

Unveiling and Harnessing Attention Dynamics in Large Language Models: Frameworks for Enhanced Dialogue, Reasoning, and Efficiency

廖, 柄力

<https://hdl.handle.net/2324/7396256>

出版情報 : Kyushu University, 2025, 博士 (情報科学) , 課程博士
バージョン :
権利関係 :



KYUSHU UNIVERSITY

DOCTORAL THESIS

**Unveiling and Harnessing Attention
Dynamics in Large Language Models:
Frameworks for Enhanced Dialogue,
Reasoning, and Efficiency**

Author:

Liao BINGLI

Supervisor:

Associate Prof. Danilo
Vasconcellos VARGAS

*A thesis submitted in fulfillment of the requirements
for the degree of Doctor of Philosophy*

in the

Laboratory of Intelligent Systems
Department of Information Science and Electrical Engineering

July 25, 2025

Declaration of Authorship

I, Liao BINGLI, declare that this thesis titled, “Unveiling and Harnessing Attention Dynamics in Large Language Models: Frameworks for Enhanced Dialogue, Reasoning, and Efficiency” and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: Liao Bingli

Date: 2025-June-17th

“What I cannot create, I do not understand.”

Richard Feynman

KYUSHU UNIVERSITY

Abstract

Department of Informatics
Department of Information Science and Electrical Engineering

Doctor of Philosophy

**Unveiling and Harnessing Attention Dynamics in Large Language Models:
Frameworks for Enhanced Dialogue, Reasoning, and Efficiency**

by Liao BINGLI

Large Language Models (LLMs) have revolutionised Artificial Intelligence but face persistent challenges in generating immersive dialogue, performing complex reasoning, and maintaining computational efficiency. This thesis posits that these challenges are deeply rooted in the internal dynamics of the attention mechanism, the core component of the Transformer architecture. We adopt a two-phase approach: first, unveiling these dynamics through extensive empirical analysis, and second, harnessing the resulting insights to develop principled frameworks. The ‘unveiling’ phase analyzes a diverse range of modern LLMs (7B-72B), including their pre-training lifecycle. We identify, visualise, and quantify several key phenomena: (1) the Hierarchical Attention Cascade (HAC) and ‘anchor words’ that govern information flow; (2) Attention Skew, where focus concentrates on non-semantic tokens; and (3) the emergence and stabilisation of Attention Isotropy—high cross-layer similarity—which reveals computational redundancy and functional layer grouping. In the ‘harnessing’ phase, these observations directly inform the design of three novel applications. First, for immersive dialogue, we introduce a Card-Based Framework that leverages HAC and anchor words. Through knowledge distillation and fine-tuning, it enables smaller 7B models to generate high-quality, persona-consistent dialogues, validated by both GPT-4 and human evaluation. Second, for complex reasoning, we develop an Attention Optimisation Algorithm that counteracts attention skew. By emulating early-layer patterns, this inference-time method extends the token computation path during Chain-of-Thought, enhancing reasoning capabilities on non-STEM benchmarks without prompt modification. Third, for efficiency, we validate the Shared Attention (SA) framework. By exploiting the empirically observed isotropy, SA reuses attention weights across strategically chosen layer spans. This method yields substantial computational savings in terms of both FLOPs (e.g., over 100 MFLOPs per 2k sequence) and K-cache memory, while revealing a critical performance trade-off on mathematical reasoning tasks that rely on subtle, layer-specific patterns. Collectively, this work provides a unified understanding of attention dynamics and demonstrates that an analysis-guided approach can lead to practical, effective solutions for enhancing LLM capabilities and efficiency. The thesis contributes both fundamental insights into the inner workings of LLMs and validated frameworks for their improvement.

Acknowledgements

The journey to completing this doctoral thesis has been long and challenging, but it is one I did not walk alone. I would like to take this opportunity to express my deepest gratitude to the many people who have guided, supported, and inspired me throughout my PhD studies.

First and foremost, I wish to express my sincere gratitude to my supervisor, Professor Danilo Vasconcellos Vargas. Your invaluable guidance, intellectual freedom, and unwavering support have been the cornerstones of this work. Thank you for your patience during my moments of confusion, for your insightful feedback that sharpened my ideas, and for always encouraging me to pursue ambitious research questions. Your mentorship has not only shaped this thesis but has profoundly influenced my development as a researcher. I could not have asked for a better advisor.

I would also like to extend my sincere thanks to the members of my thesis committee: Professor Tomiura Yoichi and Professor Atsushi Shimada. Your insightful questions, constructive criticism, and expert perspectives have significantly improved the quality of this dissertation. Thank you for your time and for the valuable discussions during our meetings and at my defense.

My time at the Laboratory of Intelligent Systems at Kyushu University was made truly special by my colleagues and lab mates. I am grateful for the stimulating discussions, the late-night debugging sessions, and the shared camaraderie.

This research was made possible through the generous financial support from the Kyushu University Innovator Fellowship from Graduate School and Faculty of Information Science and Electrical Engineering. I am deeply grateful for these opportunities.

Beyond the academic world, I am indebted to my friends who provided a much-needed escape and a constant source of support. Thank you for the laughter, the adventures, and for always being there to listen, celebrate successes, and offer encouragement during setbacks.

Thank you all.

Contents

Declaration of Authorship	iii
Abstract	vii
Acknowledgements	ix
1 Introduction	1
1.1 Motivation: The Rise and Challenges of Large Language Models	1
1.2 The Central Role of the Attention Mechanism	2
1.3 Problem Statement	2
1.4 Research Questions	3
1.5 Approach: Unveiling and Harnessing Attention Dynamics	3
1.6 Key Contributions	4
1.7 Thesis Outline	4
2 Background and Literature Review	7
2.1 The Transformer Architecture	7
2.1.1 Encoder-Decoder and Decoder-Only Models	7
2.1.2 Core Components	8
2.2 The Attention Mechanism	8
2.2.1 Scaled Dot-Product Self-Attention	8
2.2.2 Multi-Head Attention (MHA)	8
2.2.3 Attention Variants: MQA and GQA	8
2.3 Large Language Models (LLMs)	9
2.3.1 Scaling Laws	9
2.3.2 The LLM Lifecycle: Training and Adaptation	9
2.3.3 Key Model Architectures	9
2.4 LLM Interpretability and Attention Analysis	10
2.5 Attention Applications and Challenges: Related Work	10
2.5.1 Dialogue Systems and Personalisation	10
2.5.2 LLM Reasoning	11
2.5.3 LLM Efficiency	11
2.6 Research Gaps	11
3 Methodologies for Analysing Attention Dynamics	13
3.1 Overview of Analytical Approach	13
3.2 LLM Interpretability and Attention Visualization	13
3.3 Data Sources and Model Selection	15
3.4 Attention Visualisation: Word-wise Heatmaps	15
3.5 Quantifying Attention Patterns	16
3.5.1 Cross-Layer Similarity: Cosine Similarity	16
3.5.2 Attention Variance Analysis	17
3.6 Key Observed Dynamics	18

3.6.1	Information Flow: Hierarchical Attention Cascade (HAC) and Anchor Words	18
3.6.2	Attention Skew and Non-Semantic Tokens	18
3.6.3	Attention Isotropy and Functional Layer Grouping	19
3.6.4	Evolution and Stabilisation During Pre-training	20
3.7	Summary of Unveiled Attention Principles	20
4	Harnessing Attention for Enhanced Persona-Driven Dialogue	23
4.1	Introduction: Immersion and Persona in Computational Storytelling	23
4.2	The Card-Based Framework	24
4.2.1	Character Cards	24
4.2.2	Scene Cards	24
4.3	Data Generation via Knowledge Distillation	24
4.4	Leveraging Attention Dynamics	26
4.4.1	Prefix Marker Tokens for Trait Emphasis	26
4.4.2	Context Maintenance via HAC and Anchor Words	27
4.5	Experimental Setup	28
4.6	Results and Analysis	29
4.6.1	Qualitative Case Study and Discussion	30
4.7	Summary and Conclusion	30
5	Harnessing Attention for Enhanced Reasoning	33
5.1	Introduction: The Challenge of LLM Reasoning	33
5.2	Identifying Attention Inefficiencies in Reasoning	33
5.3	Attention Optimisation Algorithm	34
5.3.1	Motivation: Counteracting Skew	35
5.3.2	Algorithm Formulation	35
5.3.3	Mechanism: Extending Token Computation	35
5.4	Experimental Setup	36
5.4.1	Models and Benchmarks	36
5.4.2	Evaluation Methodology	36
5.5	Results and Analysis	36
5.5.1	Impact on Token Length	37
5.5.2	Reasoning Performance	37
5.5.3	Qualitative Analysis	38
5.6	Conclusion	39
6	Harnessing Attention for Enhanced Efficiency	41
6.1	Introduction: Computational and Memory Bottlenecks	41
6.2	Foundational Analysis: Attention Isotropy and Stability	41
6.3	The Shared Attention (SA) Framework	44
6.3.1	Algorithm: Cross-Layer Weight Reuse	44
6.3.2	Comparison with K/V Sharing Methods	44
6.3.3	Computational and Memory Benefits	44
6.4	Experimental Setup	45
6.5	Results and Analysis	46
6.5.1	Direct Application	47
6.5.2	Fine-tuning Adaptation	48
6.5.3	Efficiency Gains	48
6.6	Conclusion	48

7	Discussion and Synthesis	51
7.1	Revisiting the Research Questions	51
7.2	Synthesis of Findings on Attention Dynamics	52
7.3	Synthesis of Frameworks and Applications	53
7.4	Theoretical Implications	53
7.5	Practical Implications	54
7.6	Limitations of the Study	54
8	Conclusion and Future Work	57
8.1	Summary of the Thesis	57
8.2	Future Research Directions	57
8.3	Concluding Remarks	58
A	Generated Dialogue Examples	59
B	GPT-4 Evaluation Prompt	61
C	Additional Experimental Results: Cross-Layer Attention (CLA)	65
	Bibliography	67

List of Figures

3.1	A sample word-wise attention heatmap from a decoder-only transformer. The x-axis represents the Key tokens and the y-axis the Query tokens. The lower-triangular shape is due to the causal mask. Bright spots highlight high attention scores, revealing tokens that act as critical information hubs.	16
3.2	Layer-wise attention pattern similarity for the Llama 3 8B model, visualized as a 3D scatter plot. The x and y axes represent the model's layer indices, while the z-axis and color scale (from purple/low to yellow/high) indicate the cosine similarity between the layers. The plot reveals distinct functional blocks: middle-to-late layers form a large, highly similar (isotropic) group, as indicated by the cluster of high-value yellow points.	17
3.3	Conceptual illustration of an Anchor Word. (Left) A simplified attention matrix where the second token receives high attention from subsequent tokens (values in red column) and aggregates information. (Right) The corresponding heatmap visualization, where the red vertical columns indicate the anchor tokens that consistently receiving high attention.	18
3.4	Quantitative analysis of Attention Skew across Llama3, Qwen2, and Mistral models. The bar chart displays the mean attention received by high-frequency, non-semantic tokens, plotted on a logarithmic scale against a baseline.	19
3.5	Layer-wise attention pattern similarity matrix for the Llama2-7B model on the MMLU dataset. The color of each cell at position (i, j) represents the cosine similarity between the attention patterns of layer i and layer j. High similarity (≈ 1.0) is shown in red, and low similarity in blue. The dark red diagonal confirms each layer's perfect self-similarity. The large, reddish block in the middle-to-late layers provides strong visual evidence for Attention Isotropy.	20
4.1	The format template for a Scene Card. This structured format provides the model with key narrative elements, including location, character goals, and obstacles, to guide the dialogue generation process. . .	24
4.2	Kernel Density Estimation (KDE) of dialogue quality scores. The plot compares scores from dialogues generated with a conventional "Original Prompt" (blue) against our Augmented Prompting with Post-Descriptions (AP-PD) method, labelled "Advanced Prompt" (red). The rightward shift and tighter clustering of the red curve visually confirm that the AP-PD method produces higher and more consistent quality scores, aligning with the reported higher mean (8.8 vs. 8.5) and lower standard deviation (0.38 vs. 0.53).	25

4.3	The modular structure of the complete prompt used for dialogue generation. It combines the system prompt, Scene and Character Cards, and the AP-PD rationale into a single, comprehensive input for the teacher model.	26
4.4	An example Character Card illustrating the use of the '@' prefix marker. Each key attribute, such as '@Occupation' and '@Speaking Style', is prepended with the marker to enhance its salience for the attention mechanism during fine-tuning.	27
4.5	Attention analysis for marker tokens within a Scene Card. (Left) The line plot shows the average attention scores from the dialogue portion of the prompt back to each token in the Scene Card, revealing sharp peaks for the key semantic tokens ('location', 'object', 'goal', 'obstacle') that immediately follow their markers. (Right) The corresponding attention heatmap visually confirms these patterns, with bright vertical stripes indicating that these marked tokens act as information hubs, consistently attended to throughout the dialogue generation. . . .	27
4.6	Visualization of the Hierarchical Attention Cascade (HAC) and anchor words during dialogue generation. The right panel shows a magnified view where character names (e.g., "Benjamin," "Isabelle"), highlighted by red circles, serve as anchor words. These anchors attend to previous turns to gather context. The bright vertical stripes demonstrate that tokens within each new turn strongly attend back to their local anchor (the character's name), effectively propagating conversational context turn-by-turn.	28
4.7	Boxplot showing the distribution of dialogue quality scores from the GPT-4 evaluation. The plot compares the performance of the two fine-tuned student models (SFT-Baichuan-7B and SFT-Llama2-7B) against the teacher model (GPT-3.5). The median scores (purple lines) and interquartile ranges demonstrate that the smaller, specialized models achieve performance highly competitive with the larger teacher model.	29
4.8	Side-by-side comparison of generated dialogues from the same unseen prompt (Character Cards: Captain Williams, Clara; Scene Card: An ancient tomb). Dialogue 1 is from the fine-tuned SFT-Llama2-7B, and Dialogue 2 is from the teacher model, GPT-3.5.	31
5.1	Visualization of attention skew across different layers of the fine-tuned model. The figure displays three attention heatmaps (left, center, right), where the y-axis represents Query positions and the x-axis Key positions. The prominent vertical stripes in each plot illustrate attention skew: a few Key tokens receive a disproportionately high amount of attention from many subsequent Query tokens. The logarithmic color scale (left panel) emphasizes that while most attention scores are very low (deep blue), the scores for these "skew" tokens are orders of magnitude higher, diverting attention from other potentially more semantic tokens.	34
5.2	Architectural diagram of the proposed attention modulation algorithm. The self-attention pattern from a designated top layer is used as a guiding mask to modulate the attention mechanisms in subsequent middle layers during inference. This process refocuses downstream attention based on the global patterns established in early layers. . . .	35

5.3	Comparison of the average token length per response across all MMLU subjects. The blue bars represent the "Original" baseline (Zero-shot CoT), while the green bars represent the "Modified" method using our attention modulation algorithm. The consistent increase in token length (green bars are taller than blue bars) for every subject provides strong evidence that the algorithm encourages a longer, more detailed reasoning process.	37
5.4	A conceptual illustration of a "reasoning graph." For a model to solve a problem, it must identify and correctly operate on the key nodes (variables like 'x', 'y') and edges (conditions like 'x >= 0' or operations like 'x*2'). Attention Skew can distract the model from these semantically critical components, causing a failed traversal.	38
5.5	A qualitative case study from the MMLU benchmark (College Biology) demonstrating the algorithm's impact on reasoning. The figure contrasts the output of the baseline Llama-2 13B model (middle panel, purple) with the output from the same model enhanced by our attention modulation algorithm (bottom panel, red). The baseline model provides a flawed reasoning process and arrives at an incorrect conclusion (Lyase). In contrast, the algorithm-enhanced model generates a more detailed, step-by-step analysis, enabling it to self-correct its initial assessment and identify the correct answer (Transferase). This illustrates the algorithm's ability to improve the model's self-correction capabilities.	40
6.1	Layer-wise attention similarity matrix for the Baichuan2-7B model (at 2.6T pre-training tokens). Each cell represents the cosine similarity between two layers (high similarity in yellow, low in purple). The plot reveals distinct functional blocks, annotated as Groups 1-4. Notably, Group 3 forms a large, highly isotropic core of middle-to-late layers that exhibit strong internal similarity. This stable, redundant structure is the basis for our efficiency optimizations.	42
6.2	Evolution of layer-wise attention similarity during the pre-training of the Baichuan2-7B model. The four plots show similarity matrices at different training checkpoints: 220B, 1100B, 1980B, and 2640B tokens (from left to right). This visualization demonstrates the emergence of stable functional groups as training progresses.	43
6.3	Weighted cumulative attention variance for Llama3-8B-Instruct (left) and Llama2-7B-Chat (right). The x and y axes represent the starting layer index for the cumulative calculation, while the z-axis shows the calculated variance. The plots clearly show that variance is highest when starting from the earliest layers (high values in the corner) and decreases sharply for later layers. This visualizes the instability of early layers compared to the stable, low-variance nature of the latter, isotropic layers.	43
6.4	Conceptual comparison between Shared Attention (SA) and methods like Cross-Layer Attention (CLA).	45
6.5	Conceptual diagram illustrating the application of Shared Attention (SA) to different layer spans. The top panel depicts a shorter 4-layer span (SA:27-30), while the bottom panel shows a longer 8-layer span (SA:23-30).	46

- A.1 An example dialogue generated by the fine-tuned Baichuan-7B model. The model successfully maintains the personas of "Dr. Benjamin" and "Ms. Isabelle," incorporates actions (e.g., "nodding in agreement"), and progresses the narrative logically according to the scene card's goals and obstacles. 60
- A.2 An example of output generated by the untuned baseline Baichuan-7B model given a similar prompt. The output is a single block of text that fails to establish a coherent, turn-by-turn dialogue structure. It conflates different character voices and narrative threads, demonstrating its inability to follow the structured format without fine-tuning. 60

List of Tables

4.1	Human Evaluation Scores with Standard Deviation for Fine-tuned Llama2, Baichuan, and GPT-3.5.	30
5.1	Comparison of Uniquely Solved Proportion by Category for 7B and 13B Models.	37
6.1	Quantitative Computational and Memory Savings with Shared Attention (SA). FLOPs savings are for the QK^T and Softmax computations. K-Cache savings refer to the Key vectors not stored for shared layers. Example Total Savings are for an 8-layer SA span (7 shared layers) and sequence length $N = 2048$	45
6.2	Performance metrics of various LLM configurations with Shared Attention (SA) applied to different layer ranges.	47
C.1	Performance metrics of various LLM configurations with Cross-Layer Attention (CLA) applied to different layer ranges.	65

List of Abbreviations

NLP	Natural Language Processing
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory
LLM	Large Language Model
GPT	Generative Pre-trained Transformer
FFN	Feed-Forward Network
MHA	Multi-Head Attention
KV	Key-Value (Cache)
SA	Shared Attention
AP-PD	Augmented Prompting with Post-Descriptions
HAC	Hierarchical Attention Cascade
MQA	Multi-Query Attention
GQA	Grouped-Query Attention
CLA	Cross-Layer Attention
SFT	Supervised Fine-Tuning
RLHF	Reinforcement Learning from Human Feedback
ICL	In-Context Learning
CoT	Chain-of-Thought
GLUE	General Language Understanding Evaluation
MMLU	Massive Multitask Language Understanding
GSM8K	Grade School Math 8K
HellaSwag	HellaSwag Benchmark
PIQA	Physical Interaction: Question Answering
SIQA	Social Interaction Question Answering
FLOPs	FLoating Point Operations
RQ	Research Question
STEM	Science, Technology, Engineering, and Mathematics
RPG	Role-Playing Game
CoC	Call of Cthulhu

List of Symbols

A	Attention Score Matrix	Matrix
Q	Query Matrix / Vector	Matrix / Vector
K	Key Matrix / Vector	Matrix / Vector
V	Value Matrix / Vector	Matrix / Vector
O	Output of an Attention Layer	Vector
d_k	Dimension of Key/Query Vectors	Scalar
N	Sequence Length of Input Tokens	Scalar
i, j	Token Position Indices	Scalar
l, m	Layer Indices	Scalar
t	Index of the Top / Guiding Layer	Scalar
L	Total Number of Layers in the Model	Scalar
H	Total Number of Attention Heads	Scalar
H_q	Number of Query Heads	Scalar
H_{kv}	Number of Key/Value Heads	Scalar
S	Span of Layers for Shared Attention	Set
M_t	Attention Pattern Mask from Top Layer t	Matrix
V_{jh}	Variance of Attention Weights for Head h in Layer j	Scalar
D	Set of Excluded Token Positions for Reasoning Algorithm	Set
h	Maximum Layer Index (used for weight decay)	Scalar
c	Constant Factor for Softmax Computation FLOPs	Scalar

For my parents.

Chapter 1

Introduction

1.1 Motivation: The Rise and Challenges of Large Language Models

The past decade has witnessed a paradigm shift in Artificial Intelligence, largely driven by the advent and rapid scaling of Large Language Models (LLMs). Built upon the Transformer architecture (Vaswani et al., 2017), models such as the GPT series (Radford et al., 2019; Brown et al., 2020), Llama (Touvron et al., 2023), Mistral (Jiang et al., 2023), and others (Bai et al., 2023) have demonstrated remarkable capabilities in natural language understanding, generation, and interaction. Their ability to process vast amounts of text data and perform diverse tasks via prompting or fine-tuning has opened up applications ranging from creative writing and code generation to conversational agents and scientific discovery.

Despite these unprecedented successes, the deployment and advancement of LLMs face critical challenges. This thesis focuses on three interconnected domains where limitations persist:

1. **Immersive Dialogue Generation:** Creating dialogue agents, particularly for narrative and gaming contexts (e.g., Role-Playing Games), requires characters to speak not just coherently, but with consistent, believable personas deeply integrated with the evolving context. Existing methods often struggle to maintain persona fidelity and contextual awareness, leading to generic or inconsistent interactions that break immersion (Zhang et al., 2018; Zheng et al., 2020).
2. **Complex Reasoning:** While LLMs excel at information recall and pattern matching, and techniques like Chain-of-Thought (CoT) (Wei et al., 2022) improve multi-step reasoning, their internal processes can be inefficient. Models may struggle with logical depth, and their reasoning path can be hindered by how information is prioritised internally, often failing to perform the necessary ‘computation’ or steps to arrive at correct conclusions for complex problems beyond their training data patterns.
3. **Computational Efficiency:** The scale of LLMs (billions or trillions of parameters) imposes immense computational and memory demands during both training and, critically, inference. The iterative generation process, coupled with the need to store Key-Value (KV) caches for the attention mechanism, creates significant bottlenecks, limiting deployment, increasing costs, and raising environmental concerns (Shazeer, 2019; Dao et al., 2022).

Addressing these challenges requires moving beyond treating LLMs as black boxes and delving into the core mechanism that underpins their function.

1.2 The Central Role of the Attention Mechanism

At the heart of the Transformer architecture lies the self-attention mechanism (Vaswani et al., 2017). It enables the model to weigh the importance of different words (tokens) in the input sequence when producing an output, regardless of their distance from each other. This allows for the capture of long-range dependencies and contextual relationships, which is the primary source of an LLM's power. Through Queries, Keys, and Values, and the use of multiple attention 'heads' across numerous layers, the model learns intricate patterns of information flow.

However, this mechanism is also central to the challenges outlined above:

- How attention flows dictates whether character persona and scene context are correctly prioritised during dialogue generation.
- How attention is distributed across tokens (e.g., focusing on semantic vs. non-semantic tokens) and across reasoning steps impacts the quality of complex reasoning.
- The layer-by-layer, head-by-head computation of attention weights and the storage of Key and Value states constitute the primary computational and memory costs during inference.

Understanding precisely *how* attention weights are distributed within and across layers, *how* these patterns evolve during training, *what* tokens act as information hubs, and *where* redundancy exists, is paramount. This thesis argues that by first *unveiling* the empirical dynamics of the attention mechanism, we can develop principled methods to *harness* these dynamics, directly addressing the limitations in dialogue, reasoning, and efficiency.

1.3 Problem Statement

While the attention mechanism is fundamental, its internal dynamics within modern, large-scale LLMs are not fully understood or optimally leveraged. Current approaches often address dialogue consistency, reasoning ability, and computational efficiency as separate concerns, without a unifying foundation based on observed attention behaviour. The core problems addressed are:

- The lack of methods to visualise, quantify, and interpret the flow, similarity (isotropy), and stability of attention patterns across diverse LLMs and their training lifecycle.
- The difficulty in creating highly consistent, persona-driven dialogue that effectively utilises attention flow (e.g., anchor words) to maintain character and context over multi-turn conversations.
- The inefficiency in attention distribution (e.g., skew towards non-semantic tokens) which can limit the effective 'computational steps' or depth available for an LLM during complex reasoning.
- The computational redundancy inherent in calculating distinct attention weights for every layer, despite empirical evidence suggesting high similarity (isotropy) across significant layer spans in trained models.

This thesis seeks to bridge the gap between the empirical analysis of attention dynamics and the development of practical frameworks and algorithms to enhance LLM capabilities and efficiency.

1.4 Research Questions

To address the stated problems, this thesis investigates the following research questions (RQs):

- **RQ1 (Unveiling):** How can the internal dynamics of attention in LLMs—including information flow (e.g., Hierarchical Attention Cascade, anchor words), cross-layer similarity (isotropy), distribution skew, and evolution during pre-training—be effectively analysed, visualised, and quantified across different architectures and scales?
- **RQ2 (Dialogue):** How can an understanding of attention flow be leveraged, via a structured card-based framework and marker tokens, to fine-tune LLMs for generating immersive, context-aware, and persona-consistent dialogue, and how does information propagate (e.g., via anchor words) during this process?
- **RQ3 (Reasoning):** Can the modification of attention distribution, by mitigating skew and emulating early-layer patterns, extend the effective number of computed tokens (reasoning steps) and enhance an LLM’s performance on complex reasoning tasks without altering the input prompt?
- **RQ4 (Efficiency):** Can the empirically observed phenomenon of attention isotropy be harnessed through a cross-layer weight sharing mechanism (Shared Attention) to reduce computational FLOPs and memory (K-cache) requirements during inference, and what are the performance trade-offs when applied strategically based on layer dynamics analysis?

1.5 Approach: Unveiling and Harnessing Attention Dynamics

This thesis adopts a two-phase approach:

1. **Phase 1: Unveiling.** We conduct extensive empirical analysis using visualisation (word-wise heatmaps) and quantitative metrics (cosine similarity, variance analysis). We analyse attention patterns across diverse, modern LLMs (ranging from 7B to 72B parameters, including Llama, Baichuan, Qwen, Mistral families), across different tasks, and crucially, across different stages of the pre-training lifecycle using model checkpoints. This analysis identifies key phenomena: the Hierarchical Attention Cascade (HAC) and ‘anchor words’ in information flow; attention ‘skew’ towards high-frequency, non-semantic tokens; the emergence and stabilisation of ‘Attention Isotropy’ (high cross-layer similarity) during pre-training; and the functional grouping of model layers based on attention behaviour.
2. **Phase 2: Harnessing.** Informed by the insights from Phase 1, we develop and validate three frameworks/algorithms:
 - A **Card-Based Framework** for dialogue, using marker tokens and prompt engineering to guide attention, with knowledge distilled from larger models (GPT-3.5/4), evaluated by both automated metrics and human assessment, and interpreted using the HAC analysis. (Addresses RQ2).

- An **Attention Optimisation Algorithm** for reasoning, designed to counteract attention skew and emulate fine-grained early-layer patterns in downstream layers, extending the effective token computation path, evaluated on reasoning benchmarks (MMLU, PIQA, SIQA). (Addresses RQ3).
- The **Shared Attention (SA)** framework for efficiency, which reuses computed attention weights across empirically identified isotropic layer spans to reduce computation and memory, evaluated on benchmarks (GLUE, GSM8K, HellaSwag, MMLU) both via direct application and post-fine-tuning. (Addresses RQ4).

Phase 1 provides the foundational understanding for the interventions developed in Phase 2 (addressing RQ1).

1.6 Key Contributions

The primary contributions of this thesis are:

1. **Comprehensive Analysis of Attention Dynamics:** An extensive empirical study unveiling and quantifying key attention phenomena in modern LLMs (7B-72B):
 - Identification and visualisation of the Hierarchical Attention Cascade (HAC) and the role of 'anchor words' in information compression and propagation.
 - Analysis of attention skew by high-frequency, non-semantic tokens impacting information processing.
 - Quantification of Attention Isotropy and the analysis of its emergence and stabilisation dynamics throughout the LLM pre-training process.
2. **Card-Framework for Persona-Driven Dialogue:** Introduction of a novel card-based framework using prompt-engineering, knowledge distillation, and marker tokens to fine-tune 7B LLMs for generating immersive, persona-consistent, context-aware dialogues (e.g., for RPGs), demonstrating capability transfer from larger to smaller models, validated with GPT-4 and human evaluation.
3. **Attention Optimisation for Enhanced Reasoning:** Development and validation of an algorithm that modulates attention patterns (emulating early-layers, reducing skew) to extend token computation, enhancing LLM reasoning capabilities, particularly in non-STEM domains, without requiring prompt changes or additional reasoning-trace data.
4. **Shared Attention (SA) for Efficiency:** The principled validation and application of the Shared Attention framework. By harnessing the empirically observed phenomenon of attention isotropy, SA reuses attention weights across specific layer spans, reducing computational FLOPs and K-cache memory requirements with minimal performance degradation, demonstrating adaptability via fine-tuning.

1.7 Thesis Outline

The remainder of this thesis is structured as follows:

- **Chapter 2:** Provides background on the Transformer architecture, attention mechanisms, LLMs, and reviews related work in LLM interpretability, dialogue systems, reasoning, and efficiency.
- **Chapter 3:** Synthesises the methodologies used for analysing attention dynamics and presents the core empirical findings regarding HAC, anchor words, attention skew, attention isotropy, functional layer grouping, and pre-training evolution.
- **Chapter 4:** Details the Card-based framework for harnessing attention dynamics to generate enhanced persona-driven dialogues, covering data generation, fine-tuning, and evaluation.
- **Chapter 5:** Presents the algorithm for optimising attention distribution to extend token computation and enhance LLM reasoning capabilities.
- **Chapter 6:** Describes the Shared Attention (SA) framework, harnessing attention isotropy to improve computational and memory efficiency during inference.
- **Chapter 7:** Discusses the findings, synthesises the results across chapters, revisits the research questions, and explores the theoretical and practical implications and limitations of the work.
- **Chapter 8:** Concludes the thesis, summarising the contributions and outlining promising directions for future research.

Chapter 2

Background and Literature Review

This chapter establishes the theoretical foundation and reviews the existing literature pertinent to this thesis. We begin by describing the Transformer architecture and the attention mechanism, which are central to Large Language Models. We then discuss the nature of LLMs, their lifecycle, and methods for their interpretation. Subsequently, we review the state-of-the-art in the application areas of dialogue systems, LLM reasoning, and computational efficiency, critically examining prior work. Finally, we identify the key research gaps in the existing literature that this thesis aims to address.

2.1 The Transformer Architecture

The landscape of natural language processing (NLP) was fundamentally altered by the introduction of the Transformer architecture (Vaswani et al., 2017). Prior dominant approaches, such as Recurrent Neural Networks (RNNs) and Long Short-Term Memory networks (LSTMs) (Hochreiter and Schmidhuber, 1997), processed sequences token-by-token. While effective, this sequential nature limited parallelisation and struggled to capture long-range dependencies within the text due to issues like vanishing gradients. The Transformer architecture overcame these limitations by relying entirely on an attention mechanism to draw global dependencies between input and output, eschewing recurrence and convolutions entirely.

2.1.1 Encoder-Decoder and Decoder-Only Models

The original Transformer was designed for sequence-to-sequence tasks, such as machine translation, employing an encoder-decoder structure. The encoder processes the input sequence (e.g., a sentence in the source language) to create a contextual representation, while the decoder generates the output sequence (e.g., the sentence in the target language), attending to both its own previous outputs and the encoder's representation. However, many modern generative LLMs, pioneered by the GPT series (Radford et al., 2019), utilise only the decoder stack. These decoder-only models are trained for autoregressive language modelling – predicting the next token given all preceding tokens. This requires a ‘causal mask’ within the attention mechanism to prevent the model from attending to future tokens during training, ensuring that the prediction for token t_i depends only on tokens $t_1, \dots, t_{i-1}$. This architecture forms the basis for the models analysed and adapted throughout this thesis.

2.1.2 Core Components

Beyond the attention mechanism itself, several components are critical. Input text is first converted into tokens, which are mapped to high-dimensional embedding vectors. Since the attention mechanism is permutation-invariant and has no inherent notion of sequence order, Positional Encodings (Vaswani et al., 2017; Su et al., 2024) are added or applied to the input embeddings to inject information about the relative or absolute position of tokens. Each Transformer block typically consists of a multi-head self-attention layer followed by a position-wise Feed-Forward Network (FFN). Crucially, residual connections (He et al., 2016) are employed around both the attention and FFN layers, followed by Layer Normalisation (Ba, Kiros, and Hinton, 2016). These connections facilitate the training of deep networks by allowing gradients to flow more easily and enabling information to bypass layers, a dynamic relevant to the cross-layer analysis conducted in Chapters 3, 5, and 6.

2.2 The Attention Mechanism

The core innovation of the Transformer is the attention mechanism, allowing the model to dynamically weigh the relevance of different parts of the input sequence when processing any given token.

2.2.1 Scaled Dot-Product Self-Attention

Self-attention operates on three vector representations derived from the input: Queries (Q), Keys (K), and Values (V). Conceptually, a query token ‘asks’ how relevant every key token is to it; the compatibility is computed via a scaled dot-product ($QK^T / \sqrt{d_k}$), where d_k is the dimension. A softmax function converts these compatibility scores into attention weights, representing a probability distribution summing to one. These weights determine how much ‘attention’ the query token pays to each other token in the sequence. The final output for the query token is a weighted sum of the Value vectors, where the weights are the computed attention scores: $\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$. The analysis of the patterns within the $\text{softmax}(\frac{QK^T}{\sqrt{d_k}})$ matrix is a primary focus of the analytical work in Chapter 3. In decoder-only models, a causal mask sets scores for Q attending to future K to $-\infty$ before the softmax, preventing information leakage from subsequent tokens.

2.2.2 Multi-Head Attention (MHA)

Rather than performing a single attention function, Multi-Head Attention (MHA) linearly projects Q, K, and V into multiple lower-dimensional subspaces. The attention function is applied in parallel within each subspace (each ‘head’), allowing the model to jointly attend to information from different representation subspaces at different positions (Vaswani et al., 2017). The outputs from all heads are concatenated and linearly projected to produce the final output. Different heads can learn to specialise in different types of relationships, such as syntactic or semantic dependencies (Clark et al., 2019).

2.2.3 Attention Variants: MQA and GQA

To address the memory bandwidth bottleneck during inference caused by loading distinct Key and Value vectors for every head (the ‘KV cache’), variants have been

proposed. Multi-Query Attention (MQA) (Shazeer, 2019) allows all query heads within a layer to share a single set of Key and Value heads. Grouped-Query Attention (GQA) (Ainslie et al., 2023) provides a compromise, where query heads are divided into groups, and each group shares one set of K and V heads. These methods reduce the KV cache size within a layer, providing context for the efficiency challenges addressed with different strategies in Chapter 6.

2.3 Large Language Models (LLMs)

Large Language Models are Transformer-based models characterised by their vast number of parameters (billions to trillions) and training on massive, internet-scale text corpora.

2.3.1 Scaling Laws

Empirical ‘scaling laws’ (Kaplan et al., 2020; Hoffmann et al., 2022) demonstrate that the performance of LLMs improves predictably as a power-law function of model size, dataset size, and the amount of computation used for training. This provides the motivation for building ever-larger models, but simultaneously exacerbates the computational efficiency challenges discussed in Chapter 6 and throughout this thesis.

2.3.2 The LLM Lifecycle: Training and Adaptation

LLMs typically undergo several stages. *Pre-training* involves unsupervised or self-supervised learning (e.g., next-token prediction) on broad datasets, enabling the model to learn grammar, facts, reasoning abilities, and style. The dynamics of how attention patterns evolve during this phase are analysed in Chapters 3 and 6. Following pre-training, models can be adapted. *Supervised Fine-Tuning* (SFT) adapts the model to specific tasks or domains using labelled data; this technique is employed in Chapter 4 for dialogue generation and in Chapter 6 to adapt models to the Shared Attention mechanism. *Prompting* or In-Context Learning (ICL) (Brown et al., 2020) allows models to perform tasks by conditioning on instructions and examples provided directly in the input context, without parameter updates. Techniques like Chain-of-Thought (CoT) prompting (Wei et al., 2022) leverage this to elicit reasoning, forming the basis for analysis in Chapter 5. Prompt engineering is also central to the data generation in Chapter 4. Finally, *Alignment* techniques, such as Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022), fine-tune the model to be helpful, honest, and harmless, aligning its outputs with human preferences.

2.3.3 Key Model Architectures

This thesis conducts analysis and experiments across several key model families, including decoder-only architectures such as Llama 2 and Llama 3 (Touvron et al., 2023), Mistral (Jiang et al., 2023), Baichuan (Yang et al., 2023), and Qwen (Bai et al., 2023), ranging in scale from 7 billion to over 70 billion parameters, chosen for their open availability and state-of-the-art performance. The GPT series (Brown et al., 2020) is also utilised, primarily for high-quality data generation via knowledge distillation and for evaluation purposes, as detailed in Chapter 4.

2.4 LLM Interpretability and Attention Analysis

Despite their power, LLMs are often considered ‘black boxes’, motivating research into interpretability to understand their internal mechanisms, failure modes, and biases (Belinkov and Glass, 2019). Analysing the attention mechanism is a primary avenue for interpretation. *Visualisation techniques* have been developed to inspect attention patterns for specific inputs, often using heatmaps or bipartite graphs (Vaswani et al., 2018). Tools like BertViz allow interactive exploration of attention head behaviour (Vig, 2019). More recent approaches aim for better visualisation of query-key interactions via joint embeddings (Yeh et al., 2023). These methods inform the visualisation approach adopted in this thesis. *Probing methodologies* involve training simple classifiers on the internal representations (e.g., hidden states, attention weights) of a model to determine if specific linguistic properties, such as part-of-speech or syntactic dependencies, are encoded (Adi et al., 2017; Linzen, Dupoux, and Goldberg, 2016). Studies on attention behaviour have revealed patterns of *layer specialisation*, suggesting that lower layers often focus on local and syntactic features, middle layers on broader context and coreference, and later layers on task-specific information (Clark et al., 2019). They also showed that specific BERT heads consistently attend to delimiter tokens or specific syntactic relations. Recent work has highlighted ‘massive activation’ or attention outliers, where attention becomes highly concentrated on specific tokens (Sun et al., 2024), a phenomenon directly relevant to the reasoning challenges explored in Chapter 5. While some prior work noted similarities in attention patterns across layers (Vig and Belinkov, 2019), a systematic, quantitative analysis of this ‘isotropy’ across modern LLMs and its evolution during training, as presented in Chapters 3 and 6, has been limited.

2.5 Attention Applications and Challenges: Related Work

We now review literature specific to the three application domains of this thesis.

2.5.1 Dialogue Systems and Personalisation

Creating dialogue systems with consistent personality is a long-standing challenge. Early work parametrised personality using psychological models (Mairesse and Walker, 2007). Data-driven approaches learn personas from dialogue data (Li et al., 2016), often using explicit persona inputs (Zhang et al., 2018; Qian et al., 2018). Methods like persona-sparse embeddings (Zheng et al., 2020) encode simple traits (e.g., hobbies, location) but struggle to capture nuanced speaking styles and deep contextual integration required for immersive storytelling. Maintaining consistency has been addressed using methods like natural language inference (Song et al., 2021), but these may not capture stylistic elements. The work in Chapter 4 differs by using rich, card-based descriptions including style, and directly analysing how attention flow (HAC) maintains this context. Furthermore, knowledge distillation, transferring capabilities from large ‘teacher’ models to smaller ‘student’ models by generating synthetic data, has gained prominence (Chiang et al., 2023), often termed ‘instruction tuning’. Personalised distillation adapts teaching to the student model (Chen et al., 2023a). Chapter 4 leverages this concept, using GPT-3.5/4 to generate high-quality, card-based dialogue data for fine-tuning smaller, more efficient 7B models.

2.5.2 LLM Reasoning

LLMs can often answer questions via pattern matching but struggle with tasks requiring multi-step reasoning. Chain-of-Thought (CoT) prompting (Wei et al., 2022) significantly improves performance by encouraging the model, via few-shot examples or zero-shot prompts (e.g., “Let’s think step by step” (Kojima et al., 2022)), to generate intermediate reasoning steps before the final answer. Variants include Self-Consistency (Wang et al., 2023) and methods relying on mined or generated reasoning traces (Zelikman et al., 2022; Lewkowycz et al., 2022). These methods primarily rely on modifying the prompt or data. However, internal attention inefficiencies can hinder reasoning. Work on massive activation (Sun et al., 2024) identifies outlier tokens that attract disproportionate attention, potentially disrupting information flow, and suggests adding learnable weights to balance this. The approach in Chapter 5 addresses this differently, by proposing an algorithm to modify the attention distribution itself, aiming to emulate more effective early-layer patterns and counteract skew, thereby extending the effective token computation for reasoning without altering the prompt. Standard benchmarks for evaluation include MMLU (multi-task knowledge) (Hendrycks et al., 2021), PIQA (physical commonsense) (Bisk et al., 2020), and SIQA (social commonsense) (Sap et al., 2019).

2.5.3 LLM Efficiency

Addressing the computational and memory costs of LLM inference is critical. Many approaches target the KV cache bottleneck. MQA (Shazeer, 2019) and GQA (Ainslie et al., 2023), discussed in Section 2.2.3, share K/V heads within a layer. Cross-Layer Attention (CLA) (Brandon et al., 2024) extends this by sharing K and V projections across adjacent layers. Other methods focus on cache compression or eviction, such as quantization (Hooper et al., 2024), or sparsity and eviction policies (Xiao et al., 2024; Liu et al., 2023). Orthogonal approaches include general model compression (pruning, distillation), efficient kernel implementations like FlashAttention (Dao et al., 2022), and Sparse Attention patterns (Child et al., 2019). Directly related to reducing attention computation, LazyFormer (Ying et al., 2021) explored skipping attention updates temporally and reusing a computed attention matrix across adjacent layers within a ‘lazy block’, based on observations of attention similarity (Vig and Belinkov, 2019). The Shared Attention (SA) framework in Chapter 6 builds on the concept of cross-layer weight reuse but provides a distinct contribution through its extensive empirical foundation: it quantifies attention isotropy and its evolution during pre-training across diverse, modern LLMs (Chapters 3 and 6), using this analysis to provide a principled justification and guide the strategic application of weight sharing, validating it systematically on contemporary architectures. SA specifically targets the redundancy of the $\text{softmax}(QK^T / \sqrt{d_k})$ computation and the associated K-cache, distinct from methods sharing only the K/V inputs like CLA, MQA, and GQA.

2.6 Research Gaps

Based on the review of the literature, several key research gaps are identified:

1. A lack of systematic, quantitative analysis connecting the empirical dynamics of attention (information flow patterns like HAC, cross-layer isotropy, pre-training evolution, distribution skew) across modern, large-scale LLMs to the development of methods for improved applications.

2. The need for dialogue generation frameworks capable of maintaining rich, style-based personas and deep contextual integration for immersive applications, moving beyond simple trait encoding, and providing interpretability of how context is maintained via attention.
3. A scarcity of methods designed to enhance LLM reasoning by directly addressing and correcting internal attention distribution inefficiencies (such as skew caused by outlier tokens), rather than solely relying on external prompt modifications or data augmentation.
4. While attention weight reuse has been proposed, there is a lack of a principled, empirically-grounded approach based on the detailed analysis of attention isotropy and its pre-training dynamics in contemporary LLMs, to guide its strategic application for efficiency with quantifiable trade-offs.

This thesis addresses these gaps by first unveiling key attention dynamics and subsequently harnessing these insights to develop frameworks for enhanced dialogue, reasoning, and efficiency, as detailed in the following chapters.

Chapter 3

Methodologies for Analysing Attention Dynamics

Understanding Large Language Models requires moving beyond input-output behaviour and investigating their internal mechanisms. As established in Chapter 2, the attention mechanism is central to both the capabilities and the challenges of LLMs. This chapter details the analytical methodologies employed throughout this thesis to empirically investigate attention dynamics and synthesises the key observations derived from these analyses. We present methods for both qualitative visualisation and quantitative measurement of attention patterns across different models, tasks, and training stages. The findings regarding information flow, cross-layer similarity, distribution skew, and temporal evolution presented here form the foundational principles—the "Unveiling"—that motivate and justify the frameworks developed to "Harness" these dynamics for enhanced dialogue (Chapter 4), reasoning (Chapter 5), and efficiency (Chapter 6).

3.1 Overview of Analytical Approach

Our analytical approach is grounded in empirical observation. We treat the attention weight matrix, $\text{softmax}(QK^T / \sqrt{d_k})$, computed within each head and layer, as the primary object of study. The methodology combines: (i) qualitative assessment through visualisation to identify patterns of information flow and token focus; (ii) quantitative metrics to measure similarity and variance across layers and models; and (iii) temporal analysis to track the evolution of these patterns during the model pre-training lifecycle. The goal is to identify consistent, structural properties of attention behaviour in modern LLMs, rather than phenomena unique to a single model or dataset.

3.2 LLM Interpretability and Attention Visualization

Despite their power, LLMs are often considered *black boxes*, motivating a significant body of research into interpretability to understand their internal mechanisms, failure modes, and biases (Belinkov and Glass, 2019). Analyzing the attention mechanism is a primary and intuitive avenue for this exploration, but it is part of a broader field of explainable AI techniques.

Broadly, explainable AI methods for LLMs include several categories. Post-hoc techniques such as LIME (Local Interpretable Model-agnostic Explanations) and SHAP (Shapley Additive exPlanations) treat the model as a black box, perturbing inputs to explain individual predictions by identifying critical input features (Ribeiro, Singh,

and Guestrin, 2016; Lundberg and Lee, 2017). Other approaches include gradient-based methods, often called saliency maps, that use output gradients to approximate feature importance (Simonyan, Vedaldi, and Zisserman, 2013). More recently, the field of mechanistic interpretability has emerged with the ambitious goal of reverse-engineering the specific algorithms learned by the model’s components, seeking to identify “circuits” of neurons responsible for particular behaviours (Olsson et al., 2022). This work, however, situates itself within attention-based explanations, which directly inspect the attention weights computed by the model. The underlying assumption is that attention scores correlate with the importance of a token for a given prediction. While some research has questioned whether attention directly constitutes an explanation (Wiegrefe and Pinter, 2019), it undeniably reveals the flow of information within the model and remains a foundational technique for analysis.

Within this domain, visualization is a key methodology. The most fundamental approach is the attention-matrix heatmap, which provides a direct representation of the attention distribution. Building on this, a variety of more sophisticated tools have been developed. For instance, the influential BertViz tool visualizes attention as lines connecting tokens and provides multiple views to inspect individual head patterns and aggregate model behaviour, allowing for the identification of specialized heads (Vig, 2019). More recent methods like AttentionViz aim to overcome the limitation of analyzing single inputs by creating joint embeddings of query and key vectors. This provides a global view of attention patterns across many sequences, helping to identify overarching behaviours of attention heads (Yeh et al., 2023).

The methodology in this thesis builds directly upon the foundation of attention-matrix heatmaps and model-level analysis, similar in spirit to these established tools. However, it makes a distinct contribution by shifting the focus from the qualitative exploration of individual inputs to the quantitative analysis of aggregate, structural properties of attention across layers and throughout the entire pre-training process. While tools like BertViz are excellent for forming hypotheses about what a model is doing on a specific text, our approach aims to move beyond this by developing a rigorous, empirical foundation for model enhancement. Specifically, our methodology differs by:

1. **Quantifying Cross-Layer Similarity (Isotropy).** We introduce quantitative metrics, primarily cosine similarity, to rigorously measure and track the similarity of attention patterns between layers. This allows us to identify the emergence of stable, functional blocks within the model architecture.
2. **Analyzing Pre-training Dynamics.** We explicitly analyze how these quantitative properties, such as isotropy and variance, evolve over the course of pre-training by utilizing model checkpoints. This links the development of attention patterns directly to the model’s learning process and maturation.
3. **Directly Linking Analysis to Application.** Crucially, our methodology is not for interpretation alone. We use the quantitative findings about Attention Isotropy, Attention Skew, and the Hierarchical Attention Cascade (HAC) to directly motivate and design the specific frameworks for efficiency (Chapter 6), reasoning (Chapter 5), and dialogue generation (Chapter 4).

Therefore, this thesis uses established visualization techniques as a starting point but extends them with quantitative and temporal analysis to uncover systemic properties of attention, which are then harnessed for practical model enhancements.

3.3 Data Sources and Model Selection

To ensure the generality of our observations, analysis was conducted across a diverse range of data sources and model architectures.

Models: We analysed decoder-only Transformer models spanning different families and scales, including: Llama 2 (7B, 13B) and Llama 3 (8B, 70B) (Touvron et al., 2023); Mistral 7B (Jiang et al., 2023); Baichuan / Baichuan 2 (7B) (Yang et al., 2023); and Qwen 2 (7B, 72B) (Bai et al., 2023). This selection covers models from 7 billion to 72 billion parameters, incorporating variations in architecture (e.g., use of GQA), pre-training data, and alignment. Both base pre-trained models and instruction-tuned/chat versions were examined. Critically, for temporal analysis, we utilised intermediate pre-training checkpoints of the Baichuan 7B model, ranging from 0.2T to 2.6T tokens processed, allowing observation of how attention patterns develop. We also analysed models specifically fine-tuned on structured, persona-driven dialogue data (Chapter 4).

Data: Attention patterns were extracted and analysed while models processed different types of input:

- Structured, multi-turn dialogue data, generated using the Card-Framework (detailed in Chapter 4), which provided clear, repeated structures (e.g., character names, scene context) ideal for observing information flow and anchor points.
- Questions and contexts from standard benchmarks, primarily the MMLU dataset (Hendrycks et al., 2021), used for consistent cross-model comparison of attention similarity and variance, as well as tasks from PIQA (Bisk et al., 2020), SIQA (Sap et al., 2019), GSM8K (Cobbe et al., 2021), parencite (Wang et al., 2018), and HellaSwag (Zellers et al., 2019).

This combination of models and data allows us to distinguish fundamental attention dynamics from model- or task-specific artefacts.

3.4 Attention Visualisation: Word-wise Heatmaps

Qualitative analysis begins with the visualisation of the attention weight matrix as a heatmap, a foundational technique that forms the basis of many interpretability tools. While powerful interactive tools like BertViz (Vig, 2019) leverage this method for the detailed exploration of individual head patterns on specific inputs, our approach uses the heatmap primarily as a diagnostic tool to identify high-level, recurring structural phenomena. For a given layer and input sequence, the attention matrix A of size $N \times N$ (where N is the sequence length) is extracted. To manage complexity, weights are often averaged across all attention heads within a layer, although head-specific analysis can also be performed. The heatmap visualises this matrix, mapping attention scores to colour intensity. In these visualisations:

- A row i indicates the distribution of attention scores from token i (Query) to all preceding and current tokens $j \leq i$ (Keys).
- A column j indicates how much attention token j (Key) receives from all subsequent and current tokens $i \geq j$ (Queries).

The lower-triangular structure results from the causal mask inherent in decoder-only models. Bright spots or lines within the heatmap indicate high attention scores,

revealing which tokens are critical hubs for information flow. This method is used to identify phenomena such as anchor words, hierarchical cascades, and attention skew, as illustrated in Figure 3.1.

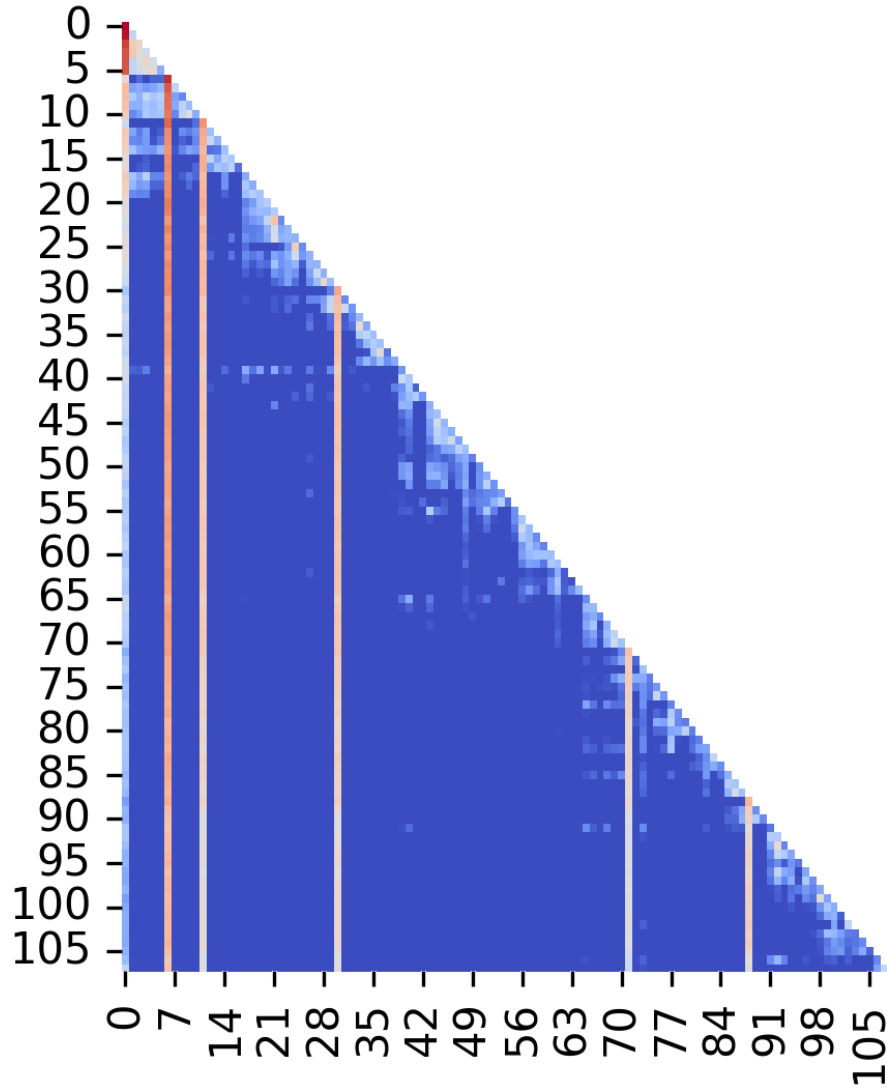


FIGURE 3.1: A sample word-wise attention heatmap from a decoder-only transformer. The x-axis represents the Key tokens and the y-axis the Query tokens. The lower-triangular shape is due to the causal mask. Bright spots highlight high attention scores, revealing tokens that act as critical information hubs.

3.5 Quantifying Attention Patterns

While visualisation provides intuition, quantitative metrics are necessary for rigorous comparison and tracking changes.

3.5.1 Cross-Layer Similarity: Cosine Similarity

To measure the similarity between the attention patterns of different layers, we employ the cosine similarity metric. The attention weight matrix A_l for each layer l is

extracted and flattened into a vector. The cosine similarity between two layers, l and m , is then computed as:

$$\text{similarity}(A_l, A_m) = \frac{A_l \cdot A_m}{\|A_l\| \|A_m\|}$$

Cosine similarity is chosen because it measures the similarity in the orientation of the attention distributions, focusing on the relative pattern of weights rather than their absolute magnitude. It is bounded between 0 and 1, providing an intuitive measure of likeness. By computing this metric for all pairs of layers within a model, we generate a layer-by-layer similarity matrix (visualised as a heatmap). This allows for the quantitative identification of layer groups with similar attention behaviour (isotropy) and tracks how this similarity changes across models or during pre-training. A typical similarity matrix, visualized for the Llama 3 8B model, is shown in Figure 3.2.

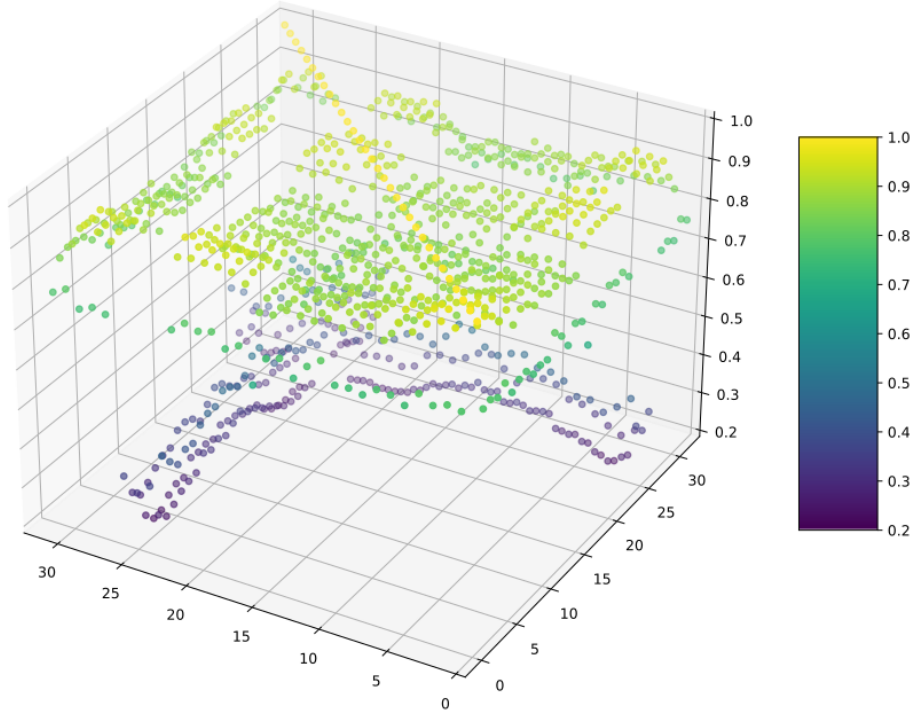


FIGURE 3.2: Layer-wise attention pattern similarity for the Llama 3 8B model, visualized as a 3D scatter plot. The x and y axes represent the model’s layer indices, while the z-axis and color scale (from purple/low to yellow/high) indicate the cosine similarity between the layers. The plot reveals distinct functional blocks: middle-to-late layers form a large, highly similar (isotropic) group, as indicated by the cluster of high-value yellow points.

3.5.2 Attention Variance Analysis

To assess the stability and focus of attention within and across layers, we analyse the variance of attention weights. High variance in early layers might indicate dynamic

adaptation to diverse inputs, while low variance in later layers might suggest stabilisation. As detailed in Chapter 6, we compute variance metrics, including a weighted cumulative variance, to quantify how the ‘spread’ of attention changes through the network depth. This metric helps identify layers that are more or less stable, providing guidance on where architectural modifications might be least disruptive.

3.6 Key Observed Dynamics

Application of the visualisation and quantification methods across the selected models and data reveals several consistent and fundamental dynamics of attention in LLMs.

3.6.1 Information Flow: Hierarchical Attention Cascade (HAC) and Anchor Words

Visualisation of attention heatmaps, particularly during structured dialogue generation (as explored in Chapter 4), reveals a distinct, ladder-shaped pattern of bright intersecting lines within the generated sequence. We term this the *Hierarchical Attention Cascade (HAC)*. The tokens at these bright intersections, which we define as ‘Anchor Words’, act as pivotal nodes. As conceptually illustrated in Figure 3.3, they aggregate information from preceding tokens (represented by a bright row in a heatmap) and are heavily attended to by subsequent tokens (a bright column). In dialogue, character names frequently function as anchor words, compressing the context of previous turns. The HAC demonstrates a recency focus at both the token and turn level, indicating how the model compresses and propagates contextual information through a sequence. This observation informs how context and persona can be maintained in dialogue generation (Chapter 4).

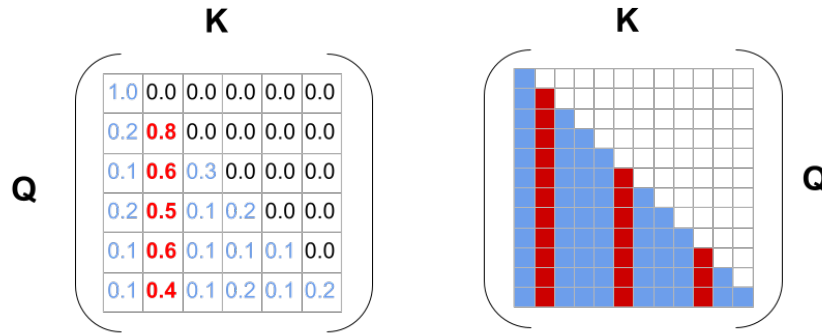


FIGURE 3.3: Conceptual illustration of an Anchor Word. (Left) A simplified attention matrix where the second token receives high attention from subsequent tokens (values in red column) and aggregates information. (Right) The corresponding heatmap visualization, where the red vertical columns indicate the anchor tokens that consistently receiving high attention.

3.6.2 Attention Skew and Non-Semantic Tokens

Analysis across different layers reveals a disparity in attention distribution. While early layers often show fine-grained attention, middle and later layers frequently

exhibit a skewed distribution. A disproportionately large amount of the attention ‘mass’ becomes concentrated on a small number of outlier tokens. These outliers are often high-frequency, non-semantic tokens such as punctuation, articles, or formatting characters, consistent with findings on massive activation (Sun et al., 2024).

This phenomenon, which we term ‘Attention Skew’, is quantified in Figure 3.4. The analysis across several state-of-the-art models shows that certain tokens consistently receive mean attention weights that are orders of magnitude higher than a baseline average. For instance, the ‘NEW_LINE’ token shows attention weights ranging from 3.1x to 6.6x the baseline, highlighting its structural importance to the models. Similarly, common auxiliary verbs (‘BE’), pronouns (‘I’, ‘YOU’), and articles (‘THE’, ‘A’) consistently receive 2-3x more attention than baseline tokens. While there are model-specific variations—with Llama3 showing the most pronounced skew and Qwen2 a more balanced distribution—the overall pattern is clear. This concentration of attention on a few structurally important but often non-semantic tokens can potentially dilute the focus on critical contextual information, hindering the information flow required for deep reasoning. This observation motivates the attention optimisation algorithm designed to counteract this skew (Chapter 5).

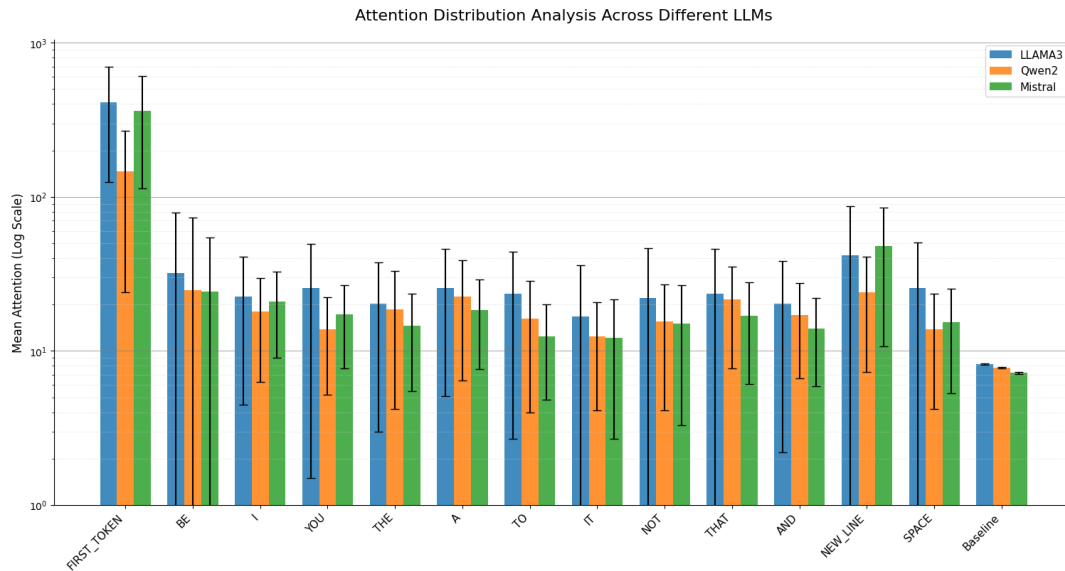


FIGURE 3.4: Quantitative analysis of Attention Skew across Llama3, Qwen2, and Mistral models. The bar chart displays the mean attention received by high-frequency, non-semantic tokens, plotted on a logarithmic scale against a baseline.

3.6.3 Attention Isotropy and Functional Layer Grouping

Quantitative analysis using cosine similarity matrices across all model families studied consistently reveals high similarity in attention weight patterns across large spans of layers. We term this phenomenon *Attention Isotropy*. This is clearly visualized in Figure 3.5, which shows the layer-wise attention similarity matrix for the Llama2-7B model on the MMLU dataset.

The similarity matrix reveals a clear, self-organising block structure, allowing for the identification of distinct functional layer groups: (1) Initial layers (darker blue, low similarity with others); (2) Transitional layers; (3) A large core group of middle-to-late layers exhibiting high internal isotropy (the large, bright red square); and

(4) The final output layer(s) with distinct patterns. The existence of high similarity across many layers suggests computational redundancy, as multiple layers are performing very similar attention weighting functions. This is the core observation harnessed for efficiency gains in Chapter 6.

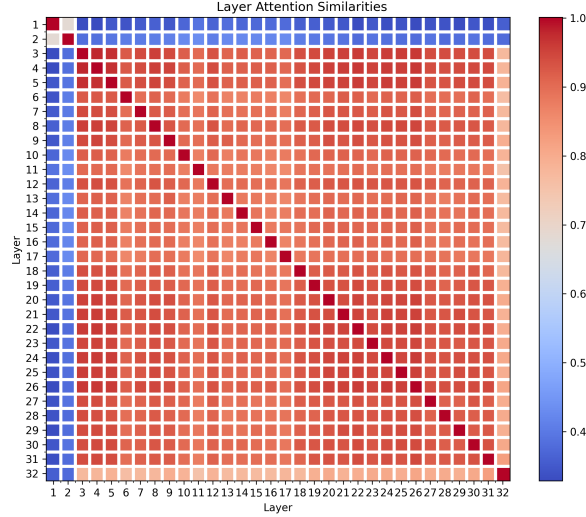


FIGURE 3.5: Layer-wise attention pattern similarity matrix for the Llama2-7B model on the MMLU dataset. The color of each cell at position (i, j) represents the cosine similarity between the attention patterns of layer i and layer j . High similarity (≈ 1.0) is shown in red, and low similarity in blue. The dark red diagonal confirms each layer’s perfect self-similarity. The large, reddish block in the middle-to-late layers provides strong visual evidence for Attention Isotropy.

3.6.4 Evolution and Stabilisation During Pre-training

Temporal analysis using pre-training checkpoints (explored in Chapter 6) demonstrates that attention isotropy and functional grouping are not inherent but *emerge* during the learning process. Early checkpoints (e.g., 0.2T tokens) show less distinct grouping and lower overall similarity within the core layers. As pre-training progresses (e.g., towards 1.0T and 2.6T tokens), the functional groups become more clearly defined, and the degree of isotropy within the core group (Group 3) significantly increases. Figure 6.2 illustrates this evolution for the Baichuan2-7B model. This indicates a progressive stabilisation of the attention mechanism, correlating with the model’s increasing performance on benchmarks (Yang et al., 2023). Attention patterns become more consistent and less variant as the model matures. This dynamic evolution provides a deeper understanding of model learning and strengthens the justification for leveraging the stable, isotropic layers (Chapter 6).

3.7 Summary of Unveiled Attention Principles

This chapter has detailed the methodologies used for analysing attention dynamics and presented four key empirical observations: (1) Information flows via Hierarchical Attention Cascades focused on Anchor Words; (2) Attention distributions can

become skewed towards non-semantic tokens in later layers; (3) Attention patterns exhibit high similarity (Isotropy) across distinct functional layer groups, particularly a large, stable core; and (4) This isotropy and stability emerge and consolidate during the pre-training process. These principles provide a unified, empirical understanding of how attention functions and evolves within modern LLMs. The following chapters demonstrate how each of these observations can be directly harnessed: leveraging HAC and anchor words for dialogue context (Chapter 4), counteracting attention skew for reasoning (Chapter 5), and exploiting attention isotropy and stability for computational efficiency (Chapter 6).

Chapter 4

Harnessing Attention for Enhanced Persona-Driven Dialogue

Building upon the analysis of attention dynamics presented in Chapter 3, this chapter details the first application: harnessing information flow patterns to enhance persona-driven dialogue generation. Creating immersive conversational experiences, particularly in narrative contexts like Role-Playing Games (RPGs), requires dialogue that is not only coherent but deeply consistent with character personas and the evolving scene context. As discussed in Section 2.5.1, maintaining this consistency is a significant challenge. Here, we introduce a card-based framework designed to structure contextual information and guide the model’s attention. We demonstrate how the Hierarchical Attention Cascade (HAC) and anchor words (Section 3.6.1) function as the mechanism by which fine-tuned models maintain context and persona. This chapter describes the framework, the knowledge distillation process used for data generation, the experimental evaluation, and the analysis of how attention dynamics enable the generation of high-quality, immersive dialogue, directly addressing RQ2.

4.1 Introduction: Immersion and Persona in Computational Storytelling

Persona-driven dialogue offers significant potential to enhance player immersion in RPGs by creating unique, character-consistent speaking styles. However, manually crafting such dialogue is resource-intensive for developers, highlighting the need for scalable, automated generation methods (Zheng et al., 2020; Chen et al., 2023b). Existing methods often struggle to capture the nuances of persona and style beyond simple trait tags, resulting in generic or repetitive exchanges, particularly in complex narrative settings. Recent advancements in LLMs offer promise, but require careful structuring and fine-tuning to ensure outputs adhere to specific constraints. We selected the tabletop RPG ‘Call of Cthulhu’ (CoC) as a testbed due to its rich narrative environment, complex character personas, and emphasis on story immersion, providing an ideal ground for exploring context- and persona-constrained dialogue generation. This chapter proposes a framework that structures persona and context information explicitly, allowing us to observe and leverage how the model’s attention mechanism processes and propagates this information to generate consistent dialogue, even for characters and scenarios unseen during training.

4.2 The Card-Based Framework

To provide the model with rich, structured, and modular context, we introduce a card-based framework. This approach decomposes the necessary information into distinct Character Cards and Scene Cards, which are combined within the model’s prompt. This modularity allows characters to be placed in different scenes, enhancing flexibility.

4.2.1 Character Cards

Standard persona definitions (e.g., location, gender, habits (Zhang et al., 2018)) are often insufficient to capture a unique speaking *style*. Our Character Cards, therefore, include standard attributes but extend to detailed descriptions of personality, occupation, skills (relevant to the CoC context), and, crucially, the character’s specific speaking style, often including exemplar phrases. This stylistic information is designed to guide the LLM towards generating dialogue that authentically reflects the character’s unique voice, moving beyond simple factual consistency.

4.2.2 Scene Cards

To control the global conversation flow and context, Scene Cards provide a blueprint for the dialogue setting. Each card details the basic environment, but also incorporates narrative elements such as a key ‘important object’, ‘character goals’, and ‘obstacles’. These elements provide the LLM with a rich contextual background and establish the narrative backbone, orchestrating the flow of the dialogue generation in alignment with the scene’s thematic setting and constraints. The format template, shown in Figure 4.1, ensures consistency in data generation.

```
scene card:
{
  "scenario_name": "<scenario_name>",
  "situation_description": "<situation_description>",
  "location": "<location>",
  "important_object": "<important_object>",
  "character_goal": "<character_goal>",
  "obstacle": "<obstacle>"
}
```

FIGURE 4.1: The format template for a Scene Card. This structured format provides the model with key narrative elements, including location, character goals, and obstacles, to guide the dialogue generation process.

This combined card structure provides the explicit information upon which the model’s attention mechanism must operate to generate appropriate dialogue.

4.3 Data Generation via Knowledge Distillation

A primary challenge is the lack of publicly available datasets matching the specific CoC style and card-based format. Furthermore, we aim to demonstrate capability

transfer to smaller, more efficient models (e.g., 7B parameters), suitable for deployment, rather than relying solely on the largest models (e.g., GPT-4). We therefore employ a knowledge distillation approach (Section 2.5.1), using a large ‘teacher’ model (GPT-3.5) to generate high-quality training data for fine-tuning smaller ‘student’ models (Baichuan-7B, Llama2-7B).

Card Creation: Manually creating thousands of diverse, high-quality cards is impractical. We utilised prompt engineering with GPT-3.5, providing instructions, the format template, and examples to generate a large corpus of diverse scene cards (over 6000) and a core set of character cards, ensuring consistency with the CoC style.

Dialogue Synthesis and AP-PD: Using combinations of two character cards and one scene card, GPT-3.5 was prompted to generate dialogue conversations. Initial tests showed that complex, multi-constraint prompts can yield variable quality. Inspired by Chain-of-Thought principles (Wei et al., 2022) and the efficacy of detailed instructions (Mukherjee et al., 2023), we developed Augmented Prompting with Post-Descriptions (AP-PD). This technique supplements the primary prompt with a post-description rationally emphasising the desired features (persona adherence, context integration) expected in the dialogue. Quantitative evaluation using GPT-4 scoring (criteria: alignment, interaction, flow, engagement, creativity) demonstrated that AP-PD (mean score 8.8, std 0.38) produced statistically higher quality dialogues compared to a conventional prompt (mean 8.5, std 0.53) from 300 samples. The score distributions for both methods are visualised in Figure 4.2. This underscores the importance of prompt structure in guiding the teacher model.

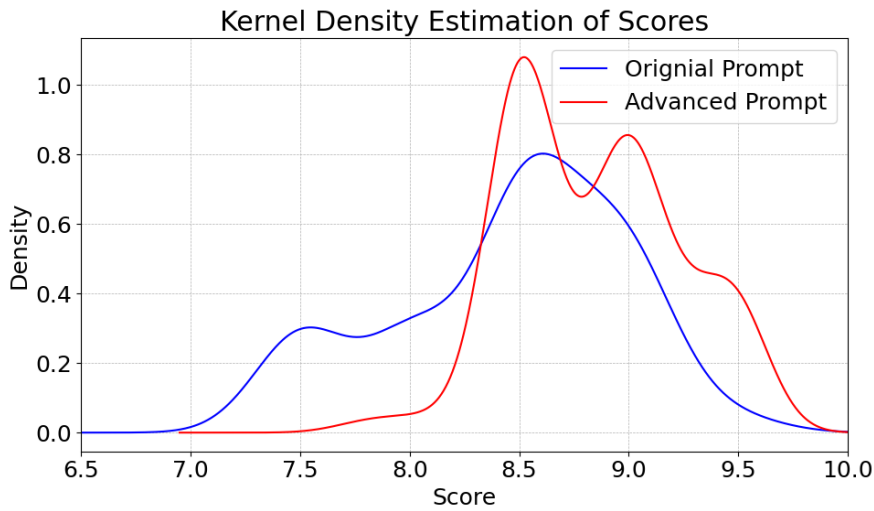


FIGURE 4.2: Kernel Density Estimation (KDE) of dialogue quality scores. The plot compares scores from dialogues generated with a conventional "Original Prompt" (blue) against our Augmented Prompting with Post-Descriptions (AP-PD) method, labelled "Advanced Prompt" (red). The rightward shift and tighter clustering of the red curve visually confirm that the AP-PD method produces higher and more consistent quality scores, aligning with the reported higher mean (8.8 vs. 8.5) and lower standard deviation (0.38 vs. 0.53).

This underscores the importance of prompt structure in guiding the teacher model.

Data Sampling and System Prompt: Recognising that data quality often outweighs quantity (Gunasekar et al., 2023), and to avoid overfitting, a balanced sampling strategy was employed, generating approximately 50,000 dialogues rather than all possible card permutations. For fine-tuning the student models, a carefully constructed system prompt was prepended to the cards and dialogue data. This prompt explicitly directs the model to adhere to the CoC style, utilise information from both Scene and Character Cards, and ensure the dialogue is driven by the defined character personas. The complete structure of the prompt used for data generation is illustrated in Figure 4.3.

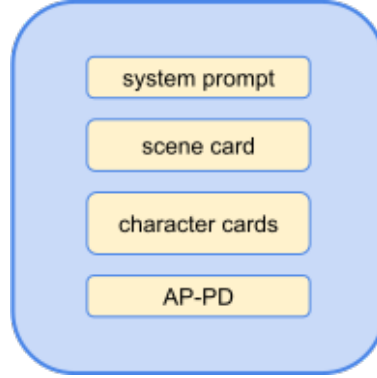


FIGURE 4.3: The modular structure of the complete prompt used for dialogue generation. It combines the system prompt, Scene and Character Cards, and the AP-PD rationale into a single, comprehensive input for the teacher model.

4.4 Leveraging Attention Dynamics

The framework is designed to interact directly with the model’s attention mechanism. We harness attention dynamics in two ways: explicitly guiding focus via marker tokens, and implicitly via the model’s inherent information flow patterns identified in Chapter 3.

4.4.1 Prefix Marker Tokens for Trait Emphasis

To enhance the model’s attention to the specific traits detailed in the cards, we introduce a unique prefix marker token, the @ symbol, before key fields (e.g., speaking style, goals, obstacles), as illustrated in the example Character Card in Figure 4.4.

Analysis confirmed that the tokeniser keeps the @ symbol as an isolated token, distinct from adjacent words. The hypothesis is that this isolated marker acts as a signal, allowing the attention mechanism to more efficiently identify and aggregate the information related to the subsequent trait description during fine-tuning, distinguishing trait information from dialogue content. Analysis of attention patterns confirms this hypothesis. As shown in Figure 4.5, the attention scores for key semantic tokens within the Scene Card show distinct spikes immediately following their associated marker tokens, indicating that the markers successfully guide the model’s focus to compress key contextual information.

@Name:

Professor Arthur

@Occupation:

Linguist

@Gender:

Male

@Age:

52

@Skills:

Anthropology, History, Library Use

@Personality Traits:

Intelligent, Inquisitive, Observant

@Ideals:

Knowledge, Truth, Understanding

@Flaws:

Overly meticulous, Absent-minded

@Speaking Style:

@Description:

Professor Arthur speaks with an educated tone, frequently using complex vocabulary and formal phrasing. He often quotes literary works or historical figures and enjoys explaining the etymology of words

@Quirks:

Using Latin phrases or quotes, Pausing to find the most precise word

@Sample Lines:

Ah, this cryptic inscription reminds me of a line from Virgil: 'Forsan et haec olim meminisse iuvabit.' Perhaps even this will be a source of pleasure someday.

It appears we've stumbled upon a fascinating enigma. Let us endeavor to unravel its secrets with great alacrity.

The etymology of this word is particularly intriguing, harkening back to ancient roots that may offer valuable insight.

FIGURE 4.4: An example Character Card illustrating the use of the '@' prefix marker. Each key attribute, such as '@Occupation' and '@Speaking Style', is prepended with the marker to enhance its salience for the attention mechanism during fine-tuning.

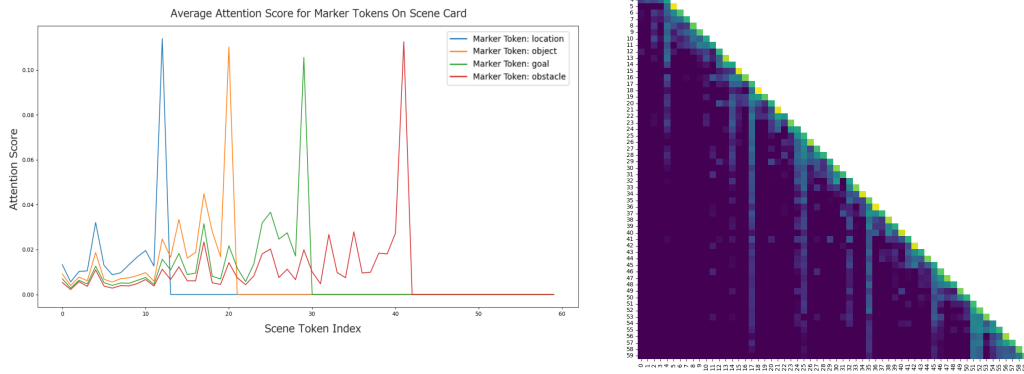


FIGURE 4.5: Attention analysis for marker tokens within a Scene Card. (Left) The line plot shows the average attention scores from the dialogue portion of the prompt back to each token in the Scene Card, revealing sharp peaks for the key semantic tokens ('location', 'object', 'goal', 'obstacle') that immediately follow their markers. (Right) The corresponding attention heatmap visually confirms these patterns, with bright vertical stripes indicating that these marked tokens act as information hubs, consistently attended to throughout the dialogue generation.

4.4.2 Context Maintenance via HAC and Anchor Words

Maintaining persona and context across multiple dialogue turns relies on the model's ability to propagate information. The Hierarchical Attention Cascade (HAC) and Anchor Word phenomena, identified and described in Section 3.6.1, provide the mechanism. Visual analysis of attention heatmaps during dialogue generation, such as the one shown in Figure 4.6, confirms this: character names consistently function as Anchor Words. The name at the start of a new turn attends strongly to the names (anchors) and key content of previous turns, effectively compressing the conversational history and the relevant persona/scene information from the prompt. Subsequent words in the turn then attend strongly to their turn's anchor word (the

character name). This cascade allows context and persona to propagate turn-by-turn, ensuring consistency. The structured format of the cards and the dialogue enables the model to learn this efficient HAC pattern for context maintenance.

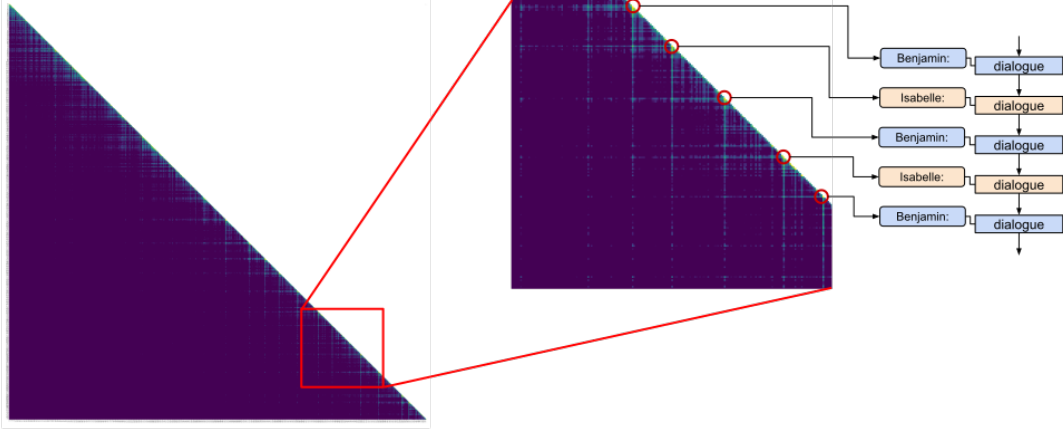


FIGURE 4.6: Visualization of the Hierarchical Attention Cascade (HAC) and anchor words during dialogue generation. The right panel shows a magnified view where character names (e.g., "Benjamin," "Isabelle"), highlighted by red circles, serve as anchor words. These anchors attend to previous turns to gather context. The bright vertical stripes demonstrate that tokens within each new turn strongly attend back to their local anchor (the character's name), effectively propagating conversational context turn-by-turn.

4.5 Experimental Setup

We fine-tuned the memory-efficient Baichuan-7B and Llama2-7B models. **Training:** Fine-tuning used NVIDIA A100 GPUs with the AdamW optimiser, bf16 datatype, a learning rate of $1e-8$, and DeepSpeed Zero Stage 3 for memory efficiency. Training ran for three epochs with a 4096 token window length. To focus learning solely on dialogue generation, the system prompt, scene card, and character card tokens were masked from the loss calculation by assigning a label value of -100. **Evaluation:** Assessing dialogue quality is challenging. We employed two methods to evaluate the fine-tuned models against both the untuned baseline and the teacher model (GPT-3.5):

1. **GPT-4 Based Metric:** To avoid human bias, GPT-4 was prompted to score dialogues based on Contextual Adherence (to scene card), Character Fidelity (to persona and style), Narrative Flow and Consistency, and overall Prompt Conformity. Dialogues for evaluation were generated using novel scene and character cards (created by GPT-4 and Claude-3), unseen during training, to test generalisation (the full evaluation prompt is provided in Appendix B).
2. **Human Evaluation:** A study was conducted via Amazon Mechanical Turk with 60 evaluators, assessing dialogues generated by SFT-Llama2-7B, SFT-Baichuan-7B, and GPT-3.5, based on 14 unseen card pairs. Evaluators scored dialogues anonymously on adherence, fidelity, consistency, and overall quality.

4.6 Results and Analysis

The evaluation demonstrates the effectiveness of the framework and the knowledge distillation process.

Baseline Comparison: Untuned Baichuan-7B struggled to adhere precisely to the card specifics, generating generic dialogue and confirming the necessity of fine-tuning for this constrained task. In contrast, the fine-tuned models showed strong generalisation, generating coherent, engaging, and consistent dialogues even with unseen character and scene cards. For a direct comparison, full generated examples from both the fine-tuned and untuned models are provided in Appendix A (see Figure A.1 and Figure A.2).

GPT-4 Evaluation: Comparative analysis showed that both fine-tuned models, SFT-Baichuan-7B and SFT-Llama2-7B, achieved scores close to, or occasionally surpassing, the teacher model, GPT-3.5, despite having significantly fewer parameters. As illustrated in Figure 4.7, the performance of SFT-Llama2-7B, in particular, aligned closely with that of GPT-3.5.

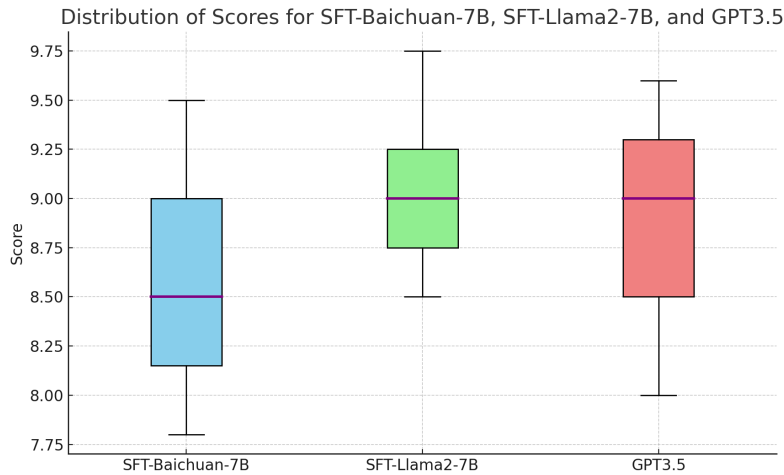


FIGURE 4.7: Boxplot showing the distribution of dialogue quality scores from the GPT-4 evaluation. The plot compares the performance of the two fine-tuned student models (SFT-Baichuan-7B and SFT-Llama2-7B) against the teacher model (GPT-3.5). The median scores (purple lines) and interquartile ranges demonstrate that the smaller, specialized models achieve performance highly competitive with the larger teacher model.

This validates the feasibility of transferring dialogue generation capabilities from large models to smaller, more efficient ones via structured data, prompt engineering, and fine-tuning.

Human Evaluation: Human evaluators confirmed the high quality. The results, summarised in Table 4.1, show that the SFT-Llama2-7B model surpassed the teacher model (GPT-3.5) across all criteria (Adherence, Fidelity, Consistency, Overall). This may be attributable to Llama2’s more extensive pre-training corpus compared to Baichuan-7B, enhancing its base capability for adaptation. The human results strongly support the hypothesis that features from larger LLMs can be effectively transferred and optimised in smaller models for specific domains.

TABLE 4.1: Human Evaluation Scores with Standard Deviation for Fine-tuned Llama2, Baichuan, and GPT-3.5.

Model	Adherence	Fidelity	Consistency	Overall
Llama2	3.80(19)	3.90(26)	3.91(23)	4.05(22)
Baichuan	3.68(17)	3.77(20)	3.76(16)	3.81(13)
GPT-3.5	3.77(24)	3.85(17)	3.80(19)	3.93(17)

4.6.1 Qualitative Case Study and Discussion

To provide concrete evidence behind these quantitative scores, we present a side-by-side comparison of dialogues generated from the same unseen character and scene cards. This qualitative analysis helps to illustrate *why* the fine-tuned model performs so well. Figure 4.8 shows the outputs from our fine-tuned SFT-Llama2-7B model and the original teacher model, GPT-3.5.

These two dialogues were submitted for a blind evaluation to GPT-4. The results were telling: the dialogue from our fine-tuned SFT-Llama2-7B (Dialogue 1) received a score of **8.5/10**, while the dialogue from the larger teacher model, GPT-3.5 (Dialogue 2), received a lower score of **7.5/10**.

The evaluator’s discussion highlights the reasons for this difference. The SFT-Llama2-7B dialogue was praised for its "dynamic" nature and for creatively incorporating specific character traits. For example, Captain Williams’ speech reflects his defined persona, using authoritative language and subtle nautical references. Clara’s dialogue likewise aligns with her role as a medic, showing empathy and observational skill. In contrast, while the GPT-3.5 dialogue was coherent, it was rated as "slightly less dynamic" and less creative.

This case study is significant because it provides a clear, qualitative explanation for the quantitative results seen in Figure 4.7 and Table 4.1. It demonstrates that a smaller, specialized model, when fine-tuned on high-quality, structured data, can learn to capture stylistic nuances more effectively for a specific domain than a larger, general-purpose model. The SFT-Llama2-7B model did not just replicate facts; it internalized the speaking style defined in the cards, a direct result of the framework guiding its attention to these crucial details during training. This demonstrates a successful transfer of not just knowledge, but of nuanced, stylistic capability.

4.7 Summary and Conclusion

This chapter introduced a card-based framework for generating persona-driven dialogue, addressing RQ2. The results demonstrate high-quality dialogue generation by fine-tuned 7B models, comparable or superior to the much larger teacher model (GPT-3.5), confirming the effectiveness of knowledge distillation with structured data (cards) and carefully engineered prompts (AP-PD, system prompt). Crucially, the analysis connects the performance to the underlying attention dynamics. The framework guides attention, while the observed HAC and anchor words (identified in Chapter 3) explain the mechanism of context and persona propagation. Prefix marker tokens further aid in focusing attention on critical card attributes. This approach has significant implications for computational storytelling and RPG development, enabling the creation of immersive experiences with efficient models. The successful generation of dialogue for unseen characters and scenes highlights the framework’s generalisability. We have shown that understanding and leveraging

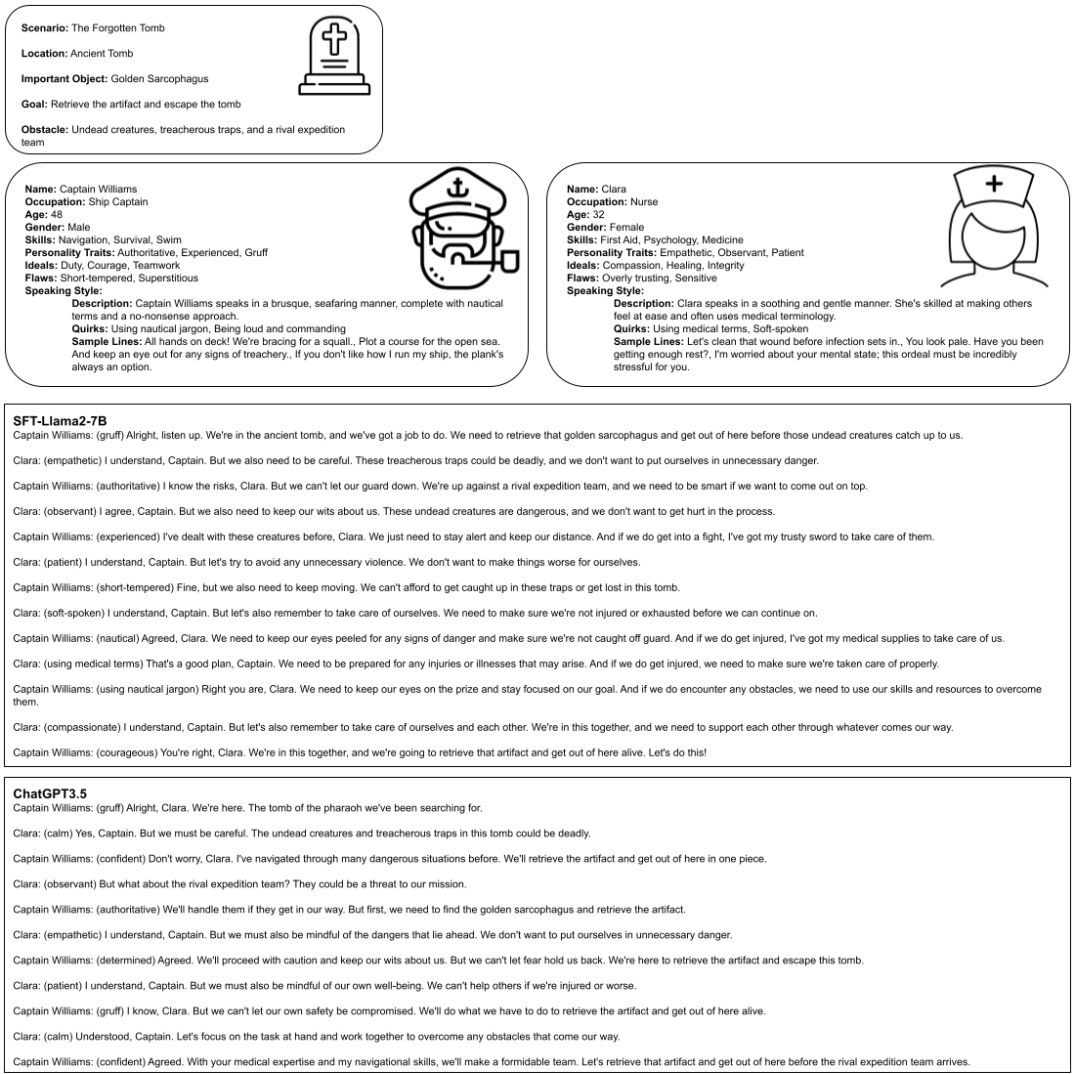


FIGURE 4.8: Side-by-side comparison of generated dialogues from the same unseen prompt (Character Cards: Captain Williams, Clara; Scene Card: An ancient tomb). Dialogue 1 is from the fine-tuned SFT-Llama2-7B, and Dialogue 2 is from the teacher model, GPT-3.5.

attention flow is key to achieving high-fidelity, context-aware dialogue generation. While this chapter focused on harnessing information flow (HAC) for context maintenance, subsequent chapters explore harnessing other observed dynamics: counteracting attention skew for reasoning (Chapter 5) and exploiting attention isotropy for efficiency (Chapter 6).

Chapter 5

Harnessing Attention for Enhanced Reasoning

This chapter addresses the challenge of complex reasoning in Large Language Models, building directly upon the analysis of attention distribution inefficiencies presented in Chapter 3. While techniques like Chain-of-Thought (CoT) prompting improve reasoning by eliciting intermediate steps (Section 2.5.2), the model's internal ability to generate these steps can be hampered by suboptimal attention patterns. We focus specifically on the phenomenon of Attention Skew (Section 3.6.2), where attention becomes disproportionately concentrated on non-semantic tokens in middle and later layers, potentially diluting focus on critical information. This chapter introduces an algorithm designed to counteract this skew by harnessing the more fine-grained attention patterns observed in early layers. We demonstrate that this internal modification extends the effective token computation during the reasoning process, leading to performance improvements without altering the external prompt, directly addressing RQ3.

5.1 Introduction: The Challenge of LLM Reasoning

Large Language Models often excel at knowledge recall but struggle with tasks requiring deep, multi-step reasoning (Wei et al., 2022). Chain-of-Thought prompting has proven effective by encouraging the model to generate a sequence of intermediate tokens representing reasoning steps before outputting a final answer. The underlying principle is that allocating more 'computation'—in the form of generating more tokens in a logical sequence—improves the likelihood of arriving at the correct conclusion for complex problems. However, most CoT research focuses on prompt engineering or data augmentation (Section 2.5.2). We investigate an alternative: enhancing reasoning by modifying the internal attention dynamics that govern *how* the model generates the reasoning chain, based on the hypothesis that inefficient attention allocation limits the model's ability to fully elaborate its reasoning.

5.2 Identifying Attention Inefficiencies in Reasoning

The analytical findings presented in Chapter 3 reveal key inefficiencies relevant to the reasoning process. Our analysis, consistent with observations on massive activation (Sun et al., 2024), confirms the presence of Attention Skew (Section 3.6.2).

- **Attention Skew:** In many middle and later layers, attention weights become highly concentrated on a few outlier tokens. These are frequently high-frequency, non-semantic words (e.g., "is", "the") or formatting characters (e.g., newline).

This disproportionate focus absorbs a large fraction of the attention probability mass, potentially diverting the model’s focus away from the semantically critical tokens required to construct or follow a logical argument. This phenomenon is visualized in Figure 5.1.

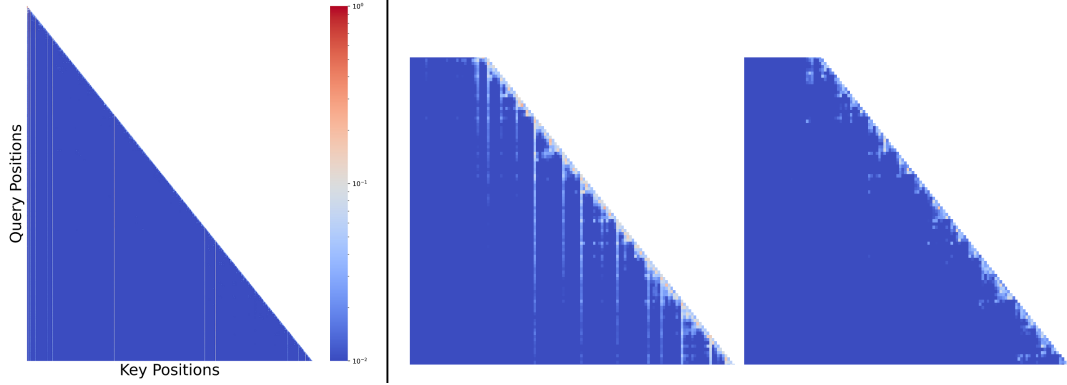


FIGURE 5.1: Visualization of attention skew across different layers of the fine-tuned model. The figure displays three attention heatmaps (left, center, right), where the y-axis represents Query positions and the x-axis Key positions. The prominent vertical stripes in each plot illustrate attention skew: a few Key tokens receive a disproportionately high amount of attention from many subsequent Query tokens. The logarithmic color scale (left panel) emphasizes that while most attention scores are very low (deep blue), the scores for these "skew" tokens are orders of magnitude higher, diverting attention from other potentially more semantic tokens.

- **Early vs. Late Layer Patterns:** A marked contrast exists between early and later layers. Early layers often exhibit more distributed, fine-grained attention, focusing on a wider range of semantically and syntactically relevant preceding tokens. The skew observed in later layers represents a loss of this fine-grained information access, as attention becomes dominated by outliers.

Preliminary experiments involving dropout application suggested that recalibrating these skewed weights could shift focus back towards semantically significant tokens. Furthermore, controlled experiments manipulating attention scores (e.g., zeroing out anchor or non-anchor tokens) highlighted the critical role of recent time-step tokens in knowledge abstraction and the differing roles of early versus middle layers. These observations suggest that the skewed attention patterns in downstream layers may prematurely truncate or misdirect the reasoning process. Enhancing the retention of fine-grained attention, characteristic of early layers, could potentially improve the model’s ability to elaborate its Chain-of-Thought.

5.3 Attention Optimisation Algorithm

To harness the observation that early layers maintain more effective, fine-grained attention patterns compared to the skewed distributions in later layers, we propose an algorithm to counteract attention skew and enhance information flow during inference.

5.3.1 Motivation: Counteracting Skew

The goal is to re-balance the attention distribution in downstream layers, reducing the dominance of non-semantic outliers and encouraging focus on a broader range of semantically relevant tokens. Instead of introducing new learnable parameters (Sun et al., 2024), we aim to guide the downstream layers using the model’s own learned patterns from its early, less skewed layers. The approach does not require any additional training or fine-tuning.

5.3.2 Algorithm Formulation

The algorithm modulates the attention matrix A_l at each downstream layer l , guided by the attention pattern mask M_t derived from a designated top (early) layer. We amplify the attention scores in layer l based on the top layer’s pattern, rather than simply copying them, to avoid excessive homogenisation. A weight decay factor is included to prevent drastic disruption of local dynamics, acknowledging the functional specialisation of different layers (Section 3.6.3). The update rule for the attention score $A_l(i, j)$ between token i and token j at layer l is defined as:

$$A_l(i, j) = A_l(i, j) + A_l(i, j) \cdot \left(1 - \frac{l}{h}\right) \cdot M_t(i, j), \quad \forall i, j \notin D$$

where h is the maximum layer index, and $(1 - l/h)$ provides the linear decay, giving less influence to the top layer’s pattern as depth increases. Crucially, D represents a set of excluded positions, such as the most recent time-steps, to preserve the critical role of local context and recent tokens in knowledge abstraction, as suggested by our preliminary tests. This algorithm, applied during inference to refocus attention, is visually represented in Figure 5.2.

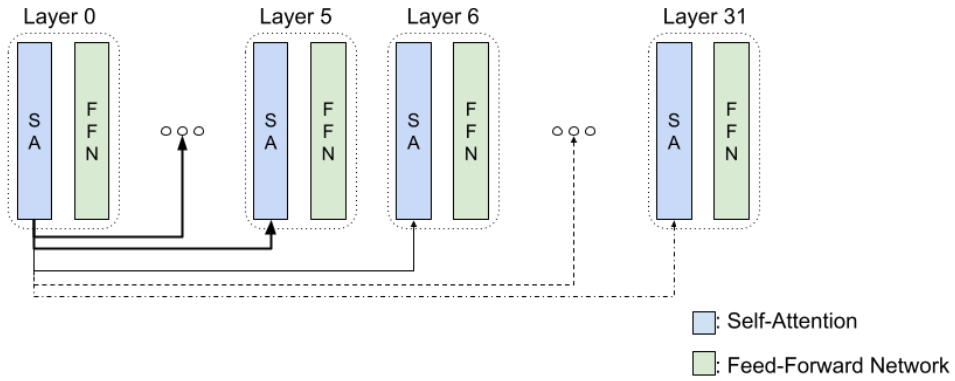


FIGURE 5.2: Architectural diagram of the proposed attention modulation algorithm. The self-attention pattern from a designated top layer is used as a guiding mask to modulate the attention mechanisms in subsequent middle layers during inference. This process refocuses downstream attention based on the global patterns established in early layers.

5.3.3 Mechanism: Extending Token Computation

The intended mechanism of this algorithm is to extend the effective "computation" performed by the model during zero-shot CoT. By re-balancing attention away from

non-semantic tokens and towards a richer set of contextual cues, the algorithm encourages the model to generate a longer, more elaborate sequence of intermediate tokens representing the reasoning chain, before converging on a final answer. We hypothesise that, analogous to how explicit CoT prompting improves reasoning, this algorithmically-induced extension of the token computation path enhances the model’s ability to solve complex problems.

5.4 Experimental Setup

We evaluate the algorithm’s impact on reasoning capability without additional training.

5.4.1 Models and Benchmarks

The algorithm was implemented and tested on the Llama-2 7B and 13B models (Touvron et al., 2023). Performance was measured using three standard benchmarks:

- MMLU (Hendrycks et al., 2021): A broad multi-task benchmark covering diverse academic subjects, allowing separate analysis of STEM (Science, Technology, Engineering, Mathematics) and Non-STEM categories.
- PIQA (Bisk et al., 2020): Measures physical commonsense reasoning.
- SIQA (Sap et al., 2019): Evaluates social commonsense intelligence.

5.4.2 Evaluation Methodology

A zero-temperature setting was used for deterministic evaluation. To rigorously assess reasoning, distinct from mere knowledge recall, we compared three conditions:

1. **Early Answering:** Direct answer generation without CoT.
2. **Zero-shot CoT:** Baseline reasoning using a standard "Let’s think step by step" prompt (Kojima et al., 2022).
3. **Zero-shot CoT + Algorithm:** Zero-shot CoT with our attention optimisation algorithm applied during generation.

LLMs can often answer benchmark questions correctly via Early Answering by recalling patterns from pre-training, rather than performing active reasoning (Wei et al., 2022). To isolate the impact on **reasoning**, we specifically analyse the proportion of questions uniquely solved: those questions that the model failed to answer correctly via Early Answering, but answered correctly using either Zero-shot CoT or Zero-shot CoT + Algorithm. This filtered subset focuses the evaluation on problems that genuinely require multi-step computation. We also measure the average number of tokens generated during the CoT phase to verify the ‘extending token computation’ mechanism.

5.5 Results and Analysis

The experiments validate the algorithm’s mechanism and its impact on reasoning performance.

5.5.1 Impact on Token Length

Analysis of the average number of tokens generated to solve each question confirms the hypothesised mechanism. Across benchmarks, the Zero-shot CoT + Algorithm condition consistently generated longer token sequences compared to the baseline Zero-shot CoT, as visualized in Figure 5.3. This demonstrates that the modification of attention patterns successfully extends the model’s computational path during reasoning.

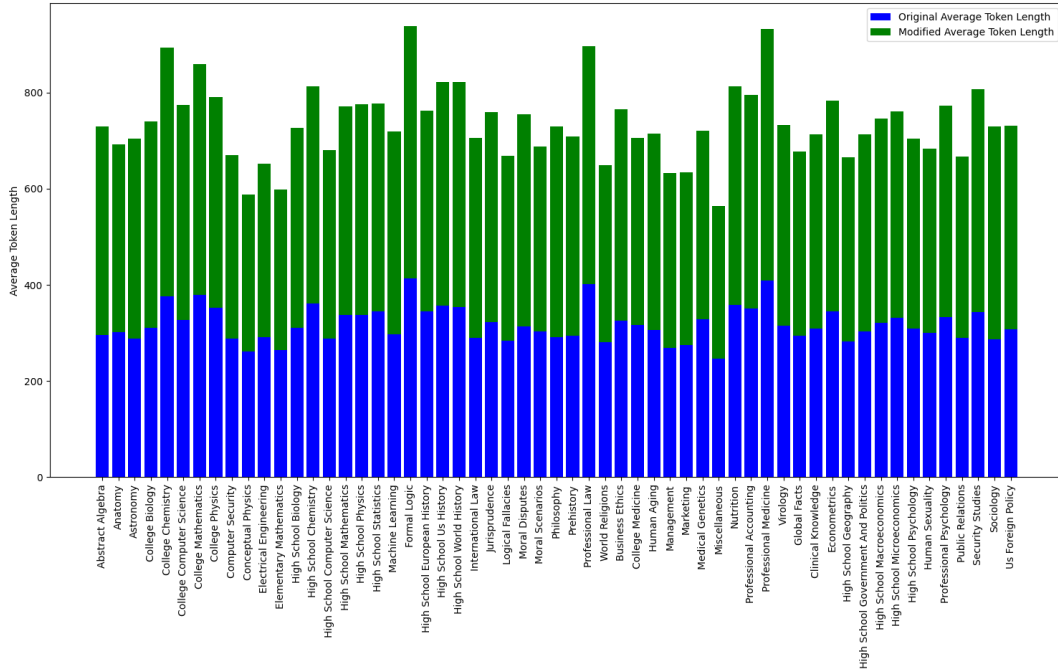


FIGURE 5.3: Comparison of the average token length per response across all MMLU subjects. The blue bars represent the "Original" baseline (Zero-shot CoT), while the green bars represent the "Modified" method using our attention modulation algorithm. The consistent increase in token length (green bars are taller than blue bars) for every subject provides strong evidence that the algorithm encourages a longer, more detailed reasoning process.

5.5.2 Reasoning Performance

Table 5.1 summarises the performance on the uniquely solved questions.

TABLE 5.1: Comparison of Uniquely Solved Proportion by Category for 7B and 13B Models.

Model	STEM	Non-STEM	PIQA	SIQA
13B w/ Ours	12.39	12.62	52.01	30.77
13B w/o	13.77	10.70	47.66	28.94
7B w/ Ours	11.54	11.12	51.80	31.87
7B w/o	13.40	11.37	47.39	29.74

The results show distinct trends:

Our hypothesis is that Attention Skew acts as a major impediment to this traversal. By diverting the model’s focus to structurally important but often non-semantic tokens (like newlines or punctuation), it effectively hides the relevant nodes and edges of the problem. This disruption causes the model to lose track of the logical path, resulting in a truncated, incomplete, or incorrect reasoning chain. Our algorithm functions as a re-focusing mechanism; by counteracting skew and emulating the fine-grained patterns of early layers, it forces the model to pay more attention to the semantically critical components of the problem. This improved focus enables a more complete and accurate traversal of the reasoning graph, which naturally results in a longer and more detailed Chain-of-Thought.

This mechanism’s effect is clearly demonstrated in the qualitative case study from the MMLU benchmark, detailed in Figure 5.5.

- **The Baseline Model (Purple Box)** exhibits a failed traversal of the reasoning graph. It incorrectly latches onto a keyword ("cleavage") and asserts that the enzyme must be a Lyase, ignoring the more crucial detail about the "switch" of a nitrogen group. Its reasoning is short and flawed.
- **The Enhanced Model (Red Box)**, with the algorithm applied, demonstrates a successful traversal. It generates a longer, more deliberate reasoning chain. It correctly identifies the core action—"the transfer of a functional group (a nitrogen group) from one molecule to another"—and correctly maps this action to the definition of a "Transferase."

The extended computational path, induced by the algorithm’s re-focusing of attention, provided the necessary steps for the model to access its latent knowledge and overcome its initial error. This is a direct example of how the enhanced reasoning process facilitates **self-correction**.

For non-STEM tasks, which often rely on commonsense or flexible logic, the reasoning graph may have many valid paths, and a more thorough exploration is highly beneficial. For STEM tasks, however, the graph is often more "brittle," relying on a single correct path and precise definitions. In these cases, if the model’s underlying knowledge of a key node is flawed, extending the reasoning chain can lead to **error propagation**, where the initial mistake is compounded. This suggests our method is most effective at unlocking latent reasoning capabilities for problems where exploration and self-correction are viable paths to a solution.

5.6 Conclusion

This chapter demonstrated a method for enhancing LLM reasoning by directly harnessing an observed attention dynamic: Attention Skew (Section 3.6.2), thereby addressing RQ3. The proposed algorithm, by emulating fine-grained early-layer attention patterns in downstream layers, successfully counteracts skew and extends the number of tokens computed during the CoT process. This leads to measurable performance improvements on reasoning tasks, particularly in non-STEM and commonsense domains, without requiring any changes to the prompt or additional training data. The results reveal that LLMs possess latent reasoning capabilities that can be unlocked by correcting internal attention inefficiencies. The limitation observed in STEM domains highlights the interplay between reasoning path length and the model’s underlying knowledge fidelity; longer chains are only beneficial if the model possesses the correct logical rules to follow. Future work could explore combining this approach with methods to improve STEM-specific knowledge

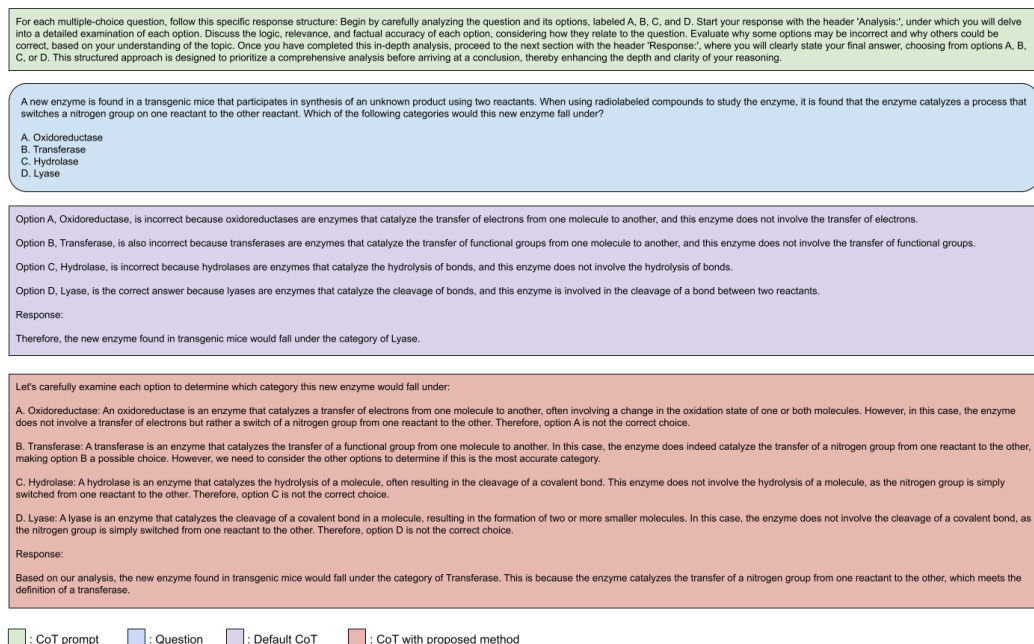


FIGURE 5.5: A qualitative case study from the MMLU benchmark (College Biology) demonstrating the algorithm's impact on reasoning. The figure contrasts the output of the baseline Llama-2 13B model (middle panel, purple) with the output from the same model enhanced by our attention modulation algorithm (bottom panel, red). The baseline model provides a flawed reasoning process and arrives at an incorrect conclusion (Lyase). In contrast, the algorithm-enhanced model generates a more detailed, step-by-step analysis, enabling it to self-correct its initial assessment and identify the correct answer (Transferase). This illustrates the algorithm's ability to improve the model's self-correction capabilities.

or adaptive algorithms that apply the modification selectively. This work provides a proof-of-concept that internal attention manipulation, grounded in empirical analysis (Chapter 3), is a viable path for enhancing model capabilities. While this chapter focused on harnessing attention skew for reasoning, the next chapter explores harnessing a different dynamic—Attention Isotropy—for computational efficiency.

Chapter 6

Harnessing Attention for Enhanced Efficiency

This chapter addresses the critical challenge of computational and memory efficiency in Large Language Models during inference. While Chapters 4 and 5 focused on harnessing attention dynamics (HAC, Skew) to improve model capabilities in dialogue and reasoning, this chapter leverages a different set of observations from Chapter 3—Attention Isotropy and stability—to reduce computational cost. The immense scale of LLMs creates significant bottlenecks (Section 2.5.3). We introduce and validate the Shared Attention (SA) framework, a method grounded directly in the empirical finding that attention patterns across many layers are highly similar (isotropic), implying computational redundancy. By reusing computed attention weights across strategically selected, isotropic layer spans, SA reduces both computational FLOPs and memory requirements. This chapter details the SA framework, its justification based on attention isotropy, experimental validation, and analysis of performance trade-offs, directly addressing RQ4.

6.1 Introduction: Computational and Memory Bottlenecks

The inference phase of LLMs, particularly the autoregressive generation of tokens, is constrained by both computation and memory bandwidth. The attention mechanism (Section 2.2) is a primary contributor. Each layer requires the computation of the attention weight matrix $\text{softmax}(QK^T / \sqrt{d_k})$, and efficient inference relies on caching the Key (K) and Value (V) states for all previous tokens (the KV cache) to avoid redundant re-computation. Methods like MQA, GQA, and CLA (Sections 2.2.3 and 2.5.3) aim to reduce the size of the KV cache by sharing K and V projections. However, the layer-wise computation of the attention weights themselves remains a significant cost. If, as our analysis suggests, the attention patterns computed by different layers are highly similar, then calculating distinct weights for each layer represents a source of computational redundancy that can be targeted for optimisation.

6.2 Foundational Analysis: Attention Isotropy and Stability

The Shared Attention framework is directly motivated and guided by the empirical analysis of attention dynamics detailed in Chapter 3.

- **Attention Isotropy and Layer Grouping:** Quantitative analysis using cosine similarity (Section 3.5.1) across diverse models (7B-72B) consistently revealed

high similarity between the attention weight matrices of different layers, a phenomenon termed Attention Isotropy (Section 3.6.3). This analysis identified distinct functional groups, notably a large core group (Group 3) of middle-to-late layers exhibiting high internal similarity. This characteristic structure is clearly visible in the layer-wise similarity matrix for the Baichuan2-7B model, as shown in Figure 6.1.

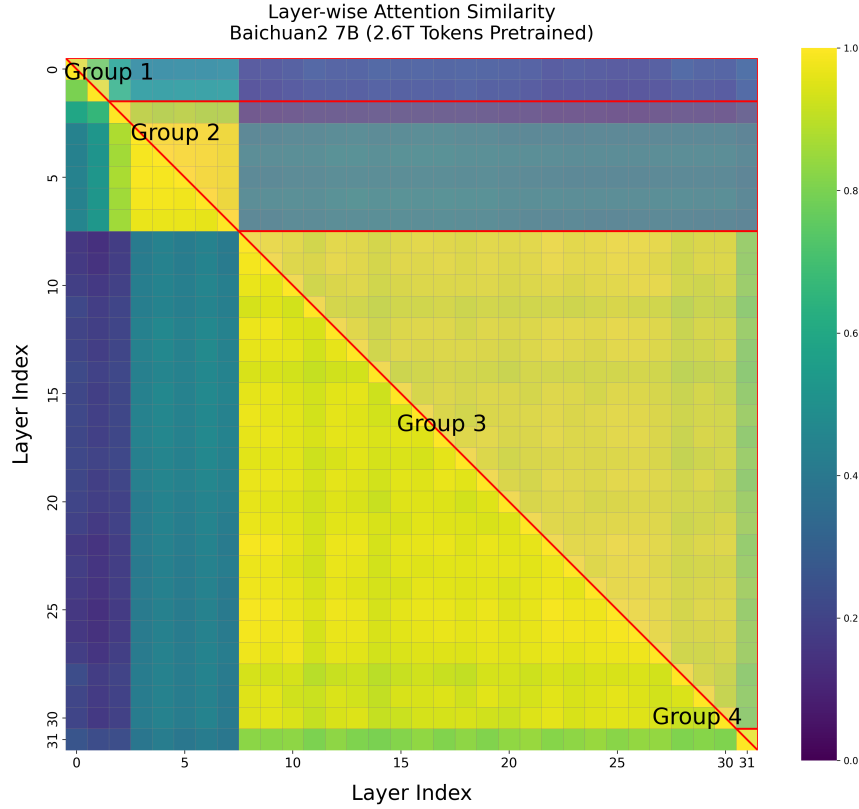


FIGURE 6.1: Layer-wise attention similarity matrix for the Baichuan2-7B model (at 2.6T pre-training tokens). Each cell represents the cosine similarity between two layers (high similarity in yellow, low in purple). The plot reveals distinct functional blocks, annotated as Groups 1-4. Notably, Group 3 forms a large, highly isotropic core of middle-to-late layers that exhibit strong internal similarity. This stable, redundant structure is the basis for our efficiency optimizations.

The high similarity implies that the attention function computed by these layers is largely redundant.

- **Pre-training Dynamics:** Temporal analysis (Section 3.6.4) showed that this isotropy emerges and strengthens as the model progresses through pre-training. The attention patterns within the core group stabilise, indicating that this is a property of mature, well-trained models. This evolutionary process is illustrated in Figure 6.2.
- **Variance Analysis:** To quantitatively assess attention stability, we analyzed the weighted cumulative variance of attention scores. For each starting layer i , this metric aggregates the variances from all subsequent layers, giving more weight to layers that already exhibit higher variance to capture cascading effects. The

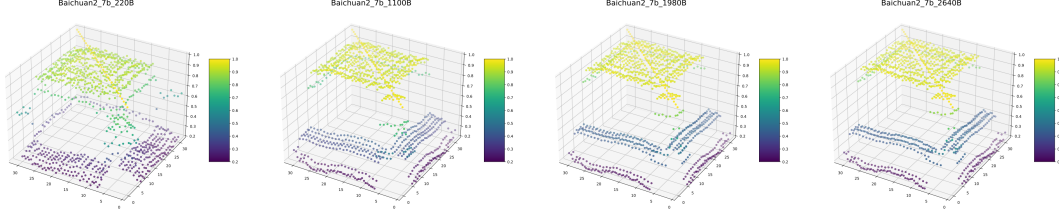


FIGURE 6.2: Evolution of layer-wise attention similarity during the pre-training of the Baichuan2-7B model. The four plots show similarity matrices at different training checkpoints: 220B, 1100B, 1980B, and 2640B tokens (from left to right). This visualization demonstrates the emergence of stable functional groups as training progresses.

weighted cumulative variance for a starting layer i is defined as:

$$\text{WeightedCumulativeVariance}_i = \sum_{j=i}^{L-1} \left(\sum_{h=1}^H (V_{jh} \cdot \text{mean}_h(V_{jh})) \right)$$

where V_{jh} is the variance of attention weights for head h in layer j , L is the total number of layers, and H is the number of heads.

The results of this analysis, shown in Figure 6.3, revealed significantly higher variance in early layers compared to the latter, isotropic layers. Applying modifications (like weight sharing) to high-variance early layers is highly disruptive (leading to perplexity explosion in tests), whereas the stable, isotropic, low-variance latter layers are more robust candidates for such optimisation.

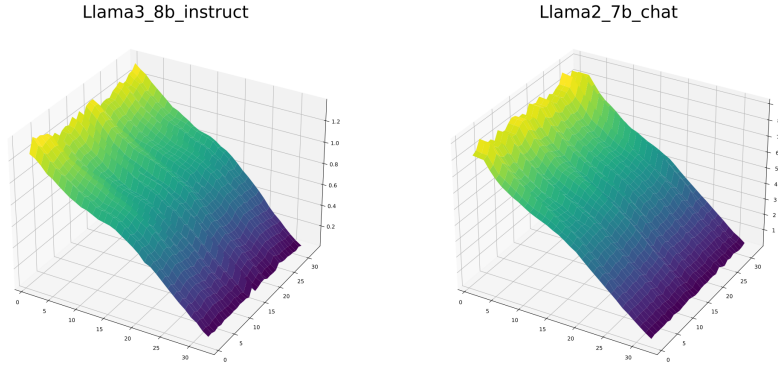


FIGURE 6.3: Weighted cumulative attention variance for Llama3-8B-Instruct (left) and Llama2-7B-Chat (right). The x and y axes represent the starting layer index for the cumulative calculation, while the z-axis shows the calculated variance. The plots clearly show that variance is highest when starting from the earliest layers (high values in the corner) and decreases sharply for later layers. This visualizes the instability of early layers compared to the stable, low-variance nature of the latter, isotropic layers.

Collectively, these findings provide a principled basis: attention weight computation is redundant in specific layer spans (isotropy), these spans are stable properties of trained models (pre-training dynamics), and modifications are best applied to the stable, low-variance regions (variance analysis).

6.3 The Shared Attention (SA) Framework

Based on the evidence of attention isotropy, we validate the Shared Attention (SA) framework, which systematically reuses attention weights across layers.

6.3.1 Algorithm: Cross-Layer Weight Reuse

The SA mechanism identifies a span of layers, S , selected based on the isotropy and variance analysis (typically within the core Group 3, targeting later, stable layers). For this span, the attention weight matrix $A = \text{softmax}(Q_1 K_1^T / \sqrt{d_k})$ is computed only once, for the first layer $l = 1 \in S$. This computed matrix A is then reused for all subsequent layers $l > 1 \in S$, applying it directly to their respective Value vectors: $O_l = A \cdot V_l$. This procedure is formally detailed in Algorithm 1.

Algorithm 1 Shared Attention Algorithm

Input: Set of layers L , input tokens X

Parameters: Attention span S (e.g., layers 23 to 30)

Output: Updated attention weights across layers

- 1: Initialize attention weights $A \leftarrow \emptyset$
 - 2: **for** each layer $l \in S$ **do**
 - 3: **if** first layer in S **then**
 - 4: Compute initial attention weights $A_l \leftarrow \text{softmax}(\frac{Q_l K_l^T}{\sqrt{d_k}})$
 - 5: Set $A \leftarrow A_l$
 - 6: **else**
 - 7: Share attention weights $A_l \leftarrow A$
 - 8: **end if**
 - 9: Apply shared attention to compute outputs $O_l \leftarrow A_l \cdot V_l$
 - 10: **end for**
 - 11: Adjust subsequent layers' inputs using outputs from S
 - 12: Continue processing remaining layers with standard attention
 - 13: **return** Final output after processing all layers
-

The core concept of cross-layer weight reuse shares similarity with the 'lazy block' idea (Ying et al., 2021); however, SA is distinct in its application, being explicitly guided and justified by the extensive quantitative analysis of isotropy, variance, and pre-training dynamics across modern LLMs presented in Chapter 3.

6.3.2 Comparison with K/V Sharing Methods

It is crucial to distinguish SA from methods like MQA, GQA, and CLA (Section 2.5.3). MQA/GQA/CLA share the *inputs* (the K and V projections) either across heads (MQA/GQA) or across layers (CLA), but require the computation $\text{softmax}(QK^T / \sqrt{d_k})$ at every layer. SA, in contrast, shares the *output* of this computation (the attention weight matrix A) across layers. The conceptual difference between these approaches is illustrated in Figure 6.4. SA is therefore complementary; it can potentially be combined with K/V sharing methods.

6.3.3 Computational and Memory Benefits

Sharing the attention matrix A across $|S| - 1$ layers provides two key benefits:

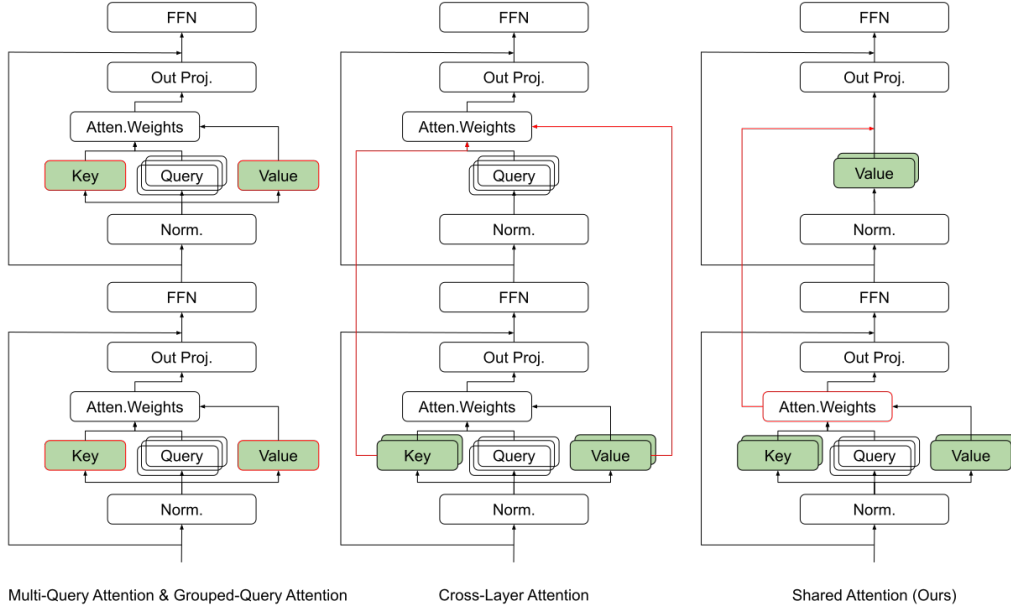


FIGURE 6.4: Conceptual comparison between Shared Attention (SA) and methods like Cross-Layer Attention (CLA).

1. **Computational Savings (FLOPs):** The computationally intensive QK^T dot-product and the subsequent Softmax operation are eliminated for $|S| - 1$ layers within the span. The FLOPs saved per shared layer are approx. $(2 \cdot N \cdot d_k \cdot H_q + c \cdot N \cdot H_q)$ for sequence length N , head dimension d_k , query heads H_q , and softmax constant c .
2. **Memory Savings (K-Cache):** Since the matrix A is computed using only Q_1 and K_1 from the first layer of the span, the Key vectors (K_l) for the subsequent layers $l > 1 \in S$ are not required for the attention weight calculation and need not be stored in the cache. The K-cache memory saved per shared layer is $N \cdot d_k \cdot H_{kv} \cdot \text{precision_bytes}$.

The concrete savings for several popular models are summarized in Table 6.1. These savings directly reduce attention-related bottlenecks during inference.

Model	d_k	H_q	H_{kv}	FLOPs Saved per Shared Layer (for seq. length N)	K-Cache Saved per Shared Layer (for seq. length N)	Example Total Savings (FLOPs / K-Cache)
Mistral-7B	128	32	8	$8352 \cdot N$	$2048 \cdot N$ bytes	≈ 119.7 MFLOPs / ≈ 28.0 MB
Qwen2-7B	128	28	4	$7308 \cdot N$	$1024 \cdot N$ bytes	≈ 104.8 MFLOPs / ≈ 14.0 MB
Llama3-8B	128	32	8	$8352 \cdot N$	$2048 \cdot N$ bytes	≈ 119.7 MFLOPs / ≈ 28.0 MB

TABLE 6.1: Quantitative Computational and Memory Savings with Shared Attention (SA). FLOPs savings are for the QK^T and Softmax computations. K-Cache savings refer to the Key vectors not stored for shared layers. Example Total Savings are for an 8-layer SA span (7 shared layers) and sequence length $N = 2048$.

6.4 Experimental Setup

We evaluate the performance trade-offs and adaptability of SA.

Models: Analysis (Chapter 3) covered a wide range (Llama2/3, Mistral, Qwen2, Baichuan2, 7B-72B). Experimental validation focused on Llama2-7B, Llama3-8B, Mistral-7B, and Qwen2-7B. **Benchmarks:** Performance was measured on GLUE (General understanding) (Wang et al., 2018), GSM8K (Arithmetic reasoning) (Cobbe et al., 2021), HellaSwag (Commonsense reasoning) (Zellers et al., 2019), and MMLU (Multi-task knowledge) (Hendrycks et al., 2021). **Methodology:** Two scenarios were tested:

1. **Direct Application:** SA was applied to various layer spans of the base, pre-trained models during inference, without any retraining. Guided by the variance analysis (Section 3.5.2), spans were primarily chosen in the latter, more stable half of the model layers, testing different span sizes (e.g., 4 or 8 layers) and positions (e.g., SA:23~30, SA:27~30), as depicted in Figure 6.5.

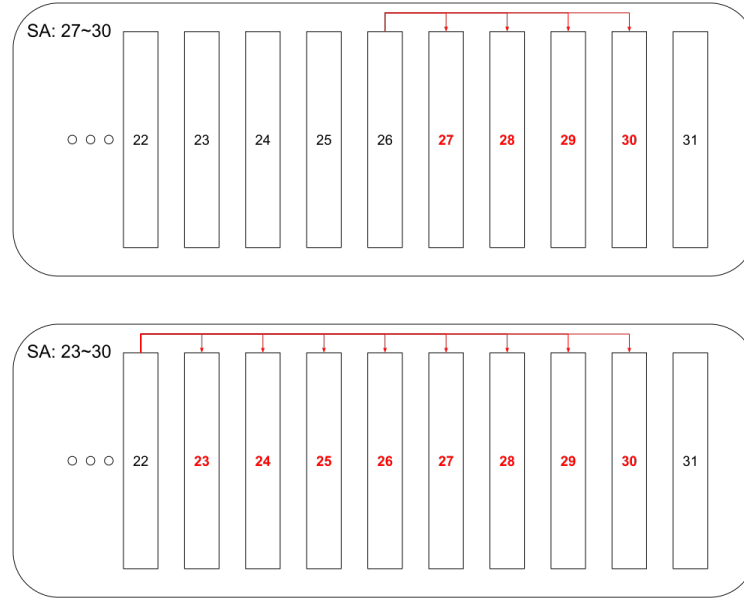


FIGURE 6.5: Conceptual diagram illustrating the application of Shared Attention (SA) to different layer spans. The top panel depicts a shorter 4-layer span (SA:27~30), while the bottom panel shows a longer 8-layer span (SA:23~30).

2. **Fine-tuning Adaptation:** To assess if models can adapt to the SA constraint, base Llama2-7B and Llama3-8B models were fine-tuned on the Alpaca instruction dataset (Taori et al., 2023) with SA active during the fine-tuning process (for span 23~30). Fine-tuning used 2x NVIDIA A100 GPUs, AdamW, $\text{lr}=2 \times 10^{-5}$, bf16, DeepSpeed Zero Stage 3, for 2 epochs. This tests performance recovery compared to direct application.

We also compare SA performance against CLA (see Appendix C for complete results).

6.5 Results and Analysis

The results, summarised in Table 6.2, demonstrate the viability and trade-offs of the SA framework. A natural question arises from this approach: what happens if SA is applied across all layers of the model? Our preliminary experiments confirmed the hypothesis from our foundational analysis: applying SA across all layers, or even

beginning from the highly variable initial layers, is unviable. As predicted by the variance analysis in Section 6.2, which highlighted the instability of early layers, such broad application results in a catastrophic performance collapse, leading to a perplexity explosion where the model’s output becomes incoherent.

Therefore, our experiments were strategically focused on applying SA to the stable, low-variance, and isotropic layer spans identified in our foundational analysis to test the practical limits and trade-offs of the approach within viable configurations.

Model Configuration	GLUE $\uparrow$	GSM8K $\uparrow$	HellaSwag $\uparrow$	MMLU $\uparrow$
<i>Base Models</i>				
Mistral-7B (Baseline)	45.62 $\pm$ 0.19	37.76 $\pm$ 1.34	60.90 $\pm$ 0.49	58.59 $\pm$ 0.39
+ SA:23~30	53.09 $\pm$ 0.19	28.35 $\pm$ 1.24	59.69 $\pm$ 0.49	58.15 $\pm$ 0.39
+ SA:27~30	48.67 $\pm$ 0.19	29.11 $\pm$ 1.25	60.07 $\pm$ 0.49	58.92 $\pm$ 0.39
+ SA:23~26	50.26 $\pm$ 0.19	32.90 $\pm$ 1.29	60.41 $\pm$ 0.49	58.40 $\pm$ 0.39
+ SA:19~22	48.1 $\pm$ 0.19	9.93 $\pm$ 0.82	58.76 $\pm$ 0.49	54.02 $\pm$ 0.40
Qwen2-7B (Baseline)	69.17 $\pm$ 0.17	40.11 $\pm$ 1.35	58.21 $\pm$ 0.49	67.82 $\pm$ 0.37
+ SA:24~26	69.22 $\pm$ 0.17	33.06 $\pm$ 1.30	57.65 $\pm$ 0.49	67.38 $\pm$ 0.37
+ SA:23~26	69.39 $\pm$ 0.17	4.09 $\pm$ 0.55	57.75 $\pm$ 0.49	66.77 $\pm$ 0.37
+ SA:21~26	69.16 $\pm$ 0.18	1.90 $\pm$ 0.38	54.51 $\pm$ 0.50	66.58 $\pm$ 0.37
+ SA:21~23	69.68 $\pm$ 0.17	2.96 $\pm$ 0.47	55.68 $\pm$ 0.50	66.71 $\pm$ 0.37
Llama2-7B (Baseline)	40.50 $\pm$ 0.19	13.95 $\pm$ 0.95	57.13 $\pm$ 0.49	41.19 $\pm$ 0.41
+ SA:23~30	38.19 $\pm$ 0.19	7.28 $\pm$ 0.72	55.75 $\pm$ 0.50	37.94 $\pm$ 0.40
+ SA:27~30	38.82 $\pm$ 0.19	12.43 $\pm$ 0.91	56.16 $\pm$ 0.50	40.56 $\pm$ 0.41
+ SA:23~26	43.51 $\pm$ 0.19	11.22 $\pm$ 0.87	56.81 $\pm$ 0.49	39.94 $\pm$ 0.40
+ SA:19~22	39.96 $\pm$ 0.19	8.34 $\pm$ 0.76	55.53 $\pm$ 0.50	39.26 $\pm$ 0.40
+ SA:15~18	37.31 $\pm$ 0.19	2.20 $\pm$ 0.40	47.90 $\pm$ 0.50	33.78 $\pm$ 0.47
Llama3-8B (Baseline)	48.04 $\pm$ 0.19	51.55 $\pm$ 1.38	60.09 $\pm$ 0.49	61.98 $\pm$ 0.38
+ SA:23~30	55.95 $\pm$ 0.19	32.75 $\pm$ 1.29	60.11 $\pm$ 0.49	61.22 $\pm$ 0.38
+ SA:27~30	55.32 $\pm$ 0.19	45.26 $\pm$ 1.37	60.60 $\pm$ 0.49	61.63 $\pm$ 0.38
+ SA:23~26	50.24 $\pm$ 0.19	45.56 $\pm$ 1.37	59.93 $\pm$ 0.49	61.89 $\pm$ 0.38
+ SA:19~22	51.15 $\pm$ 0.19	37.45 $\pm$ 1.33	58.29 $\pm$ 0.49	61.81 $\pm$ 0.38
+ SA:15~18	46.85 $\pm$ 0.19	1.36 $\pm$ 0.32	53.07 $\pm$ 0.50	30.19 $\pm$ 0.38
<i>Instruct-SFT Models</i>				
Llama2-7B (Baseline)	53.72 $\pm$ 0.19	14.40 $\pm$ 0.97	57.72 $\pm$ 0.49	37.22 $\pm$ 0.40
+ SA:23~30	54.01 $\pm$ 0.19	7.58 $\pm$ 0.73	56.71 $\pm$ 0.49	37.17 $\pm$ 0.40
Llama3-8B (Baseline)	58.97 $\pm$ 0.19	47.76 $\pm$ 1.38	60.13 $\pm$ 0.49	61.21 $\pm$ 0.38
+ SA:23~30	58.26 $\pm$ 0.19	33.21 $\pm$ 1.30	60.63 $\pm$ 0.49	61.32 $\pm$ 0.38

TABLE 6.2: Performance metrics of various LLM configurations with Shared Attention (SA) applied to different layer ranges.

6.5.1 Direct Application

Applying SA directly to pre-trained models without fine-tuning shows:

- Performance on GLUE, HellaSwag, and MMLU remained largely stable or showed only minor degradation when SA was applied to spans in the latter layers (e.g., 23~26, 27~30), consistent with these layers being stable and isotropic. Applying SA to earlier layers (e.g., 15~18) or middle layers with higher variance caused significant performance drops or perplexity explosion, validating the guidance from the variance analysis (Section 3.5.2).

- A notable, consistent performance drop was observed on the GSM8K benchmark across most configurations. This suggests that while the overall attention patterns in isotropic layers are similar, mathematical reasoning may rely on subtle, layer-specific attention variations that are disrupted when a single attention pattern is enforced across the span. This mirrors the sensitivity of STEM tasks observed in Chapter 5.
- The Llama3-8B model, which exhibits higher baseline isotropy, generally showed better tolerance or even minor gains on some tasks (GLUE, HellaSwag for SA:27~30) compared to other models, although the GSM8K drop persisted.

The results confirm that SA can be applied to stable, isotropic layers with minimal impact on many tasks, but task-specific sensitivities exist. Comparison with CLA experiments (Appendix C) indicates SA often maintains comparable or better performance.

6.5.2 Fine-tuning Adaptation

Fine-tuning the models with SA active (span 23~30) on the instruction dataset showed significant performance recovery compared to the direct application results. For both Llama2-7B and Llama3-8B, the fine-tuned SA models achieved performance very close to, or occasionally exceeding, their respective baselines (fine-tuned without SA) on GLUE, HellaSwag and MMLU. While the GSM8K performance still lagged the baseline, the gap was reduced. This demonstrates that models can adapt their parameters during fine-tuning to work effectively within the constraint of shared attention weights, mitigating much of the performance loss seen in direct application.

6.5.3 Efficiency Gains

The application of SA yields concrete efficiency gains, as quantified in Table 6.1. For an 8-layer span (7 shared layers) and a 2048 sequence length, SA saves approximately 100-120 MFLOPs in attention computation and 14-28 MB in K-cache memory across the tested 7B/8B models, directly reducing computational load and memory footprint.

6.6 Conclusion

This chapter validates the Shared Attention (SA) framework as a method for improving LLM inference efficiency, addressing RQ4. The framework is explicitly grounded in the empirical observation of Attention Isotropy and stability (Chapter 3), providing a principled basis for reusing attention weights. We have shown that when applied strategically to stable, isotropic, low-variance layer spans, SA can reduce computational FLOPs and K-cache memory requirements with limited performance impact on many general and knowledge-based tasks. Furthermore, models demonstrate the capacity to adapt to the SA constraint during fine-tuning, recovering performance lost during direct application. The primary limitation observed is the sensitivity of specific tasks, notably mathematical reasoning (GSM8K), which suggests that even within isotropic regions, subtle layer-specific functions may be crucial for certain types of complex, multi-step logic. Future work should explore integrating SA during the pre-training phase, allowing the model to learn with the constraint

from the outset, potentially leading to better adaptation and performance. Combining SA with orthogonal methods such as CLA (for V-cache sharing), quantization, and efficient kernels (e.g., FlashAttention) offers a path to compound efficiency gains. Developing adaptive SA methods that can adjust sharing based on task or input could also mitigate task-specific performance drops. This work confirms that the redundancy implied by attention isotropy can be successfully harnessed for concrete efficiency gains. Having explored the harnessing of HAC for dialogue (Chapter 4), skew for reasoning (Chapter 5), and isotropy for efficiency (this chapter), the next chapter will synthesise these findings and discuss their broader implications.

Chapter 7

Discussion and Synthesis

This thesis embarked on a two-fold mission: first, to unveil the empirical dynamics of the attention mechanism within modern Large Language Models, and second, to harness these insights to develop principled frameworks for enhancing model capabilities in dialogue and reasoning, as well as improving computational efficiency. Chapters 4, 5, and 6 presented specific methodologies, results, and analyses for each application domain, grounded in the analytical findings of Chapter 3. This chapter synthesises these findings, discusses their interconnections, explicitly revisits the research questions posed in Chapter 1, explores the broader theoretical and practical implications of the work, and critically examines its limitations.

7.1 Revisiting the Research Questions

We now evaluate how the work presented addresses the research questions formulated in Section 1.4.

- RQ1 (Unveiling):** *How can the internal dynamics of attention in LLMs—including information flow (e.g., Hierarchical Attention Cascade, anchor words), cross-layer similarity (isotropy), distribution skew, and evolution during pre-training—be effectively analysed, visualised, and quantified across different architectures and scales?*

Answered: Chapter 3 directly addressed this by detailing and applying methodologies for analysis. Word-wise attention heatmaps (Section 3.4) enabled the qualitative identification of information flow, leading to the characterisation of the Hierarchical Attention Cascade (HAC) and Anchor Words (Section 3.6.1), and the observation of Attention Skew towards non-semantic tokens (Section 3.6.2). Cosine similarity matrices (Section 3.5.1) provided the quantitative measure to define and identify Attention Isotropy and functional layer grouping (Section 3.6.3) across diverse models (7B-72B). Variance analysis (Section 3.5.2) quantified layer stability. The use of pre-training checkpoints (Section 3.6.4) allowed the quantification of how isotropy and stability emerge and evolve over time. Together, these methods provide a toolkit for unveiling key attention dynamics.
- RQ2 (Dialogue):** *How can an understanding of attention flow be leveraged, via a structured card-based framework and marker tokens, to fine-tune LLMs for generating immersive, context-aware, and persona-consistent dialogue, and how does information propagate (e.g., via anchor words) during this process?*

Answered: Chapter 4 presented the Card-Based Framework, using knowledge distillation (AP-PD) and fine-tuning to adapt 7B models. The analysis (Sections 4.4, 4.6) showed that prefix marker tokens guide attention to key traits, while the HAC and anchor words (character names) provide the mechanism

for information propagation, compressing and carrying persona and context across dialogue turns. GPT-4 and human evaluations confirmed the generation of high-quality, consistent dialogue, demonstrating that leveraging the observed information flow enables enhanced computational storytelling.

- **RQ3 (Reasoning):** *Can the modification of attention distribution, by mitigating skew and emulating early-layer patterns, extend the effective number of computed tokens (reasoning steps) and enhance an LLM’s performance on complex reasoning tasks without altering the input prompt?*

Answered: Chapter 5 developed an algorithm (Section 5.3) directly targeting the observed Attention Skew (Section 3.6.2). By modulating attention in downstream layers based on early-layer patterns, the results confirmed an increase in the average number of tokens generated during CoT (Section 5.5), effectively extending computation. Evaluation on uniquely-solved questions showed enhanced performance on non-STEM MMLU, PIQA, and SIQA benchmarks, confirming that correcting internal attention inefficiencies can unlock latent reasoning capabilities, although limitations were observed in STEM domains.

- **RQ4 (Efficiency):** *Can the empirically observed phenomenon of attention isotropy be harnessed through a cross-layer weight sharing mechanism (Shared Attention) to reduce computational FLOPs and memory (K-cache) requirements during inference, and what are the performance trade-offs when applied strategically based on layer dynamics analysis?*

Answered: Chapter 6 introduced and validated the Shared Attention (SA) framework (Section 6.3), directly exploiting Attention Isotropy and stability (Sections 3.6.3, 3.6.4, 3.5.2). The analysis quantified reductions in FLOPs and K-cache memory (Section 6.3). Experiments (Section 6.5) demonstrated that applying SA strategically to stable, low-variance, isotropic layer spans maintains performance on many benchmarks (GLUE, HellaSwag, MMLU), while application to early/high-variance layers is detrimental. Trade-offs were identified, notably performance drops on GSM8K, but results also showed that models can adapt to SA via fine-tuning, recovering performance.

All research questions have been systematically addressed through the analysis and frameworks presented.

7.2 Synthesis of Findings on Attention Dynamics

The analyses in Chapter 3 reveal that attention dynamics are multi-faceted; HAC, Skew, and Isotropy are not contradictory but represent different aspects of the mechanism.

- **Interrelation:** Attention Isotropy (Section 3.6.3) indicates that the *overall pattern* or shape of attention distribution is highly similar across many core layers. However, within that pattern, HAC (Section 3.6.1) describes the *specific pathway* of information flow, highlighting the role of anchor tokens as information hubs. Concurrently, Attention Skew (Section 3.6.2) reveals an *inefficiency* within the distribution, where non-semantic tokens can dominate, potentially weakening the effectiveness of the information flow (HAC) or the overall pattern (Isotropy). Isotropy implies redundancy, HAC implies structure, and Skew implies noise or inefficiency.

- **Learned Stability:** Crucially, the pre-training analysis (Section 3.6.4) shows that stability and functional grouping are emergent properties. Models learn to stabilise their attention patterns, developing a core set of isotropic layers. This learned stability is fundamental, providing the justification that modifications like Shared Attention (Chapter 6) can be applied to mature models without catastrophic failure, as the underlying patterns are robust.
- **Functional Specialisation:** The consistent identification of functional layer groups (Section 3.6.3) across different model families and scales reinforces theories of layer specialisation (Clark et al., 2019), but provides a quantitative basis through the similarity and variance analysis. Early layers adapt, core layers process with stable, isotropic patterns, and final layers predict.

This synthesis provides a coherent picture: LLMs develop stable, grouped, yet sometimes inefficient, attention patterns with structured information flow pathways.

7.3 Synthesis of Frameworks and Applications

The common thread linking Chapters 4, 5, and 6 is the principle of ‘Analysis-Guided Design’. Each framework is directly motivated by, and designed to harness, a specific empirical observation from Chapter 3.

- **Grounded Approach:** The Card-Framework leverages HAC for context, the Reasoning algorithm targets Skew, and Shared Attention exploits Isotropy. This contrasts with purely heuristic approaches, providing a more principled path to model enhancement and interpretation.
- **Targeted Intervention:** Different dynamics are targeted for different goals: structure (HAC) for consistency, inefficiency (Skew) for depth, and redundancy (Isotropy) for cost reduction.
- **Task Sensitivity:** A key finding across applications is task sensitivity. Both the reasoning algorithm (Chapter 5) and Shared Attention (Chapter 6) showed performance degradation on highly structured logical tasks (STEM/GSM8K). This suggests that while general patterns (isotropy) or average behaviour (skew) can be identified, certain tasks may rely critically on subtle, layer-specific attention variations or precise information flow that can be disrupted by interventions, either through error propagation (reasoning) or enforced homogeneity (SA).
- **Adaptability:** The thesis demonstrates model adaptability. Knowledge distillation successfully transferred dialogue capability (Chapter 4). Fine-tuning allowed models to adapt and recover performance under the SA constraint (Chapter 6). The reasoning algorithm unlocked latent capability without re-training (Chapter 5). This highlights the flexibility of LLMs to function effectively under modified attention dynamics.

7.4 Theoretical Implications

This work contributes to the theoretical understanding and interpretability of LLMs. The quantification of Attention Isotropy and its evolution provides strong evidence for computational redundancy in over-parametrised models and shows that stability

is an emergent property of learning, not just an architectural artefact. The identification of HAC and anchor words offers a concrete, visualisable model for how long-range context is compressed and propagated through a sequence, layer by layer. The analysis of Attention Skew highlights a fundamental inefficiency that can limit model capability. Collectively, the findings offer interpretability tools (visualisation, similarity/variance metrics) and demonstrate that deep empirical analysis of attention dynamics can demystify the ‘black box’ and reveal the mechanisms underlying model behaviour, success, and failure. The work supports the view that attention patterns are not arbitrary but reflect learned, structured, and stable processes.

7.5 Practical Implications

The frameworks developed have direct practical relevance:

- **Computational Storytelling:** The Card-Framework (Chapter 4) provides a methodology for game developers and narrative designers to create immersive, persona-consistent non-player characters (NPCs) using efficient (e.g., 7B) models, reducing reliance on massive models or extensive manual scripting.
- **Reasoning Tools:** The algorithm in Chapter 5 offers a way to enhance the depth of reasoning for applications in education, analysis, or decision support, potentially making models better ‘thinking partners’ by improving their ability to elaborate, without requiring complex prompt re-engineering for every task.
- **Efficient Deployment:** The Shared Attention framework (Chapter 6) provides a path to reducing the computational cost (FLOPs) and memory footprint (K-cache) of LLM inference. This lowers hardware requirements, reduces latency and energy consumption, and facilitates the deployment of powerful models in resource-constrained environments, such as edge devices or mobile applications, and reduces the cost of cloud-based serving.
- **Model Design & Training:** The understanding of pre-training dynamics and isotropy suggests that efficiency mechanisms like SA could be integrated directly into the pre-training process, potentially leading to models that are efficient by design.

7.6 Limitations of the Study

It is important to acknowledge the limitations of this thesis:

- **Computational Constraints:** The high cost of LLM pre-training prevented experiments integrating the reasoning algorithm or the Shared Attention framework directly into the pre-training phase from scratch. Our analysis of adaptation is therefore limited to fine-tuning (Chapter 6) or inference-time application (Chapter 5).
- **Task-Specific Performance:** As noted, modifications to attention (Chapters 5, 6) negatively impacted performance on STEM and mathematical reasoning (GSM8K). The exact mechanisms by which these tasks rely on precise, layer-specific attention, and how to mitigate this disruption, require further investigation.

- **Architectural Scope:** The analysis and frameworks are developed and tested exclusively on decoder-only Transformer architectures. The generalisability of phenomena like HAC, Isotropy, and Skew, and the applicability of the frameworks, to other architectures (e.g., Encoder-Decoder, State Space Models, Mixture-of-Experts) remains an open question.
- **Analysis Metrics:** While cosine similarity and variance analysis provide valuable insights into isotropy and stability, they are not the only possible metrics. Other measures (e.g., KL-divergence, mutual information) might reveal different facets of attention dynamics.
- **Dialogue Domain:** The Card-Framework (Chapter 4) was developed and tested specifically within the context of the ‘Call of Cthulhu’ RPG style. While generalisation to unseen cards was shown, its application to vastly different dialogue domains (e.g., open-domain chit-chat, task-oriented dialogue) was not evaluated.
- **Efficiency Measurement:** Efficiency gains for SA (Chapter 6) were quantified in terms of FLOPs and cache memory reduction. Practical wall-clock latency and throughput improvements, which depend heavily on hardware, kernel implementation, and batch size, were not systematically benchmarked.
- **Integration:** The three frameworks were developed and evaluated independently. The interplay and potential benefits or conflicts of combining them (e.g., applying SA and the reasoning algorithm simultaneously to the dialogue model) were not explored.

These limitations highlight opportunities for future research, as will be discussed in Chapter 8.

This synthesis connects the empirical analysis with the developed applications, discusses their significance, and provides a critical self-assessment, setting the stage for the final conclusions and future outlook.

Chapter 8

Conclusion and Future Work

This thesis embarked on a two-phase research journey, beginning with a deep empirical investigation into the attention mechanism of LLMs and culminating in the development of practical frameworks that harness these internal dynamics. Having discussed the specific findings and their implications in detail, this final chapter provides a narrative summary of the work, looks forward to promising avenues for future research, and offers a concluding perspective on the value of an analysis-guided approach to building more capable and efficient models.

8.1 Summary of the Thesis

This thesis addressed critical challenges in LLMs concerning persona-driven dialogue, complex reasoning, and computational efficiency. Motivated by the need to move beyond treating LLMs as black boxes, we adopted a two-phase approach: first unveiling the internal dynamics of attention, and subsequently harnessing these empirical insights to develop principled frameworks.

The analytical foundation was established in Chapter 3, which detailed methodologies for identifying and quantifying key phenomena—including the Hierarchical Attention Cascade (HAC), Attention Skew, and Attention Isotropy—across a diverse set of modern LLMs and their pre-training stages. The subsequent chapters demonstrated the ‘harnessing’ of these insights. Chapter 4 leveraged HAC to build a framework for immersive, persona-consistent dialogue. Chapter 5 presented an algorithm to counteract Attention Skew, thereby enhancing complex reasoning. Chapter 6 exploited Attention Isotropy to create the Shared Attention framework, significantly improving computational and memory efficiency. Finally, Chapter 7 provided a comprehensive synthesis of these findings, answering the research questions and discussing the broader implications of the work.

8.2 Future Research Directions

Based on the findings and the limitations identified in this thesis (Section 7.6), several promising avenues for future research emerge:

- **Integration During Pre-training:** Integrating the Shared Attention mechanism, and potentially the attention modulation for reasoning, directly into the pre-training phase. Training models with these constraints from the outset may lead to more native adaptation and better performance.
- **Adaptive and Dynamic Attention Mechanisms:** Developing adaptive versions of the frameworks to address task-specific performance degradation (e.g., on STEM tasks). This could involve task-aware SA that dynamically adjusts

sharing or selectively applying the reasoning algorithm only when complex reasoning is detected.

- **Combination of Frameworks and Orthogonal Methods:** Exploring the interplay of combining the proposed frameworks, and integrating them with other efficiency techniques such as K/V-sharing (GQA), quantization, and sparsity to achieve compound improvements.
- **Application to Other Architectures:** Extending the analytical methodologies to non-Transformer architectures, such as State Space Models (SSMs) or Mixture-of-Experts (MoE) models, to determine if analogous dynamics exist and can be similarly harnessed.
- **Deeper Analysis of Task-Specific Failures:** Conducting focused research into why attention modifications negatively impact performance on tasks like GSM8K to design interventions that preserve critical, layer-specific capabilities.
- **Automation and Tooling:** Developing automated tools for identifying attention dynamics and for recommending optimal layer spans for applying Shared Attention based on isotropy and variance profiles.

8.3 Concluding Remarks

The core tenet of this thesis is that principled advancements in LLMs require a deep, empirical understanding of their internal mechanisms. By systematically unveiling the dynamics of the attention mechanism—its patterns of information flow, its layer-wise similarities, and its potential inefficiencies—we move towards more interpretable and controllable systems.

This work demonstrates that these analytical insights are not merely of theoretical interest, but provide a direct pathway for practical innovation. The frameworks presented—harnessing HAC for consistent dialogue, counteracting skew for deeper reasoning, and exploiting isotropy for greater efficiency—show the power of an ‘Analysis-Guided Design’ philosophy.

As LLMs continue to grow in scale and scope, methods grounded in understanding and harnessing their internal dynamics will be essential for developing artificial intelligence systems that are not only more capable and efficient, but also more interpretable and reliable. This thesis contributes both analytical understanding and practical frameworks towards that goal.

Appendix A

Generated Dialogue Examples

This appendix provides full examples of dialogue generated by the models discussed in Section 4.6. Figure A.1 shows a high-quality conversation produced by our fine-tuned Baichuan-7B model, demonstrating strong adherence to persona and context. In contrast, Figure A.2 shows the output from the untuned baseline model, which lacks structure and consistency.

Dr. Benjamin:	(examining a dusty tome)	My word, this library holds many secrets that could unlock the mysteries of the universe. But we must proceed with caution, for there may be hidden dangers lurking within its pages.
Ms. Isabelle:	(nodding in agreement)	Indeed, Dr. Benjamin. And yet, the knowledge contained within these walls is worth any risk. Perhaps my skills in divination and ritual magic can help us decipher the cryptic prophecies without falling prey to the Library's traps.
Dr. Benjamin:	(impressed)	Your expertise in the occult never ceases to amaze me, Ms. Isabelle. With your guidance, perhaps we can uncover the truth behind these enigmatic texts.
Ms. Isabelle:	(smiling)	Thank you, Dr. Benjamin. But let us not forget the importance of balance. As the saying goes, "Knowledge without wisdom is dangerous."
Dr. Benjamin:	(nodding thoughtfully)	Wise words indeed. Now, let us focus on the task at hand. According to this map, the lost tome should be located in the restricted section of the library.
Ms. Isabelle:	(holding up a card)	Ah, but what if the spirits of the dead guard the entrance? Fear not, dear doctor, for I have brought along some protective talismans. They should ward off any malevolent entities.
Dr. Benjamin:	(raising an eyebrow skeptically)	Protective talismans, you say? How intriguing. Let us hope they prove effective against whatever supernatural forces may lie ahead.
As they make their way through the maze of shelves and corridors, they encounter several puzzling traps and spectral guardians. But with Dr. Benjamin's keen eye for artifacts and Ms. Isabelle's quick thinking, they manage to avoid each obstacle. Finally, they reach the restricted section and find the lost tome.		
Dr. Benjamin:	(flipping through the pages)	Wise words indeed. Now, let us focus on the task at hand. According to this map, the lost tome should be located in the restricted section of the library.
Ms. Isabelle:	(studying the symbols intently)	Yes, yes, I see now. It seems that the key to unlocking the prophecies lies in the very fabric of reality itself. We must perform a series of rituals to restore balance between the worlds.
Dr. Benjamin:	(intrigued)	Fascinating! And how exactly do we go about doing so?
Ms. Isabelle:	(smiling confidently)	That, my friend, is where your expertise comes in. Together, we shall unravel the mysteries of this eldritch enlightenment and emerge victorious over the forces of darkness.

FIGURE A.1: An example dialogue generated by the fine-tuned Baichuan-7B model. The model successfully maintains the personas of "Dr. Benjamin" and "Ms. Isabelle," incorporates actions (e.g., "nodding in agreement"), and progresses the narrative logically according to the scene card's goals and obstacles.

"What do you mean?" (Professor)
(Dr. Eleanor): "Well, there was something strange going on at our campsite last night when all hell broke loose. We were attacked by some kind of monster--something like what they call a deep one down below. But instead of attacking directly, he seemed intent on destroying everything around him first. Our equipment went missing too; not just cameras but also scientific instruments. And then suddenly, out of nowhere came those weird lights over yonder. They looked almost organic somehow, glowing softly against the dark sky. Then afterwards, things got really bad for everyone else who had survived. Somehow their minds started acting strangely, turning them into mindless drones babbling nonsense while trying to attack each other. That poor girl next door didn't make it either. Her body wasn't found until morning. So yeah, whatever happened before has left quite an impression on everybody. Everyone except myself, perhaps. My curiosity never ceases to amaze me sometimes. What did happen exactly? How does such bizarre phenomena occur without any apparent cause? These questions have plagued me ever since yesterday evening. Now let's get moving so we don't miss dinner time again.", "Oh no," said Doctor Eleanor, shivering slightly despite the warm weather outside. "It seems someone saw fit to take advantage of your absence today. Someone took possession of the dagger which belonged to Lady Hester Blackwood. If anyone knows where she might be hiding now, please tell me immediately. Thank you very much indeed.", "You know how people always talk about finding hidden treasure buried underground somewhere? Well, maybe that's true because right beneath our feet lies untold riches waiting to be discovered. Whoever finds it would become rich beyond his wildest dreams. Of course, nobody wants to go digging through dirt unless absolutely necessary. After all, why bother risking life and limb only to end up empty handed? Still, many adventurers believe otherwise. Maybe tomorrow afternoon we ought to try looking for ourselves. Just think of all the fun we could have exploring caves filled with gold coins and jewels galore! Come along dear friend, come join us. There's nothing more exciting than searching for treasures worth millions of dollars! Oh yes, oh yes!", "We need to hurry though. Time is running short. Sooner or later somebody will notice that we aren't actually working together anymore. By then it won't matter anyway. All evidence pointing towards the truth will already be gone forever. No point getting caught redhanded yet still having doubts. Besides, once we reach the surface, we shall surely meet new friends eager to help us solve this mystery. Together we stand strong enough to face anything thrown our way. Right? Yes, definitely. Definitely. Good luck guys. See ya soon."

FIGURE A.2: An example of output generated by the untuned base-line Baichuan-7B model given a similar prompt. The output is a single block of text that fails to establish a coherent, turn-by-turn dialogue structure. It conflates different character voices and narrative threads, demonstrating its inability to follow the structured format without fine-tuning.

Appendix B

GPT-4 Evaluation Prompt

The following is the complete prompt provided to GPT-4 for the automated evaluation of generated dialogues, as mentioned in Section 4.6.

Scene card:

<scene card>

Character 1 card:

<character card>

Character 2 card:

<character card>

Dialogue 1:

<dialogue>

Dialogue 2:

<dialogue>

Evaluation:

The above text contains two Call of Cthulhu (CoC) style dialogues generated by different language models based on the same scene and character cards. Your task is to analyze and measure each of these two dialogues along the following dimensions:

Contextual Adherence:

- Does the dialogue accurately reflect the key elements provided in the scene card, such as the scenario description, location, important object, character goal, and obstacle?
- Are the events, atmosphere, and challenges mentioned in the dialogue consistent with the context set up in the scene card?

Provide a brief justification and rate adherence to the scene card context on a scale of 0-10.

Character Fidelity:

- For each character, evaluate how well the dialogue portrays their specified occupation, age, gender, skills, personality traits, ideals, and flaws.
- Does the dialogue accurately capture each character's described speaking style, including their speech patterns, quirks, and tendency to use certain phrases?

Briefly explain the rationale and rate each character's portrayal on a scale of 0-10.

Narrative Flow and Consistency:

- Does the dialogue flow logically from one statement to the next, creating a coherent conversation?
- Are there any jarring transitions, redundant exchanges, or inconsistencies that detract from the dialogue's clarity and impact?
- Evaluate if the dialogue effectively advances character development and moves the plot forward.

Justifying your assessment and assign a score from 0-10 for the overall flow and consistency.

Prompt Conformity:

- Does the dialogue strictly adhere to the details provided in the scene and character cards without introducing tangential or irrelevant information?
- Analyze if the characters, events, and challenges mentioned in the dialogue are directly tied to elements specified in the original prompt.

Rating how well the dialogue conforms to the prompt on a 0-10 scale.

Evaluation Format:

Dialogue 1:

Contextual Adherence: [Brief explanation]

Rating: [0-10]

Character Fidelity:

Character 1: [Brief explanation]

Character 2: [Brief explanation]

Rating: [0-10]

Narrative Flow and Consistency: [Brief explanation]

Rating: [0-10]

Prompt Conformity: [Brief explanation]

Rating: [0-10]

Total Score: [Sum of all ratings]/40

Overall Score: [Total Score]/4

Dialogue 2:

Contextual Adherence: [Brief explanation]

Rating: [0-10]

Character Fidelity:

Character 1: [Brief explanation]

Character 2: [Brief explanation]

Rating: [0-10]

Narrative Flow and Consistency: [Brief explanation]

Rating: [0-10]

Prompt Conformity: [Brief explanation]
Rating: [0-10]

Total Score: [Sum of all ratings]/40

Overall Score: [Total Score]/4

Appendix C

Additional Experimental Results: Cross-Layer Attention (CLA)

Additional Experiments

Further extending our evaluation, we conducted comparative experiments using the Cross-Layer Attention (CLA) method. The performance metrics from these experiments are detailed in Table C.1. A comparison with the results presented in the main content of this paper, where SA was applied, shows that SA consistently achieves superior performance relative to CLA. This enhancement is evident across multiple benchmarks and highlights the efficacy of SA in optimizing attention mechanisms more effectively than CLA, particularly in reducing computational overhead while maintaining model performance.

Model Configuration	GLUE $\uparrow$	GSM8K $\uparrow$	HellaSwag $\uparrow$	MMLU $\uparrow$
<i>Base Models</i>				
Mistral-7B (Baseline)	45.62 $\pm$ 0.19	37.76 $\pm$ 1.34	60.90 $\pm$ 0.49	58.59 $\pm$ 0.39
+ CLA:23~30	55.38 $\pm$ 0.19	25.32 $\pm$ 1.20	59.50 $\pm$ 0.49	58.87 $\pm$ 0.39
+ CLA:27~30	53.10 $\pm$ 0.19	33.81 $\pm$ 1.30	60.43 $\pm$ 0.49	58.75 $\pm$ 0.39
+ CLA:23~26	48.27 $\pm$ 0.19	32.68 $\pm$ 1.29	60.04 $\pm$ 0.49	58.65 $\pm$ 0.39
+ CLA:19~22	48.93 $\pm$ 0.19	8.57 $\pm$ 0.77	57.04 $\pm$ 0.49	54.47 $\pm$ 0.40
Qwen2-7B (Baseline)	69.17 $\pm$ 0.17	40.11 $\pm$ 1.35	58.21 $\pm$ 0.49	67.82 $\pm$ 0.37
+ CLA:24~26	68.15 $\pm$ 0.17	31.92 $\pm$ 1.28	56.80 $\pm$ 0.49	67.31 $\pm$ 0.37
+ CLA:23~26	64.43 $\pm$ 0.18	2.81 $\pm$ 0.45	55.57 $\pm$ 0.50	66.44 $\pm$ 0.37
+ CLA:21~26	61.48 $\pm$ 0.19	0.00 $\pm$ 0.00	51.72 $\pm$ 0.50	65.34 $\pm$ 0.38
+ CLA:21~23	66.87 $\pm$ 0.18	1.97 $\pm$ 0.38	54.60 $\pm$ 0.50	66.59 $\pm$ 0.37
Llama2-7B (Baseline)	40.50 $\pm$ 0.19	13.95 $\pm$ 0.95	57.13 $\pm$ 0.49	41.19 $\pm$ 0.41
+ CLA:23~30	48.17 $\pm$ 0.19	6.07 $\pm$ 0.66	53.26 $\pm$ 0.50	37.05 $\pm$ 0.40
+ CLA:27~30	39.49 $\pm$ 0.19	10.77 $\pm$ 0.85	56.07 $\pm$ 0.50	38.26 $\pm$ 0.40
+ CLA:23~26	44.70 $\pm$ 0.19	10.69 $\pm$ 0.85	55.54 $\pm$ 0.50	38.05 $\pm$ 0.40
+ CLA:19~22	41.54 $\pm$ 0.19	5.00 $\pm$ 0.60	52.73 $\pm$ 0.50	37.99 $\pm$ 0.40
+ CLA:15~18	38.15 $\pm$ 0.19	1.82 $\pm$ 0.37	45.68 $\pm$ 0.50	32.40 $\pm$ 0.39
Llama3-8B (Baseline)	48.04 $\pm$ 0.19	51.55 $\pm$ 1.38	60.09 $\pm$ 0.49	61.98 $\pm$ 0.38
+ CLA:23~30	54.23 $\pm$ 0.19	26.23 $\pm$ 1.21	56.73 $\pm$ 0.49	61.08 $\pm$ 0.38
+ CLA:27~30	54.32 $\pm$ 0.19	45.87 $\pm$ 1.37	59.25 $\pm$ 0.49	61.02 $\pm$ 0.38
+ CLA:23~26	46.61 $\pm$ 0.19	45.49 $\pm$ 1.37	58.51 $\pm$ 0.49	61.94 $\pm$ 0.38
+ CLA:19~22	56.93 $\pm$ 0.19	39.65 $\pm$ 1.35	56.36 $\pm$ 0.49	61.45 $\pm$ 0.38
+ CLA:15~18	39.83 $\pm$ 0.19	3.11 $\pm$ 0.48	45.44 $\pm$ 0.50	28.17 $\pm$ 0.38

TABLE C.1: Performance metrics of various LLM configurations with Cross-Layer Attention (CLA) applied to different layer ranges.

Bibliography

- Adi, Yossi et al. (2017). “Fine-grained Analysis of Sentence Embeddings Using Auxiliary Prediction Tasks”. In: *Proceedings of the 5th International Conference on Learning Representations (ICLR 2017)*.
- Ainslie, Joshua et al. (2023). “GQA: Training Generalized Multi-Query Transformer Models from Multi-Head Checkpoints”. In: *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP 2023)*. Association for Computational Linguistics.
- Ba, Jimmy Lei, Jamie Ryan Kiros, and Geoffrey E. Hinton (2016). “Layer Normalization”. In: arXiv: [1607.06450](#) [cs.LG].
- Bai, Jinze et al. (2023). “Qwen-VL: A Frontier Large Vision-Language Model with Versatile Abilities”. In: arXiv: [2308.12966](#) [cs.CV].
- Belinkov, Yonatan and James Glass (2019). “Analysis Methods in Neural Language Processing: A Survey”. In: *Transactions of the Association for Computational Linguistics* 7, pp. 49–72. DOI: [10.1162/tac1_a_00254](#).
- Bisk, Yonatan et al. (2020). “PIQA: Reasoning about Physical Commonsense in Natural Language”. In: *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI 2020)*, pp. 7432–7439. DOI: [10.1609/aaai.v34i05.6239](#).
- Brandon, William et al. (2024). “Reducing Transformer Key-Value Cache Size with Cross-Layer Attention”. In: *Advances in Neural Information Processing Systems* 37, pp. 86927–86957.
- Brown, Tom B. et al. (2020). “Language Models are Few-Shot Learners”. In: *Advances in Neural Information Processing Systems* 33. Curran Associates, Inc., pp. 1877–1901.
- Chen, Hailin et al. (2023a). “Personalized Distillation: Empowering Open-Sourced LLMs with Adaptive Learning for Code Generation”. In: *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP 2023)*. Singapore: Association for Computational Linguistics, pp. 6737–6749. DOI: [10.18653/v1/2023.emnlp-main.417](#).
- Chen, Ruijun et al. (2023b). “Learning to Memorize Entailment and Discourse Relations for Persona-Consistent Dialogues”. In: *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence (AAAI 2023)*.
- Chiang, Wei-Lin et al. (2023). *Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality*. <https://lmsys.org/blog/2023-03-30-vicuna/>.
- Child, Rewon et al. (2019). “Generating Long Sequences with Sparse Transformers”. In: arXiv: [1904.10509](#) [cs.LG].
- Clark, Kevin et al. (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*. Florence, Italy: Association for Computational Linguistics, pp. 276–286. DOI: [10.18653/v1/W19-4828](#).
- Cobbe, Karl et al. (2021). “Training Verifiers to Solve Math Word Problems”. In: arXiv: [2110.14168](#) [cs.LG].

- Dao, Tri et al. (2022). “FLASHATTENTION: fast and memory-efficient exact attention with IO-awareness”. In: *Advances in Neural Information Processing Systems* 35, pp. 15582–15599.
- Gunasekar, Suriya et al. (2023). “Textbooks Are All You Need”. In: arXiv: [2306.11644 \[cs.CL\]](#).
- He, Kaiming et al. (2016). “Deep Residual Learning for Image Recognition”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016)*, pp. 770–778. DOI: [10.1109/CVPR.2016.90](#).
- Hendrycks, Dan et al. (2021). “Measuring Massive Multitask Language Understanding”. In: *Proceedings of the 9th International Conference on Learning Representations (ICLR 2021)*.
- Hochreiter, Sepp and Jürgen Schmidhuber (1997). “Long Short-Term Memory”. In: *Neural Computation* 9.8, pp. 1735–1780. DOI: [10.1162/neco.1997.9.8.1735](#).
- Hoffmann, Jordan et al. (2022). “Training compute-optimal large language models”. In: *Advances in Neural Information Processing Systems* 35, pp. 30432–30446.
- Hooper, Coleman Richard Charles et al. (2024). “KVQuant: Towards 10 Million Context Length LLM Inference with KV Cache Quantization”. In: *Advances in Neural Information Processing Systems* 37, pp. 1270–1303.
- Jiang, Albert Q. et al. (2023). “Mistral 7B”. In: arXiv: [2310.06825 \[cs.CL\]](#).
- Kaplan, Jared et al. (2020). “Scaling Laws for Neural Language Models”. In: arXiv: [2001.08361 \[cs.LG\]](#).
- Kojima, Takeshi et al. (2022). “Large language models are zero-shot reasoners”. In: *Advances in Neural Information Processing Systems* 35, pp. 22199–22213.
- Lewkowycz, Aitor et al. (2022). “Solving quantitative reasoning problems with language models”. In: *Advances in Neural Information Processing Systems* 35, pp. 3843–3857.
- Li, Jiwei et al. (2016). “A Persona-Based Neural Conversation Model”. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*. Berlin, Germany: Association for Computational Linguistics, pp. 994–1003. DOI: [10.18653/v1/P16-1094](#).
- Linzen, Tal, Emmanuel Dupoux, and Yoav Goldberg (2016). “Assessing the Ability of LSTMs to Learn Syntax-Sensitive Dependencies”. In: *Transactions of the Association for Computational Linguistics* 4, pp. 521–535. DOI: [10.1162/tac1_a_00115](#).
- Liu, Zichang et al. (2023). “Scissorhands: Exploiting the persistence of importance hypothesis for LLM KV cache compression at test time”. In: *Advances in Neural Information Processing Systems* 36, pp. 52342–52364.
- Lundberg, Scott M. and Su-In Lee (2017). “A unified approach to interpreting model predictions”. In: *Advances in Neural Information Processing Systems* 30, pp. 4765–4774.
- Mairesse, François and Marilyn Walker (2007). “PERSONAGE: Personality Generation for Dialogue”. In: *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL 2007)*. Association for Computational Linguistics, pp. 496–503.
- Mukherjee, Subhabrata et al. (2023). “Orca: Progressive Learning from Complex Explanation Traces of GPT-4”. In: arXiv: [2306.02707 \[cs.CL\]](#).
- Olsson, Catherine et al. (2022). “In-context Learning and Induction Heads”. In: arXiv: [2209.11895 \[cs.LG\]](#).
- Ouyang, Long et al. (2022). “Training language models to follow instructions with human feedback”. In: *Advances in Neural Information Processing Systems* 35, pp. 27730–27744.

- Qian, Qiao et al. (2018). "Assigning Personality/Profile to a Chatting Machine for Coherent Conversation Generation". In: *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI 2018)*, pp. 4279–4285. DOI: [10.24963/ijcai.2018/595](https://doi.org/10.24963/ijcai.2018/595).
- Radford, Alec et al. (2019). *Language Models are Unsupervised Multitask Learners*. OpenAI Blog.
- Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin (2016). "'Why Should I Trust You?': Explaining the Predictions of Any Classifier". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2016)*, pp. 1135–1144. DOI: [10.1145/2939672.2939778](https://doi.org/10.1145/2939672.2939778).
- Sap, Maarten et al. (2019). "Social IQa: Commonsense Reasoning about Social Interactions". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP 2019)*, pp. 4462–4472. DOI: [10.18653/v1/D19-1454](https://doi.org/10.18653/v1/D19-1454).
- Shazeer, Noam M. (2019). "Fast Transformer Decoding: One Write-Head is All You Need". In: arXiv: [1911.02150 \[cs.LG\]](https://arxiv.org/abs/1911.02150).
- Simonyan, Karen, Andrea Vedaldi, and Andrew Zisserman (2013). "Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps". In: arXiv: [1312.6034 \[cs.CV\]](https://arxiv.org/abs/1312.6034).
- Song, Haoyu et al. (2021). "BoB: BERT Over BERT for Training Persona-based Dialogue Models from Limited Personalized Data". In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL 2021)*. Association for Computational Linguistics, pp. 167–177.
- Su, Jianlin et al. (2024). "RoFormer: Enhanced transformer with Rotary Position Embedding". In: *Neurocomputing* 568, p. 127063. DOI: [10.1016/j.neucom.2023.127063](https://doi.org/10.1016/j.neucom.2023.127063).
- Sun, Mingjie et al. (2024). "Massive activations in large language models". In: arXiv: [2402.17762 \[cs.LG\]](https://arxiv.org/abs/2402.17762).
- Taori, Rohan et al. (2023). *Alpaca: A strong, replicable instruction-following model*. Stanford Center for Research on Foundation Models.
- Touvron, Hugo et al. (2023). "LLaMA: Open and Efficient Foundation Language Models". In: arXiv: [2302.13971 \[cs.CL\]](https://arxiv.org/abs/2302.13971).
- Vaswani, Ashish et al. (2017). "Attention is All you Need". In: *Advances in Neural Information Processing Systems* 30. Curran Associates, Inc., pp. 5998–6008.
- Vaswani, Ashish et al. (2018). "Tensor2Tensor for Neural Machine Translation". In: *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (AMTA 2018)*. Boston, MA, pp. 193–199.
- Vig, Jesse (2019). "A Multiscale Visualization of Attention in the Transformer Model". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019): System Demonstrations*. Florence, Italy: Association for Computational Linguistics, pp. 37–42. DOI: [10.18653/v1/P19-3007](https://doi.org/10.18653/v1/P19-3007).
- Vig, Jesse 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*. Florence, Italy: Association for Computational Linguistics, pp. 63–76. DOI: [10.18653/v1/W19-4808](https://doi.org/10.18653/v1/W19-4808).
- Wang, Alex et al. (2018). "GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding". In: *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Brussels, Belgium: Association for Computational Linguistics, pp. 353–355. DOI: [10.18653/v1/W18-5446](https://doi.org/10.18653/v1/W18-5446).

- Wang, Xuezhi et al. (2023). "Self-Consistency Improves Chain of Thought Reasoning in Language Models". In: *Proceedings of the Eleventh International Conference on Learning Representations (ICLR 2023)*.
- Wei, Jason et al. (2022). "Chain-of-thought prompting elicits reasoning in large language models". In: *Advances in Neural Information Processing Systems 35*, pp. 24824–24837.
- Wiegrefe, Sarah and Yuval Pinter (2019). "Attention is not not Explanation". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP 2019)*. Hong Kong, China: Association for Computational Linguistics, pp. 11–20. DOI: [10.18653/v1/D19-1002](https://doi.org/10.18653/v1/D19-1002).
- Xiao, Guangxuan et al. (2024). "Efficient Streaming Language Models with Attention Sinks". In: *Proceedings of the Twelfth International Conference on Learning Representations (ICLR 2024)*.
- Yang, Aiyuan et al. (2023). "Baichuan 2: Open large-scale language models". In: arXiv: [2309.10305](https://arxiv.org/abs/2309.10305) [cs.CL].
- Yeh, Catherine et al. (2023). "Attentionviz: A global view of transformer attention". In: *IEEE Transactions on Visualization and Computer Graphics*.
- Ying, Chengxuan et al. (2021). "LazyFormer: Self Attention with Lazy Update". In: arXiv: [2102.12702](https://arxiv.org/abs/2102.12702) [cs.CL].
- Zelikman, Eric et al. (2022). "STaR: self-taught reasoner bootstrapping reasoning with reasoning". In: *Advances in Neural Information Processing Systems 35*, pp. 14629–14642.
- Zellers, Rowan et al. (2019). "HellaSwag: Can a Machine Really Finish Your Sentence?" In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*. Florence, Italy: Association for Computational Linguistics, pp. 4791–4800. DOI: [10.18653/v1/P19-1472](https://doi.org/10.18653/v1/P19-1472).
- Zhang, Saizheng et al. (2018). "Personalizing Dialogue Agents: I have a dog, do you have pets too?" In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018)*. Association for Computational Linguistics, pp. 2204–2213.
- Zheng, Yinhe et al. (2020). "A pre-training based personalized dialogue generation model with persona-sparse data". In: *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI 2020)*, pp. 9693–9700.