THE SAME BUT DIFFERENT: STRUCTURAL SIMILARITIES AND DIFFERENCES IN MULTILINGUAL LANGUAGE MODELING

Ruochen Zhang[1*] **Qinan Yu**[2*†] **Matianyu Zang**[1] **Carsten Eickhoff**[3] **Ellie Pavlick**[1]

[1]Brown University [2]Stanford University [3]University of Tübingen

ABSTRACT

We employ new tools from mechanistic interpretability in order to ask whether the internal structure of large language models (LLMs) shows correspondence to the linguistic structures which underlie the languages on which they are trained. In particular, we ask (1) when two languages employ the same morphosyntactic processes, do LLMs handle them using shared internal circuitry? and (2) when two languages require different morphosyntactic processes, do LLMs handle them using different internal circuitry? Using English and Chinese multilingual and monolingual models, we analyze the internal circuitry involved in two tasks. We find evidence that models employ the same circuit to handle the same syntactic process independently of the language in which it occurs, and that this is the case even for monolingual models trained completely independently. Moreover, we show that multilingual models employ language-specific components (attention heads and feed-forward networks) when needed to handle linguistic processes (e.g., morphological marking) that only exist in some languages. Together, our results provide new insights into how LLMs trade off between exploiting common structures and preserving linguistic differences when tasked with modeling multiple languages simultaneously.

1 INTRODUCTION

As large language models (LLMs) have become the undisputed state of the art for building English language technology, there is decided interest in replicating their success across the full range of human languages. However, very little is known about the internal structure of LLMs, and whether such structure is conducive to acquiring broad multilingual capabilities. In fact, recent research has produced seemingly contradictory findings, such as evidence that multilingual models adopt language-specific representations (Tang et al., 2024; Choenni et al., 2024), while simultaneously showing good transfer across languages even in cases that would appear to have no superficial similarities that can be exploited to aid such transfer (Pires et al., 2019). Given the importance of building technology for diverse languages, there is a need for a more precise understanding of how LLMs represent structural similarities and differences across languages, and whether such representations accord with our intuitive understanding of how languages work.

In this paper, we employ tools from the growing subfield of *mechanistic interpretability* in order to ask whether the internal structure of LLMs show correspondence to the linguistic structures which underlie the languages on which they are trained. We focus on only the most minimal criteria of correspondence. In particular, we ask (1) when two languages employ the same morphosyntactic processes, do LLMs handle them using shared internal circuitry? and (2) when two languages require different morphosyntactic processes, do LLMs handle them using different internal circuitry? While these questions seem simple, their answers are non-obvious. LLMs readily employ overlapping circuitry for tasks that do not necessarily seem "the same" to humans (Merullo et al., 2024), and at the same time, neural networks frequently differentiate concepts due to surface form varia-

*Equal Contribution. Correspondence to ruochen_zhang@brown.edu.

†Work done at Brown University.

tion (Olah et al., 2020), even when humans would easily identify them as part of the same abstract category.

Using English and Chinese multilingual and monolingual models, we analyze the internal circuitry involved in two tasks, one focusing on indirect object identification (IOI) which is virtually identical between the languages, and one which involves generating paste tense verbs that require morphological marking in English but not in Chinese. Our contributions are as follows:

- We show that a multilingual model uses a single circuit to handle the same syntactic process independently of the language in which it occurs (§3.4).
- We show that even monolingual models trained independently on English and Chinese each adopt nearly the same circuit for this task (§3.5), suggesting a surprising amount of consistency with how LLMs learn to handle this particular aspect of language modeling.
- Finally, we show that, when faced with similar tasks that require language-specific morphological processes, multilingual models still invoke a largely overlapping circuit, but employ language-specific components as needed. Specifically, in our task, we find that the model uses a circuit that consists primarily of attention heads to perform most of the task, but employs the feed-forward networks in English only to perform morphological marking that is necessary in English but not in Chinese (§4).

Together, our results provide new insights into how LLMs trade off between exploiting common structures and preserving linguistic differences when tasked with modeling multiple languages simultaneously. Our experiments can lay the groundwork for future works which seek to improve cross-lingual transfer through more principled parameter updates (Wu et al., 2024), as well as work which seeks to use LLMs in order to improve the study of linguistic and grammatical structure for its own sake (Lakretz et al., 2021; Misra & Kim, 2024).

2 Analysis Methods

In this work, we are interested in analyzing how large language models (LLMs) trained in different languages differ in terms of the algorithms and mechanisms they invoke to handle various aspects of language processing. To do this, we employ a few recently developed analysis techniques, described below. These techniques are similar in spirit, but differ in certain details that matter for our analysis. For the most part, we find converging evidence for the paper's main claims across both techniques. When results differ in interesting ways, we comment in our results sections.

2.1 Path Patching

Path patching (Wang et al., 2023; Goldowsky-Dill et al., 2023; Vig et al., 2020; Hanna et al., 2023; Tigges et al., 2023b) has become the most standard and widely-accepted technique within the still-new subfield of *mechanistic interpretability*. The goal of path patching is to localize specific *circuits* within the weights in a trained neural network that play a causal role in model behavior. The setup requires a pair of contrastive inputs, one referred to as the *clean* input and the other as the *corrupted* input.

Path patching caches the activations for both inputs and then replaces the values of individual heads on the clean input with the values those heads would have taken had they been run on the corrupted input. In this way, the method aims to find the specific important head which maximally explains the final logits. Working backward, i.e., through patching the important heads at each layer, path patching has been used to identify full circuits that carry out the task. On its own, path patching only identifies important heads. To gain insight into the specific functions of these heads, path patching is usually used with logit attribution (Nostalgebraist, 2020; Belrose et al., 2023; Dar et al., 2023; Yu et al., 2023) which projects activations into the vocabulary space, as well as with bespoke analysis techniques invented by prior work to explain specific types of heads, such as duplicate-token detection heads or copy heads (Wang et al., 2023).

The advantage of path patching is primarily its wide adoption, which makes it easier to trust results, and enables us to compare with prior work in order to vet the results we are seeing (i.e., checking that we reproduce prior work when we expect to do so). The primary downside is that the method