across all segments: PAM vs p_len ($r = 0.69, p < 0.0001$), QAM vs u_len ($r = 0.63, p < 0.0001$), SAM vs a_len ($r = 0.78, p < 0.0001$).

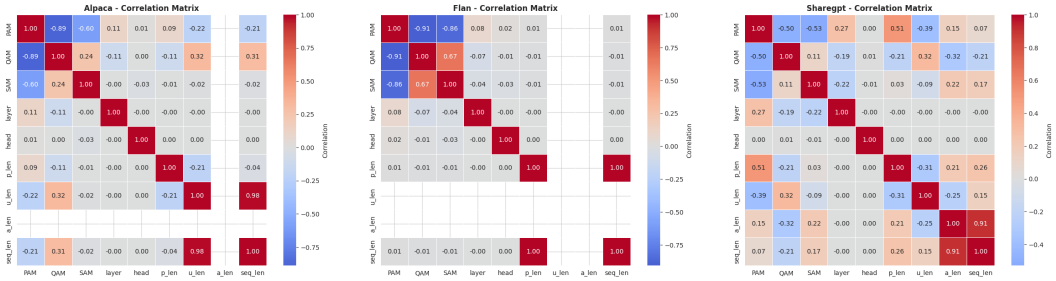

Figure 8: Correlation matrices for each dataset. Note the strong negative correlation between PAM and QAM/SAM, reflecting the zero-sum nature of attention allocation.