# How Does a Multilingual LM Handle Multiple Languages?

Shashwat Bhardwaj
2023AIY7528
Team Stark
Indian Institute of Technology Delhi
aiy237528@iitd.ac.in

November 26, 2024

## Abstract

In this mini-project, the aim is to explore how multilingual language models, in this case, BLOOM-1.7B, process the knowledge and transfer it across different languages. The investigation is divided into three tasks: we start by exploring word embeddings similarity across languages to estimate semantic alignment; second, we probe the internal behavior of the model using attention maps and transformer hooking techniques to better understand the model's multilingual representations. Final experiment: We test the cross-lingual transferability of the model by fine-tuning it on English datasets and then testing it on Bengali and Gujarati datasets. The results show the strengths and weaknesses of the model in multilingual understanding and transfer learning.

## Introduction

This report investigates the multilingual capabilities of the BLOOM-1.7B language model through three tasks.

**Task 1** examines word embeddings in English, French, and Hindi using cosine similarity to evaluate the semantic alignment of embeddings across languages.

**Task 2** explores BLOOM's internal representations through attention map visualizations and probing techniques like short-circuiting and bypassing layers, revealing how layers encode language understanding.

**Task 3** evaluates cross-lingual transferability on Bengali and Gujarati datasets by fine-tuning BLOOM-1.7B, RoBERTa, and Indic-BERT on English datasets and analyzing performance on low-resource languages via accuracy plots and confusion matrices. This highlights factors affecting cross-lingual transfer.

## Task 1: Similarity Between Word Embeddings in Different Languages

In this task, we investigate the multilingual semantic alignment in BLOOM-1.7B by examining word embeddings of semantically identical words across three languages: English, French, and Hindi. The key steps and analyses performed are outlined below:

### Dataset Preparation

A parallel dataset of translated words was utilized, containing the same words across English, French, and Hindi. These words serve as a basis to evaluate the alignment of word embeddings produced by BLOOM-1.7B.

### Embedding Extraction

For each word in the dataset, its corresponding embedding was extracted from BLOOM-1.7B. The embeddings represent the semantic information of the words in the high-dimensional vector space. These embeddings were then reduced to a three-dimensional space using Principal Component Analysis (PCA) for visualization purposes.

## Visualization

A 3D scatter plot was created to visualize the embeddings of words in the reduced vector space. Each point in the plot corresponds to a word embedding, color-coded based on its language (English, French, or Hindi).

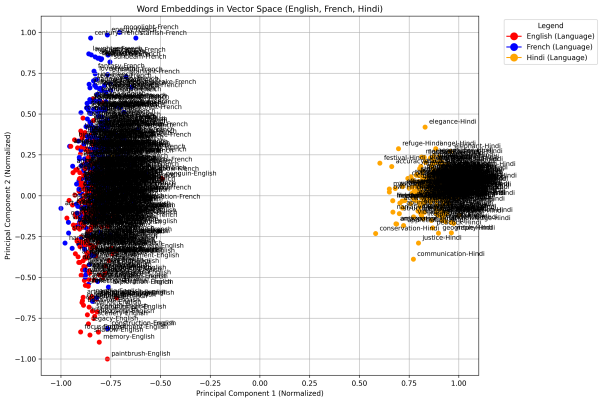


Figure 1: 3D Word Embeddings Visualization for English, French, and Hindi

## Clustering and Similarity Analysis

To further analyze the embeddings:

- **Clustering:** KMeans clustering was applied to group embeddings into clusters, representing patterns across languages.
- **Cosine Similarity:** The cosine similarity between embeddings of the same word in different languages (e.g., English-French, English-Hindi, French-Hindi) was calculated. This metric quantifies the semantic closeness of the embeddings across languages.

The results indicate that English and French, being linguistically closer, also exhibit proximity in the embedding space, whereas Hindi, belonging to the Indic class of languages, occupies a distinct embedding space. This highlights the extent to which BLOOM-1.7B aligns the embeddings of semantically identical words in its multilingual representation. The cosine similarity values offer valuable insights into the model's ability to capture multilingual relationships.

## 'Word Meaning' as a dimension spans the embedding space

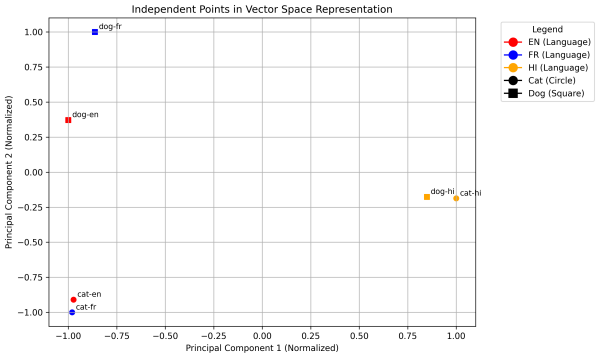


Figure 2: Cat and Dog in English, French and Hindi

As we can see from the plot, cat in english and french appear close in embedding space and similarly dog also, but the embeddings of cat are really very close in english and french.

But as per Hindi's perspective it lies completely away from english and french in embedding space, signifying belonging ness to different class of language families.

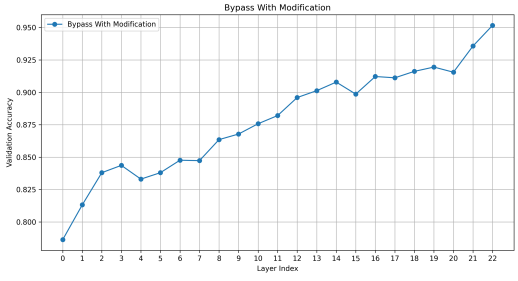


Figure 3: Byepass with modification

## Output

- A CSV file was generated to record the cosine similarity scores for each word pair across the three languages.

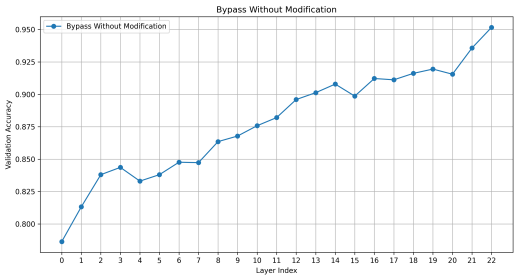


Figure 4: Byepass without modification

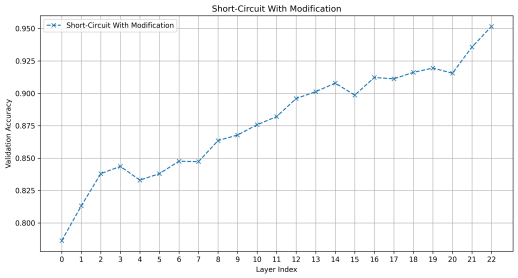


Figure 5: Shortcircuit with modification

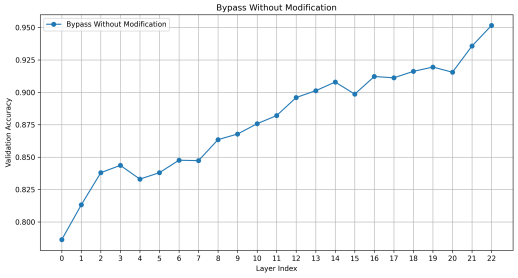


Figure 6: Shortcircuit without modificationn

- The clustering results were overlaid onto the 3D plot, with each cluster represented by a unique color and the centers marked for reference.

These analyses collectively showcase BLOOM-1.7B's capability to align semantically identical words across multiple languages, demonstrating its multilingual semantic representation strengths.

# Task 2: Attention Map Visualization and Transformer Hooking

This task explores the internal workings of BLOOM-1.7B by visualizing attention maps and employing hooking techniques to understand the role of individual layers and their contribution to the model's overall performance.

## Attention Map Visualization

Attention maps were generated for various layers and heads of BLOOM-1.7B to investigate how the model attends to different tokens in input sentences. Two types of attention visualizations were produced:

- **Single-Head Attention Maps:** These maps illustrate the attention weights for individual heads, highlighting token-to-token relationships.

- **Aggregated Attention Maps:** These maps combine attention weights across all heads in a layer, providing a holistic view of the layer's focus.

The visualizations were created for input sentences in English and Punjabi to observe cross-lingual behavior.

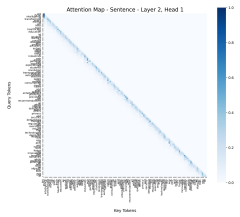


Figure 7: Attention Map: English, Layer 2, Head 1

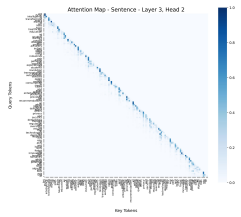


Figure 8: Attention Map: English, Layer 3, Head 2

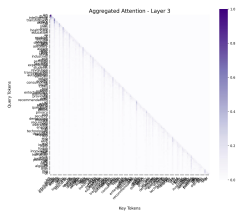

Figure 9: Aggregated Attention Map: English, Layer 7

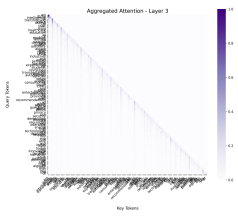

Figure 10: Aggregated Attention Map: English, Layer 12

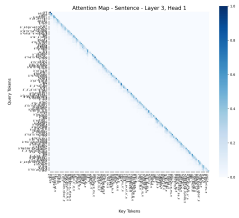

Figure 11: Attention Map: Punjabi, Layer 3, Head 1

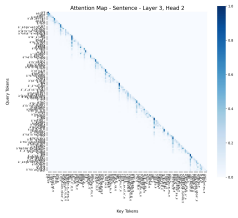

Figure 12: Attention Map: Punjabi, Layer 3, Head 2

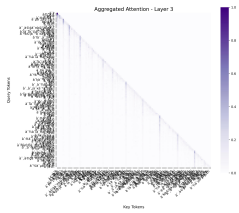

Figure 13: Aggregated Attention Map: Punjabi, Layer 7

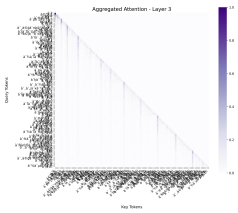

Figure 14: Aggregated Attention Map: Punjabi, Layer 12

### Transformer Hooking and Probing Techniques

To understand the role of individual layers in BLOOM-1.7B, probing techniques were employed using transformer hooking mechanisms. These techniques include:

- **Bypassing Layers:** This technique involves bypassing a specific layer by equating its output to the previous layer's output. It helps analyze the contribution of the bypassed layer to the model's performance.
- **Short-Circuiting Layers:** This technique involves zeroing out the outputs of a specific layer to measure its impact on downstream tasks. It allows us to identify the importance of the layer for specific tasks.

Logistic regression probes were applied to classify hidden states extracted from each layer, both with and without bypassing or short-circuiting. The classification accuracy was tracked across all layers to evaluate the significance of individual layers.

One critical observation is that as we hook into deeper layers of the model, the hidden states become more and more enriched which is evident from increasing accuracies across layers hooks.

### Observations

- **Attention Maps:** The attention visualizations showed that BLOOM-1.7B effectively attends to critical tokens across layers and languages, demonstrating its cross-lingual understanding capabilities.
- **Transformer Hooking:** The probing experiments revealed significant variations in performance when bypassing or short-circuiting layers. Specific layers showed higher contributions to the downstream task, indicating their importance in processing critical features.

These experiments provide deeper insights into BLOOM-1.7B's internal architecture and its ability to encode meaningful representations for multilingual tasks.

## Task 3: Cross-Lingual Transferability

In this task, three models—BLOOM-1.7B, RoBERTa, and IndicBERT—were fine-tuned and evaluated on Bengali and Gujarati datasets. The Bengali dataset contained two sentiment categories (Positive and Negative), while the Gujarati dataset had three categories (Business, Entertainment, and Tech). The goal was to analyze the models'

ability to generalize across languages and datasets. The results include train-validation loss plots and a performance comparison table.

## Train-Validation Loss Plots

The train and validation loss plots for each model and dataset are presented below. These plots illustrate the convergence behavior and generalization ability of the models during fine-tuning.

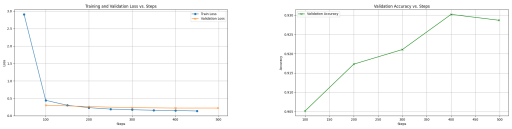

Figure 15: Train-Val Loss and Val Accuracy Plot on Gujarati Dataset BLOOM 1.7B model)

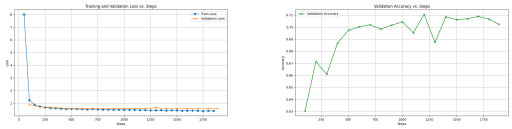

Figure 16: Train-Val Loss and Val Accuracy Plot of Bengali Dataset BLOOM 1.7B model

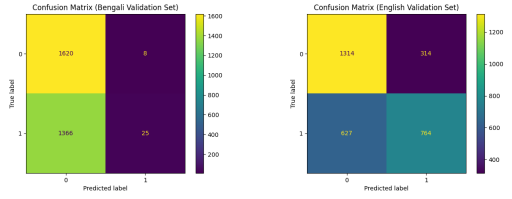


Figure 17: Confusion Matrix Val on English- Bengali Dataset BLOOM 1.7B model

## Performance Results

The table below summarizes the evaluation results for each model on both the English and native-language versions of the Bengali and Gujarati datasets.

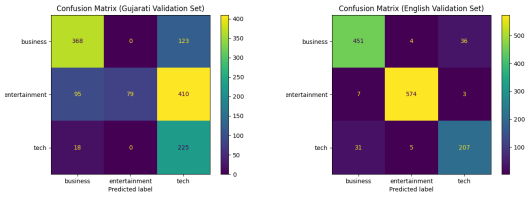


Figure 18: Confusion Matrix Val on English- Gujarati Dataset BLOOM 1.7B model

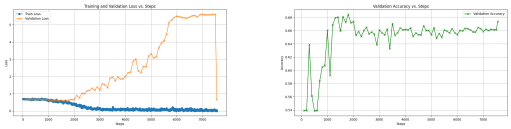

Figure 19: Train-Val Loss and Val Accuracy Plot of Bengali Dataset IndicBERT model)

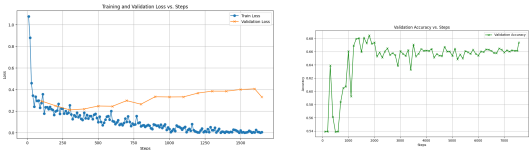

Figure 20: Train-Val Loss and Val Accuracy Plot of Bengali Dataset IndicBERT model

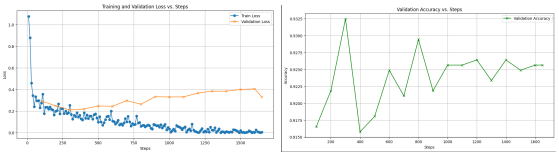

Figure 21: Train-Val Loss and Val Accuracy Plot of Gujarati Dataset RoBERTa model)

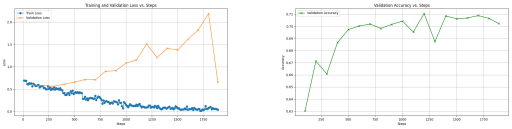

Figure 22: Train-Val Loss and Val Accuracy Plot of Bengali Dataset RoBERTa

| Model | Dataset | English Accuracy | Native Accuracy |
|---|---|---|---|
| RoBERTa | Bengali | 0.9385 | 0.4355 |
| IndicBERT | Bengali | 0.9256 | 0.6851 |
| BLOOM-1.7B | Bengali | 0.6883 | 0.5449 |
| RoBERTa | Gujarati | 0.9385 | 0.4355 |
| IndicBERT | Gujarati | 0.9256 | 0.6851 |
| BLOOM-1.7B | Gujarati | 0.9347 | 0.5099 |

Table 1: Accuracy Results for Bengali and Gujarati Datasets [0-1]

### Observations

- **Bengali Dataset:** IndicBERT outperformed BLOOM-1.7B and RoBERTa in native Bengali sentiment classification. The English versions showed significantly higher accuracy compared to native datasets.
- **Gujarati Dataset:** Similar trends were observed for Gujarati, with IndicBERT achieving the highest accuracy in the native language, though RoBERTa and BLOOM-1.7B performed better on the English dataset.
- **Cross-Lingual Performance:** The performance drop from English to native datasets highlights the challenge of cross-lingual transfer, especially for underrepresented languages.

### Possible Explanation for Results

The observed disparity in performance between the English datasets and the native-language datasets (Bengali and Gujarati) can be attributed to the following factors:

- **Training Data Representation:** Pretrained models have richer representations for English due to the abundance of high-quality English corpora compared to low-resource languages like Bengali and Gujarati.
- **Language Complexity and Script Differences:** Bengali and Gujarati are morphologically rich and use scripts distinct from the Latin script, increasing the difficulty for models to process them effectively.
- **Model Pretraining Bias:** BLOOM-1.7B and RoBERTa are pretrained on datasets biased toward high-resource languages, whereas IndicBERT benefits from pretraining on Indic languages.
- **Cross-Lingual Transfer Limitations:** Cross-lingual transfer is less effective due to structural and phonological differences between English and Indic languages.
- **Fine-Tuning Data Size:** Smaller fine-tuning datasets for Bengali and Gujarati limit the models' ability to adapt to these languages effectively.
- **Class Complexity:** Gujarati's three-class classification task is inherently more challenging than Bengali's binary classification, contributing to lower performance.

## Future Scope of Improvements

- In Task-2, my probing currently is giving the same outputs whether I short circuit it or bypass it. I will try to investigate this further.
- I will also try to visualize eigenvectors of attention maps to uncover more structural patterns.
- Additionally, I plan to train the Bloom 1.7B model for more epochs if the validation loss continues to decrease significantly.

## Github Link of Repository

The code for this mini-project, including the experiments and visualizations, is available on GitHub at the following link:

`https://github.com/raja17021998/LLM-AIL-821-Project`

## References

[1] BigScience Workshop, "BLOOM: A 176B-parameter Open Multilingual Language Model", arXiv preprint, arXiv:2211.05100, 2022.

[2] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... Stoyanov, V., "RoBERTa: A Robustly Optimized BERT Pretraining Approach", arXiv preprint, arXiv:1907.11692, 2019.

[3] Kakwani, D., Shukla, A., Khanuja, S., Awasthi, A., Sinha, R., Shrivastava, M., "IndicNLPSuite: Monolingual Corpora, Evaluation Benchmarks and Pre-trained Multilingual Language Models for Indian Languages", Findings of ACL, 2020.

[4] MacQueen, J., "Some Methods for classification and Analysis of Multivariate Observations", Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1, pp. 281-297, 1967.

[5] Jolliffe, I. T., "Principal Component Analysis", Springer Series in Statistics, Springer, New York, 2002.

[6] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I., "Attention Is All You Need", Advances in Neural Information Processing Systems, 2017.

[7] Singhal, A., "Modern Information Retrieval: A Brief Overview", IEEE Data Engineering Bulletin, vol. 24, no. 4, pp. 35-43, 2001.

[8] Clark, K., Khandelwal, U., Levy, O., Manning, C. D., "What Does BERT Look At? An Analysis of BERT's Attention", Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, 2019.