Ability Transfer Through Language Mixing

Petr Hyner[1,2], **Jan Mrógala**[1,2], **Jan Hůla**[1,3]

[1]Institute for Research and Application of Fuzzy Modeling, University of Ostrava
[2]Department of Informatics and Computers, University of Ostrava
[3]Czech Technical University in Prague, Czech Republic

Correspondence: petr.hyner@osu.cz

Abstract

We systematically investigate cross-lingual ability transfer in language models through controlled experiments across three problem sets: algorithmic addition, graph navigation, and natural language modeling. Our experimental design creates high-resource and low-resource "language" pairs differing in vocabulary, grammar, and computational requirements. We show that training on mixed datasets consistently enables strong positive transfer, significantly improving low-resource language performance compared to training on a small amount of data in isolation. We observe improvements from 0% to 100% accuracy in arithmetic tasks, from 24% to 98% accuracy in graph navigation tasks, and 69.6% perplexity reduction in natural language modeling. We demonstrate that transfer effectiveness depends on computational complexity and linguistic differences, where grammar modifications support stronger transfer than vocabulary modifications. These findings provide compelling evidence that cross-lingual ability transfer is a robust mechanism which contributes to the quality of large language models in low-resource languages.

1 Introduction

Language models have enormous potential for disseminating knowledge and significantly enhancing human problem-solving capabilities. Their performance typically relies on large training corpora, which are predominantly composed of content written in English. Consequently, it is likely that certain topics are well-represented in English but scarcely covered in less commonly used languages. A pessimistic expectation is that models trained under such conditions would fail to answer certain queries in the underrepresented languages. However, several empirical studies provide evidence which suggests that this is not necessarily the case, and that abilities learned in one language can transfer to others (Shaham et al., 2024).

To investigate this phenomenon systematically, we propose an experimental setup that enables us to draw clear and decisive conclusions about cross-lingual transfer. Specifically, we study three distinct setups in which a language model is trained via next-token prediction on datasets consisting of sequences from the same domain but expressed in two different languages. One of these languages is well-represented, such that training solely on it would yield good performance, while the other is sparsely represented, leading to poor performance if trained on in isolation.

Our experiments demonstrate that training on a mixture of the two languages allows the model to transfer capabilities from the high-resource language to the low-resource one. The three experimental setups are designed to reflect different types of abilities that language models can acquire. In the first setup, the task is to learn a basic algorithm (addition), the second setup involves learning to navigate graphs and the third focuses on standard language modeling, with performance measured via perplexity. The languages used in these setups differ in vocabulary and grammar, and in the case of the addition task, also in numerical encoding (Roman versus Arabic numerals) which also requires different underlying algorithm.

Across all three tasks, we observe a consistent positive transfer from the high-resource to the low-resource language. These results strongly suggest that such transfer is likely to occur in current state-of-the-art multilingual language models, even when some languages are significantly underrepresented.

The remainder of this paper is organized as follows. Section 2 reviews related work on cross-lingual transfer, parameter merging, and symbolic reasoning transfer in language models. Section 3 describes our experimental setup, including the three distinct tasks we use to study ability transfer,

Code available at: https://github.com/pe-hy/ability-transfer

Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 1374–1381

the different language representations for each task, and our model architecture and training procedures. Section 4 presents our main experimental results, analyzing the factors that influence transfer effectiveness. Section 5 discusses the implications of our findings and acknowledges the limitations of our approach. Finally, Section 6 concludes with a summary of our contributions and directions for future research.

2 Related Work

When it comes to leveraging the capabilities of LLMs to transfer knowledge from a well-represented language to an underrepresented one, recent works show that, for example, English-centric pre-trained language models can be effectively transferred to other languages with only a tiny fraction of data (<1% of original data). Model with the newly learned language yields performance on par with fully multilingual models (Lee et al., 2025).

Building on this foundation, Schäfer et al. (2024) investigate the unintuitive finding that language imbalance can drive cross-lingual generalization in language models. Using controlled experiments on perfectly equivalent "cloned languages," the authors demonstrate that the presence of a predominant language in the training data boosts the performance of less frequent languages and strengthens the alignment of model representations across languages. They found that a bilingual training dataset with a 90/10 language split can yield better performance on both languages than a balanced 50/50 split, and this effect is amplified with scale.

Another work by Kang and Kim (2025) investigates how factual knowledge is transferred between languages in large language models. They observe that despite multilingual capabilities, LLMs often exhibit "language-binding", where factual knowledge remains tied to the input language, leading to inconsistent recall. Their work highlights that LLMs tend to process information monolingually, influenced by the input's linguistic form. To address this, they propose a Language-to-Thought (L2T) prompting strategy, which aims to decouple the reasoning process from the input language, demonstrating that aligning an LLM's internal thought with the required knowledge is crucial for successful cross-lingual transfer, even without direct translation-based learning.

Similarly, Yu et al. (2024) shows that language models can absorb new capabilities by merging parameters from other models without retraining. They used a method that merges task-specific models, creating composites that often outperform individual source models. This suggests that blending parameters (LoRA adapters, delta weights) can sum abilities across models.

Collectively, these studies illustrate aspects of ability transfer via language and domain mixing: models trained in one context (language or task) can leverage that knowledge in another, especially when training data are combined or structured cleverly. In particular, multilingual and multi-task approaches often allow a model to inherit or enhance abilities across languages and domains (Chen et al., 2024, 2025). These findings motivate our work in which we systematically study the transfer of model abilities in a controlled setup in which we vary certain aspects of generated sequences.

3 Experimental Setup

This section details the data generation process and the structure of training datasets across all experiments; furthermore, we discuss the used training paradigm and model architecture.

3.1 Data

To systematically investigate cross-lingual ability transfer, we design three distinct experimental paradigms that vary along three key dimensions: **task type**, **vocabulary/grammar differences**, and **algorithmic complexity**. Our experimental framework distinguishes between two fundamental task categories:

1. **Problem-solving tasks** where models must learn specific algorithms to generate correct solutions (Addition, Graph Navigation)

2. **Language modeling tasks** where models generate text following natural language patterns (TinyStories (Eldan and Li, 2023))

This categorization allows us to isolate different mechanisms of cross-lingual transfer and understand how it varies with different problem complexities.

For all problem-solving datasets, we systematically identify two critical thresholds: (1) the minimum number of training samples required for successful generalization and (2) the failure threshold where models begin to lose their ability to generalize effectively.

In language modeling, we determine only the second threshold by manually observing a significant drop in generated text quality, while also noting worsened evaluation perplexity. Concerning the first threshold, we simply use all available source language for training.

3.1.1 Addition

Arithmetic addition serves as our primary problem-solving task, chosen for its algorithmic clarity and scalable complexity. While addition appears straightforward for humans with reasonable operand lengths, it presents interesting challenge for Transformers (Nogueira et al., 2021), particularly as operand length increases.

We generate addition problems where both operands have lengths randomly sampled from $3-9$ digits. We format the data in a way, where each digit is tokenized separately, e.g. `32 + 10 = 42` is tokenized as `3 2 + 1 0 = 4 2`, following previous work (Lee et al., 2023), where authors demonstrated improved performance with least-significant-token-first ordering, we reverse the output sequence: `3 2 + 1 0 = 2 4`.

To study transfer across different mathematical representations, we implement four distinct "languages" for arithmetic:

- **Standard**: Standard decimal notation (`3 2 + 1 0 = 2 4`)
- **Letter**: Different vocabulary representation ($0 \to A$, $1 \to B$, ..., $9 \to J$) with modified syntax (`! A B , C D = E F`)
- **Roman**: Roman numeral representation (`X X I + I V = V X X`)
- **Hexadecimal**: Base-16 arithmetic (`1 A + B = 5 2`)

These variants implicitly contain various algorithmic complexities. The letter mapping preserves the underlying decimal addition algorithm while only changing surface syntax, whereas Roman and hexadecimal variants require different algorithms. This setup allows us to then distinguish between surface-level transfer and deeper transfer of model capabilities.

3.1.2 Graph Navigation Problem

Another task for which we test transfer of abilities from one language to another is graph navigation. Graph navigation can be seen as a prototypical task which in the most simplest form tests compositional reasoning where one needs to compose several steps seen independently to reach a target state from an initial state in a fixed graph.

Each instance of this problem is determined by a initial state and a target state and the task of the model is to produce sequence of tokens which corresponds to a valid path between these two states. To evaluate the trained model, unseen pairs of initial and target states are used.

Our training data for this task consist of different shortest paths between each two nodes in the fixed randomly generated Erdős–Rényi graph with 100 nodes. We set the probability of edge creation parameter to 0.02, which results in 152 edges. There are, therefore, 10000 possible pairs of initial and target states from which we sample the training and testing data.

To introduce language diversity in this task we can again change the vocabulary and the grammar. To change the vocabulary, we re-index the nodes so that the new token indices are unique (by adding a large constant to original indices). To change the grammar we express each path as a sequence of edges instead of sequence of nodes.

For our transfer experiments, we test two different target language sample sizes (256 and 512) representing low-performing and moderately-performing baselines respectively, in order to show that cross-lingual transfer benefits occur across different low-resource scenarios.

Khona et al. (2024) provided a partial explanation for how Transformers are able to learn graph navigation. They have shown that the Transformer embeds the nodes of the graph into a vector space such that the distances between node embeddings reflect shortest path metric on the graph and then use this metric to decide which node to expand next.

3.1.3 TinyStories

TinyStories is a synthetic natural language dataset which consists of text written in simple English, uses a limited vocabulary and simple grammatical structures, making it ideal for studying language modeling in controlled conditions. For experiments, we must first translate the dataset and effectively obtain a parallel corpus. This parallel corpus can be in any language, but we deliberately chose the Czech language, since a high quality translation was already obtained in (Hyner et al., 2024).

We can then train a tokenizer on both corpora to obtain a shared vocabulary via the BPE algorithm. Since the amount of text of both languages is similar, we can assume that the tokenizer will contain an unbiased set of tokens, capturing both languages efficiently. The total number of tokens (vocab. size) is set to 30000.

3.2 Model and Training Details

For Addition and Graph Navigation Problem, we use the LitGPT (AI, 2023) implementation of GPT architecture (Radford et al., 2018), where we train from scratch and compute loss only from the model output w.r.t. ground truth after some pre-set delimiter. For example, in the Addition problem set, this would be the equals sign (=). We then measure the model quality using an appropriate metric, in the Addition case, we evaluate using simple exact match accuracy metric. In Graph Navigation problem set, we use a validity metric, which we explain further in Section 4. The model size varies across experiments, which we show in Table 1.

For these two tasks, we use simple word-level tokenizers rather than typical subword tokenizers (BPE (Sennrich et al., 2016), WordPiece (Song et al., 2021)). Since we deliberately structure our text data so that each meaningful unit (number, operator, etc.) is separated by whitespace, WordLevel tokenization naturally aligns with our data structure, whereas subword tokenizers would unnecessarily fragment these atomic units.

For the TinyStories data, we train GPT-Neo (Black et al., 2021) models from the Transformers library (Wolf et al., 2020) for open-ended generation. Here, we measure the quality using the perplexity metric.

We evaluate cross-lingual transfer by training models on mixed datasets containing varying proportions of source and target language examples. We compare the relevant metric to the setup where the model is trained on the individual languages alone. For evaluation, we use held-out test sets of 1024 examples for Addition and Graph Navigation tasks, and measure perplexity on 4096 examples for TinyStories. The test sets remain the same across seeds. In the mixed setting, we evaluate on both languages independently using separate test sets for each language.

All models were trained on a single Nvidia 40GB A100 GPU. For problem-solving tasks, it takes approximately 1 hour to train each model. Training full TinyStories models took 40 hours, training limited TinyStories models took 3 hours.

Table 1: Used Model Parameters

Parameter	Addition	Graph Search	TinyStories
Layers	12	6	4
Heads	8	4	16
Emb. Size	128	64	768
Batch Size	1024	64	80[a]
Learning Rate	0.002	0.003[b]	0.0005
Scheduler	Lin. Incr.	Cos. Warmup	Constant
Block Size	64	128	512
Model Size	$\sim 2.5 \times 10^6$	$\sim 3.1 \times 10^5$	$\sim 3.3 \times 10^7$

[a] Gradient Accumulation is set to 16. [b] Maximum learning rate; increase per batch is determined by warmup steps (1560 for 10k training samples).

We note that we train each **mixed** model 3 times to obtain and report the mean value and standard deviation of the corresponding evaluation metric.

4 Experimental Results

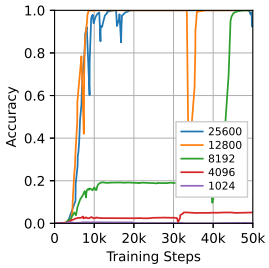
In this section, we present our main empirical contribution: language mixing can significantly improve model performance on target languages across three distinct problem domains. Our results demonstrate what we term "ability transfer" - the phenomenon where model capabilities acquired in one linguistic representation transfer to enhance performance in another. This transfer effect proves particularly beneficial for target languages with limited training data, which suggests that cross-lingual training can help as an effective form of data augmentation for low-resource scenarios.

We evaluate transfer effectiveness using three different metrics for each task type. For arithmetic problems, we use **exact match accuracy**, measuring whether the generated sequence of tokens precisely match the ground truth sequence. For graph navigation tasks, we use **validity**, which assesses whether the generated node sequence represents a valid path in the graph with correct starting and ending nodes, regardless of optimality - acknowledging that models may discover shortest paths, sub-optimal paths, or reproduce the exact training path. For language modeling tasks, we measure **perplexity** to evaluate how well the model predicts the probability distribution of target language text, with lower values indicating better language understanding.

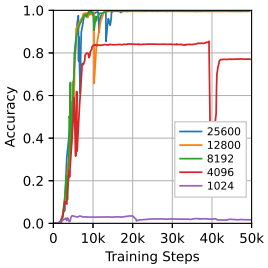
4.1 Addition

We begin by training models on each arithmetic representation across varying sample sizes to es-

tablish the two thresholds previously defined in Section 3.1. In Figure 1, we show exact match accuracy results for standard and Roman addition as representative examples. For standard addition, we observe that models fail to generalize at 4k samples, while with 8k samples, the model achieves perfect accuracy. Roman addition shows a more gradual performance curve, where a model achieved over 80% accuracy even at 4k samples.



(a) Standard Addition dataset training curves showing the impact of training sample size on model performance.



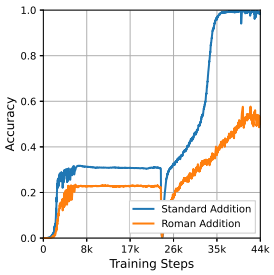
(b) Roman Addition dataset training curves showing the impact of training sample size on model performance.

Figure 1: Comparison of training sample size impact across different datasets. We note that in Roman Addition, even a model trained on 4k samples achieves $> 80\%$ exact match accuracy.

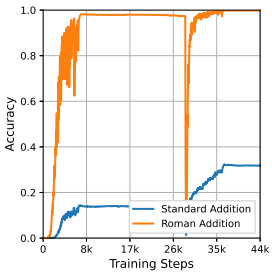
Using these thresholds, we design transfer experiments with previously determined sample ratios for source and target languages. We train models with identical hyperparameters on mixed datasets while varying language proportions. The models are then evaluated on exact match accuracy for both languages.

Figure 2 visualizes that transfer direction might impact model quality. When standard addition serves as the source language, models achieve substantial improvements on target languages. However, when we reverse the language roles, using Roman addition as the source, models fail to learn standard addition despite having access to double the target language training samples.

Table 2 shows transfer results across all language pairs, comparing model performance when models are trained independently on individual languages or on mixed datasets. The results show that mixed training always improves performance on the second language while maintaining high accuracy on the first language. Interestingly, letter and hexadecimal addition achieve near-perfect transfer (98-100% accuracy), while Roman addition shows moderate improvements.



(a) Standard → Roman transfer shows +40% accuracy improvement compared to Roman-only training.



(b) Roman → Standard transfer fails to generalize, despite doubled target language sample size.

Figure 2: Exact match accuracy, where target language is Roman and source language is Standard in 2a and switched in 2b. The accuracies correspond to Table 2. We observe asymmetric transfer effects between standard and Roman addition, demonstrating that transfer direction in this case determines model quality.

4.2 Graph Navigation Problem

The process to obtain the training data sizes is the same as in Section 4.1. However, the goal of this experiment is slightly different. While with addition, we did not explicitly discuss how specifically the languages differ, in this experiment, the goal is to observe the impact of lexical vs. syntactic manipulation on ability transfer. Here, we aimed to show the relationship between linguistic similarity and transfer effectiveness. Table 3 demonstrates that the type of cross-lingual difference significantly impacts transfer capability. When mixing sequential node format with edge format, we observe nearly perfect transfer, strongly outperforming baseline performance. This suggests that models can effectively learn to map between different grammatical representations of the same underlying problem structure. The vocabulary variant produces more modest improvements. The combination of both types of modifications presents a significantly more difficult task, while still showing decent improvements over baseline performance.

4.3 TinyStories

Table 4 shows cross-lingual transfer results for natural language modeling. When training with only 10% Czech data, mixing with 90% English data improves Czech perplexity from 28.67 to 8.72. While the mixed approach doesn't reach the performance of full Czech training (5.65), it shows that cross-lingual transfer provides substantial benefits for low-resource language modeling scenarios.

Different from our algorithmic tasks where trans-

Language Pair	Dataset Size		Lang 1 Accuracy		Lang 2 Accuracy	
	Lang 1	Lang 2	Alone	Mixed	Alone	Mixed
Standard + Letter	11520	1280	100%	98.6% ± 1.4%	0%	98.2% ± 1.6%
Standard + Roman	8192	2048	100%	98.9% ± 1.7%	19.42%	59.38% ± 3.5%
Standard + Hex	8192	4096	100%	97.5% ± 2.1%	0%	97.94% ± 2.9%
Letter + Standard	8192	4096	100%	98.4% ± 1.8%	22.98%	98.1% ± 1.5%
Roman + Standard	8192	4096	100%	98.2% ± 2.0%	22.98%	32% ± 5.8%
Hex + Standard	25600	4096	100%	98.73% ± 1.3%	22.98%	96.09% ± 2.7%

Table 2: Cross-lingual transfer results showing the impact of language mixing on model performance across different addition tasks.

Language Pair	Dataset Size		Lang 1 Accuracy		Lang 2 Accuracy	
	Lang 1	Lang 2	Alone	Mixed	Alone	Mixed
Diff. Vocab	2048	256	97.3%	99.3 ± 0.9%	23.9%	57.3 ± 4.2%
Diff. Grammar	2048	256	98.3%	98.9 ± 1.1%	23.9%	98.1 ± 1.8%
Diff. Gram. & Vocab	2048	256	98.3%	98.8 ± 1.2%	23.8%	47.6 ± 3.6%
Diff. Vocab	2048	512	97.3%	98.3 ± 1.6%	64.5%	82.6 ± 2.9%
Diff. Grammar	2048	512	98.3%	99.4 ± 0.8%	64.5%	97.2 ± 1.7%
Diff. Gram. & Vocab	2048	512	98.3%	99.5 ± 0.7%	68.4%	85.2 ± 3.1%

Table 3: Cross-lingual transfer results showing the impact of language mixing on model performance across different graph navigation trace formats. This table examines three types of cross-lingual differences: (1) **Diff. Grammar** mixes sequential node lists (`1 , 2 , 3 , 4`) with consecutive node pairs (`( 1 , 2 ) , ( 2 , 3 )`); (2) **Diff. Vocab** mixes consecutive node pairs with numerical offset (`( 10001 , 10002 ), ( 10002 , 10003 )`) with standard indices (`( 1 , 2 ), ( 2 , 3 )`); and (3) **Diff. Grammar and Vocab** mixes sequential node lists with offset consecutive pairs, combining both grammatical and vocabulary differences. Results show grammar differences enable strongest transfer while vocabulary differences show moderate transfer.

Training Configuration	Data Composition		Test Perplexity
	Czech	English	
English-only	0%	100%	3.94
Czech-only (Limited)	10%	0%	28.67 ± 1.3
Czech-only (Full)	100%	0%	5.65
Mixed Training	**10%**	**90%**	**8.72** ± 1.2

Table 4: Cross-lingual transfer results for TinyStories language modeling. Mixed training with 10% Czech and 90% English data improves Czech language performance compared to Czech-only training with limited data.

fer enabled models to solve previously impossible problems (0% $\rightarrow$ 100% accuracy), language modeling shows more grounded improvements. This suggests different transfer mechanisms: algorithmic tasks perhaps benefit from learning discrete computational procedures that generalize across representations, while language modeling involves learning continuous distributions over vocabularies and grammatical structures.

The Czech-English language pair provides a challenging test case, as these languages differ substantially in morphology, syntax, and vocabulary. The observed transfer indicates that models can leverage common semantic concepts and patterns even across typologically distant languages.

5 Discussion

Our experiments provide compelling evidence that cross-lingual ability transfer is a robust phenomenon that occurs across fundamentally different types of tasks. The consistent positive transfer we observe suggests that this phenomenon extends beyond surface-level linguistic similarities and leverages computational mechanisms within transformer architectures. The experimental results provided several interesting observations. The asymmetric transfer patterns we observe, particularly in arithmetic tasks where standard notation transfers effectively to Roman numerals but not vice versa, suggest that transfer effectiveness depends on the computational complexity and learnability of the underlying algorithms. This asymmetry indicates that transfer is not merely about shared semantic concepts but involves the transferability of learned computational procedures. The graph navigation experiments reveal that grammatical differences facilitate stronger transfer than vocabulary differences alone. The near-perfect transfer (98.05% accuracy) when mixing sequential node lists with edge representations suggests that models can effectively learn to map between different grammatical encodings of the same underlying relational structure.

6 Conclusion

This work provides systematic empirical evidence for cross-lingual ability transfer in language models through controlled experiments across three distinct problem domains. Our findings demonstrate that models can effectively transfer capabilities from high-resource languages to low-resource ones, even when the linguistic representations differ substantially in vocabulary, grammar, and underlying algorithms.

Unlike previous work that primarily studied transfer in natural language tasks, our synthetic problem domains enable precise measurement of transfer effectiveness and isolation of contributing factors. We demonstrate that language mixing consistently improves performance on target low-resource languages across problem-solving tasks (addition, graph navigation) and language modeling (TinyStories). In arithmetic tasks, we observed improvements from 0 % to 100% accuracy, while in language modeling, we achieved a 69.6% perplexity reduction for Czech when mixed with English training data.

Limitations

Despite the compelling evidence for cross-lingual ability transfer, our study has several important limitations that should be considered when interpreting these results. Our experiments focus on controlled synthetic tasks that may not fully capture the complexity of real-world language use. The synthetic datasets have uniform difficulty distributions and balanced complexity within each language. Real-world multilingual datasets exhibit significant variation in domain coverage, text quality, and complexity across languages. These factors could substantially influence transfer effectiveness in practical applications. Furthermore, our experiments use relatively small transformer models compared to contemporary large language models. Transfer mechanisms may behave differently at scale, and larger models might exhibit more robust transfer across greater linguistic distances. Additionally, we focus exclusively on decoder-only transformer architectures, and other architectures might show different transfer characteristics.

Acknowledgments

This article has been produced with the financial support of the European Union under the : Biography of Fake News with a Touch of AI: Dangerous Phenomenon through the Prism of Modern Human Sciences project no.: CZ.02.01.01/00/23_025/0008724 via the Operational Programme Jan Ámos Komenský. Model training and evaluation was supported by the Ministry of Education, Youth and Sports of the Czech Republic through the e-INFRA CZ (ID:90254).

References

Lightning AI. 2023. Litgpt. https://github.com/Lightning-AI/litgpt.

Sid Black, Leo Gao, Phil Wang, Connor Leahy, and Stella Biderman. 2021. Gpt-neo: Large scale autoregressive language modeling with mesh-tensorflow.

Nuo Chen, Zinan Zheng, Ning Wu, Ming Gong, Dongmei Zhang, and Jia Li. 2024. Breaking language barriers in multilingual mathematical reasoning: Insights and observations. *Preprint*, arXiv:2310.20246.

Zhipeng Chen, Kun Zhou, Liang Song, Wayne Xin Zhao, Bingning Wang, Weipeng Chen, and Ji-Rong Wen. 2025. Extracting and transferring abilities for building multi-lingual ability-enhanced large language models. *Preprint*, arXiv:2410.07825.

Ronen Eldan and Yuanzhi Li. 2023. TinyStories: How small can language models be and still speak coherent english? *Preprint*, arxiv:2305.07759 [cs].

Petr Hyner, Petr Marek, David Adamczyk, Jan Hůla, and Jan Šedivý. 2024. Stealing brains: From english to czech language model. In *Proceedings of the 16th International Joint Conference on Computational Intelligence*, pages 606–612, Porto, Portugal. SCITEPRESS - Science and Technology Publications.

Eojin Kang and Juae Kim. 2025. Llms are globally multilingual yet locally monolingual: Exploring knowledge transfer via language and thought theory. *Preprint*, arXiv:2505.24409.

Mikail Khona, Maya Okawa, Jan Hula, Rahul Ramesh, Kento Nishi, Robert Dick, Ekdeep Singh Lubana, and Hidenori Tanaka. 2024. Towards an understanding of stepwise inference in transformers: A synthetic graph navigation model. *arXiv preprint arXiv:2402.07757*.

Jungseob Lee, Seongtae Hong, Hyeonseok Moon, and Heuiseok Lim. 2025. Cross-lingual optimization for language transfer in large language models. *Preprint*, arXiv:2505.14297.

Nayoung Lee, Kartik Sreenivasan, Jason D. Lee, Kangwook Lee, and Dimitris Papailiopoulos. 2023. Teaching arithmetic to small transformers.

Rodrigo Nogueira, Zhiying Jiang, and Jimmy Lin. 2021. Investigating the limitations of transformers with simple arithmetic tasks. *Preprint*, arxiv:2102.13019 [cs].

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Anton Schäfer, Shauli Ravfogel, Thomas Hofmann, Tiago Pimentel, and Imanol Schlag. 2024. The role of language imbalance in cross-lingual generalisation: Insights from cloned language experiments. *Preprint*, arXiv:2404.07982.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Uri Shaham, Jonathan Herzig, Roee Aharoni, Idan Szpektor, Reut Tsarfaty, and Matan Eyal. 2024. Multilingual instruction tuning with just a pinch of multilinguality. *Preprint*, arXiv:2401.01854.

Xinying Song, Alex Salcianu, Yang Song, Dave Dopson, and Denny Zhou. 2021. Fast WordPiece tokenization. *Preprint*, arxiv:2012.15524 [cs].

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. 2024. Language models are super mario: Absorbing abilities from homologous models as a free lunch. *Preprint*, arXiv:2311.03099.