Indic Morpho-Semantic Circuits

Druhan Rajiv Shah
IIIT Hyderabad

Sidharth K
IIIT Hyderabad

Anshul Krishnadas Bhagwat
IIIT Hyderabad

Abstract

Mechanistic interpretability methods have been used to obtain task-specific circuits in autoregressive language models that demonstrate understanding of syntactic and semantic information to an extent. However, such analyses have been focused on a single language, usually English, and the circuits obtained have reflected its structure. In this project, we explore circuits identifying indirect objects in a small-scale model for Hindi and compare its complexity and its use of language-specific linguistic features to the standard IOI circuit obtained by [cite]. We find that the circuits differ quite strongly and reflect simultaneously the morphosyntactic system and a dependency grammar structure, suggesting a difference in model learning across languages at a syntactic level, which has impacts on general conclusions drawn from mechanistic analyses.

Introduction

The advent of large-scale generative language models has revolutionized natural language processing (NLP), yet their internal workings remain largely opaque. The field of Mechanistic Interpretability (MI) seeks to address this by reverse-engineering the specific, human-understandable algorithms that models learn during training (Elhage et al., 2021).

A significant portion of MI research has concentrated on English-language models, identifying canonical circuits for tasks like indirect object identification (IOI) (Wang et al.). These circuits often exploit the relatively fixed Subject-Verb-Object (SVO) word order of English. This focus creates a critical gap in our understanding of how a Transformer-based model adapts to the typological diversity of human languages. Hindi, for instance, uses a more flexible but primarily Subject-Object-Verb (SOV) order, marking grammatical roles with distinct postpositional case markers (Verma, 1970).

This study investigates how these typological differences shape the internal circuits of language models. We train a small, autoregressive model (~18M parameters) from scratch on Hindi. Using causal interventions, we identify the neural circuit it develops to identify the recipient of an action and contrast its underlying algorithm with the previously documented IOI circuit for English. By analyzing the mechanisms that emerge in response to a morphologically rich, flexible-word-order language, we can move beyond knowing *that* models work for different languages to understanding precisely *how* they adapt their computational strategies.

The advent of large-scale generative language models has revolutionized natural language processing (NLP), yet their internal workings remain largely opaque. This "black box" problem is a significant barrier to building truly trustworthy and reliable AI systems. The field of Mechanistic Interpretability (MI) seeks to address this by reverse-engineering the specific, human-understandable algorithms that models learn during training (Elhage et al., 2021).

However, a lot of MI research has been concentrated on English-language models, creating a critical gap in our understanding of how a Transformer-based model adapts to the typological diversity of human languages. Different languages encode grammatical relationships in fundamentally different ways. For instance, English relies heavily on a fixed Subject-Verb-Object (SVO) word order. In contrast, Hindi uses a more flexible but primarily Subject-Object-Verb (SOV) order (Verma, 1970), marking grammatical roles with distinct

postpositional particles.

In this project, we explore how these typological differences shape the internal circuits of language models. We will train a small, autoregressive model ($\sim$ 18M parameters) from scratch, one each on English, Hindi, and Telugu. We will then use visualisations, probes, and causal interventions to identify and compare the neural circuits each model develops to process semantic roles.

By comparing the circuits that emerge in response to these distinct grammatical strategies, we can move beyond simply knowing that models work for different languages to understanding precisely how they adapt their computational mechanisms. This study will attempt to provide a mechanistic, comparative analysis of semantic role processing in generative models across different language families, offering foundational insights into the functioning of Transformer-based AI systems.

Background and Related Work

SRL is a central process in NLP that seeks to uncover the roles that words (or their computational counterparts: tokens) perform as arguments to the sentence's core verb or predicate. The PropBank (Palmer et al., 2005) schema provides a set of roles that each argument is classified into, including `ARG0` (the entity performing the action out of their own volition), `ARG1` (the entity on whom the action is performed), and `ARGM-LOC` (the verb modifier that indicates location) among others. While classifier models trained to classify tokens by semantic role are numerous, we aim to study generative models and their use of semantic roles instead.

The representation of semantic roles in each language is unique, with English relying on word order, and Hindi using separate postpositional particles (case markers) to indicate, but not necessarily determine semantic roles (Vaidya et al., 2011). The work by Ghosh et al. relies on the well-established idea that languages which have no Dominant word order (like Hindi) encode information about semantic roles in sentences with morphosyntactic features like *kāraka* markers or *vibhaktī* affixes (Vaidya et al., 2011). Consequently, autoregressive language models can be expected to use differing methods to encode and use information like semantic roles in order to generate tokens.

The field of Mechanistic Interpretability (MI) seeks to reverse-engineer the internal algorithms learned by transformer-based language models during training. This approach moves beyond performance metrics to explain the specific computational mechanisms underlying a model's behavior. The fundamental unit of analysis in MI is the transformer circuit: (Elhage et al., 2021) a subgraph of model parameters and activations responsible for a discrete task. Research in this area has successfully identified key circuits, such as Induction Heads (Olsson et al.), and circuits for indirect object identification (Wang et al.), which are critical for in-context learning and recall. However, a significant portion of MI research has concentrated on English-language models. This focus creates a critical gap in understanding how a transformer's architecture adapts to the vast typological diversity of human languages. Grammatical relationships are encoded in fundamentally different ways across language families, and the circuits developed to process them are likely to differ accordingly.

Background and Related Work

The field of Mechanistic Interpretability seeks to reverse-engineer the internal algorithms learned by transformers. The fundamental unit of analysis is the transformer circuit: a subgraph of model components responsible for a discrete capability (Elhage et al., 2021). Prior work has successfully identified key circuits, such as Induction Heads for in-context learning (Olsson et al.) and, most relevant to this study, circuits for Indirect Object Identification (IOI) (Wang et al.). The IOI circuit in English models was found to rely on specialized heads, such as "Name Mover Heads," that copy subject and object names to the final token position, and "S-Inhibition Heads" that suppress the subject's name, allowing the indirect object to be predicted. This mechanism is closely tied to the positional regularities of English syntax.

However, many languages do not rely on fixed word order to encode semantic roles. Languages with no dominant word order, such as Hindi, often encode this informa-

tion through rich morphosyntactic features like *kāraka* (case) markers (Vaidya et al., 2011, , ghosh.etal2024). For example, the agentive marker *ne* and the dative/accusative marker *ko* are critical for identifying the agent and patient/recipient, respectively. It is therefore plausible that autoregressive models trained on such languages will develop circuits that process these morphological cues, rather than relying on positional heuristics.

Experimental Setup

Model Training

We train a GPT-2 style transformer language model with ~18M parameters (6 layers, 6 heads per layer) on a simple Hindi corpus. The model size is sufficient to learn basic linguistic structures while remaining amenable to detailed circuit analysis.

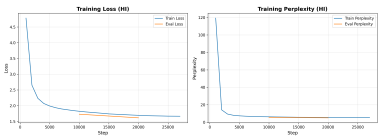

Figure 1: Loss and Perplexity during training

Dataset

We use a Hindi machine-translated variant of the TinyStories dataset[1]. This ensures the semantic content is simple and controlled, allowing the model to focus on learning grammatical structure without the confound of complex world knowledge.

Task Definition

To probe the model's understanding of semantic roles, we design a targeted sentence completion task analogous to the IOI task. We create prompt templates that require the model to identify the recipient (the indirect object) of a ditransitive verb. A typical Hindi template is: `jab [S] aur [IO] bāzār gaye, [S] ne [O] dī.` (Translation: `When [S] and [IO] went to the market, [S] gave [O] to`). Correct completion requires either identifying the noun associated with the `ko` marker or inhibiting the noun with the `ne` marker. We also test on scrambled variants (e.g., OSV order) to verify that the identified circuit is robust to word order changes.

Circuit Analysis

Our analysis proceeds in two stages: localizing crucial components and reverse-engineering their interactions.

1. **Component Localization via Causal Tracing:** We use activation patching (Vig et al.) to identify the model components (attention heads, MLP layers) causally responsible for correct recipient identification. We run the model on a clean prompt and a corrupted prompt where the subject and indirect object roles are swapped. A component is considered critical if patching its activation from the clean run into the corrupted run restores the correct prediction (i.e., flips the output logit difference $\log p(\text{IO}) - \log \text{p(S)}$ from negative to positive).

2. **Information Flow Analysis:** Once critical heads are identified, we analyze their attention patterns to understand the algorithm they implement. We look for heads that move information between key tokens, such as from a noun to its case marker or from a case marker to the verb.

Results

The model achieves high accuracy on the completion task (>93%), including on scrambled-order variants, indicating it has learned a robust mechanism for role identification. Circuit analysis reveals a mechanism starkly different from the canonical English IOI circuit.

The Canonical English IOI Circuit (for comparison)

As established by Wang et al., the English IOI circuit relies on positional and lexical cues. Key components include "Name Mover Heads" that copy both the subject and indirect object names to the final position, and "S-Inhibition Heads" that attend to the subject name tokens earlier in the sequence to suppress their prediction. The circuit's success depends on the consistent SVO ordering of names in the prompt.

The Hindi Morpho-Syntactic Circuit

In contrast, the Hindi model implements a multi-stage, compositional algorithm that directly leverages morphology. We obtained the following different types of heads:

Role Identifier Heads These marked specific syntactic and semantic categories of

[1]Available as OmAlve/TinyStories-Hindi on the Hugging Face Hub.

tokens. Notably, these were independent of the corresponding token's position.

Information Mover Heads As the name suggests, they moved information from the residual stream of one token to another based on relations between the tokens.

Name Mover Heads These essentially decided which token to copy from the context as the final output.

As a result, we were able to obtain the following circuit, notably distinct from the standard IOI circuit in its use of dependency and other syntactic information:

[width=.9]../Assets/IOI_circuit

Figure 2: Hindi IOI Circuit

Crucially, causal patching shows that the output is most sensitive to the activations of the `ko` token, other kāraka particles and their corresponding nouns, regardless of their absolute position in the sentence. This circuit directly computes semantic roles from morphosyntax, a fundamentally more abstract strategy than the positional heuristics of the English IOI circuit.

Discussion

Our comparative analysis reveals that architecturally identical models learn fundamentally different algorithms when trained on typologically distinct languages. The canonical English IOI circuit learns a "cheap" heuristic that exploits the rigidity of English word order. In contrast, the Hindi model develops a more sophisticated, compositional circuit that identifies and binds morphological markers to their respective nouns to robustly identify semantic roles, even in scrambled sentences.

This provides strong mechanistic evidence that Transformer models are not learning a universal set of linguistic operations. Instead, their internal circuits are highly adapted to the statistical patterns and grammatical encoding strategies of the training language. This has significant implications for mechanistic interpretability: conclusions drawn from analyzing English-only models are not guaranteed to be generalizable. A cross-linguistic approach to MI is a necessity for understanding the true range of algorithms learned by these models. Our findings underscore the importance of linguistic typology in shaping the computational mechanisms of AI, a critical consideration for building models that are not just performant but also genuinely interpretable and robust.

Limitations and Future Work

This work can be extended in several key directions. First, applying this analysis to more than one language from more than one language family can verify the generality of the claims we make. Second, applying this analysis to circuits for multiple semantic relations and possibly dependency relations could be a step towards deciding if transformer-based models learn something similar to a dependency grammar. Finally, automating the discovery and comparison of these circuit "motifs" across languages could provide a powerful new tool for computational typology.

Compute declarations

The relevant repository for this project is on GitHub (linked here). All code was run through virtual GPUs provided by Kaggle, and IIITH's HPC cluster Ada.

References

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, and 1 others. 2021. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 1(1):12.

Poulami Ghosh, Shikhar Vashishth, Raj Dabre, and Pushpak Bhattacharyya. A Morphology-Based Investigation of Positional Encodings. *Preprint*, arXiv:2404.04530.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, and 7 others. In-context Learning and Induction Heads. *Preprint*, arXiv:2209.11895.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Ashwini Vaidya, Jinho Choi, Martha Palmer, and Bhuvana Narasimhan. 2011. Analysis of the Hindi Proposition Bank using dependency structure. In *Proceedings of the 5th Linguistic Annotation Workshop*, pages 21–29, Portland, Oregon, USA. Association for Computational Linguistics.

Shivendra Kishore Verma. 1970. Word order in hindi. *Archiv orientalni*, 38:28–32.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Simas Sakenis, Jason Huang, Yaron Singer, and Stuart Shieber. Causal Mediation Analysis for Interpreting Neural NLP: The Case of Gender Bias. *Preprint*, arXiv:2004.12265.

Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the Wild: A Circuit for Indirect Object Identification in GPT-2 small. *Preprint*, arXiv:2211.00593.