



Extending ROMANSETU to Diverse Languages

Group 35

Anusha Pant, Nav Sanya Anand, Paul Kurian, Sebastian Escalante, and Shreyas Malewar





Motivation and Problem Definition

Low-resource languages

Large language models (LLMs) frequently encounter difficulties when processing languages other than English like Hindi or Korean

Inadequate training data


There is inadequate training data available for low-resource languages

Challenges with non-latin scripts

Processing and understanding non-latin scripts is challenging given different character sets, and grammatical structures.

ROMANSETU by Husain et al.

ROMANSETU demonstrated an improvement in performance for LLM tasks performed on Hindi text.



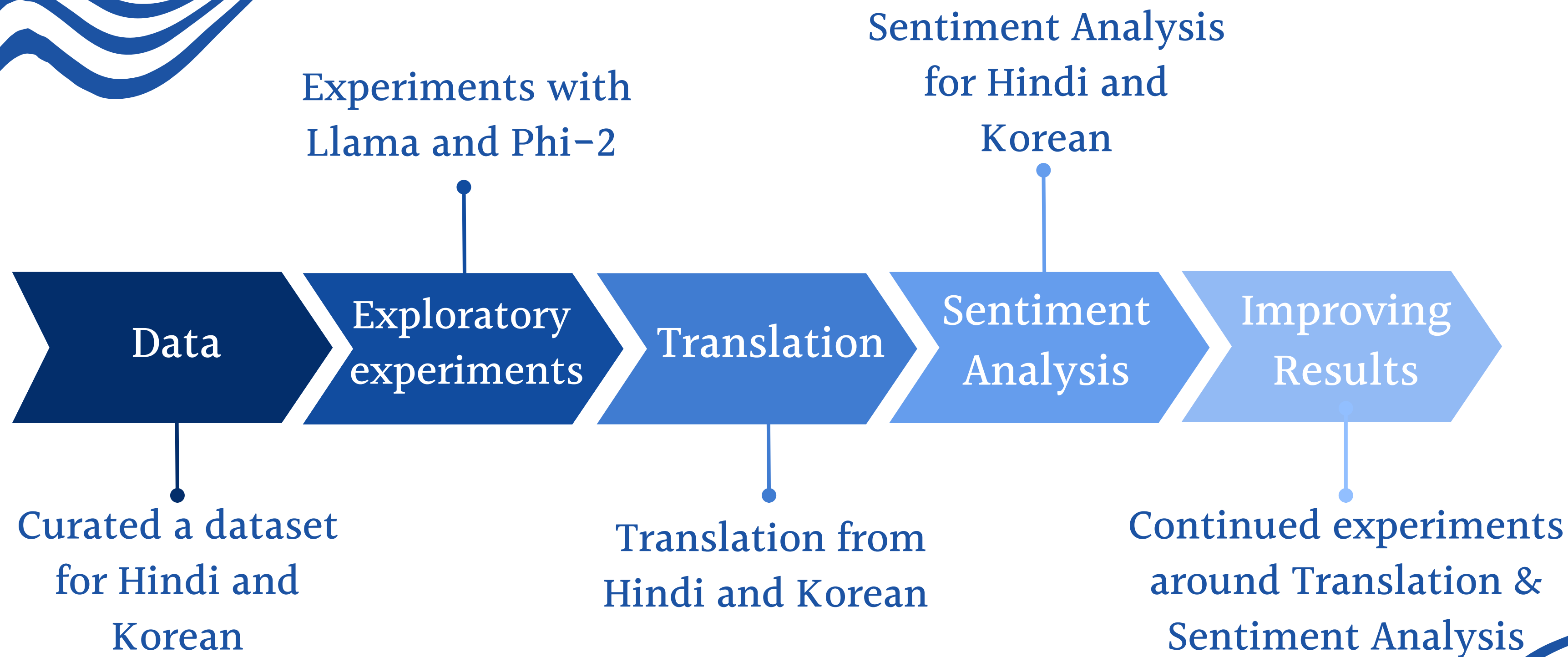
Impact of ROMANSETU

नमस्ते
↓
namaste

- The authors propose fine-tuning LLMs with romanized Hindi
- The tasks performed were machine translation from Hindi to English and sentiment analysis on Hindi
- Romanization significantly improved the performance and inference efficiency of LLaMA-7B on these tasks
- Can this improvement be replicated for other non-Latin, low-resource languages?

Solution





Dataset

Hindi

- For Translation we use the English_Hindi_Dataset* containing 19239 sentences.
- For sentiment analysis we use the Hindi Language sentiment dataset** containing 9077 reviews with negative, positive, and neutral rating.

* Preet Viradiya (2022) - Kaggle

** Mahesh M (2021) - Kaggle

Korean

- For Translation we use the Korean - English Parallel Corpus* containing 4564 sentences.
- For sentiment analysis we use the KR3: Korean Restaurant Reviews with Ratings** dataset containing 1236 reviews with negative, positive, and neutral rating.

* Yeejoon Lee et al (2022) - Kaggle

** Ramzel Renz Logo (2020) - Kaggle

Romanization Tools

IndiXlit for Hindi:

- Utilizes the IndiXlit toolkit for transliterating Indic scripts, including Hindi, into other writing systems.
- Romanization Process:
 - Iterates through each Hindi sentence using the XlitEngine library to generate its Romanized version.

Revised Romanization for Korean:

- Refers to a specific system for converting Hangul (the Korean alphabet) into the Roman script.
- Romanization Process:
 - Iterates through each Korean sentence and uses the Romanizer library for romanization.





Tiny-Llama 1.1B

Uses the same
architecture and
tokenizer as Llama 2
but has only 1.1 Billion
parameters and pre-
trained on 3 trillion
tokens



Results

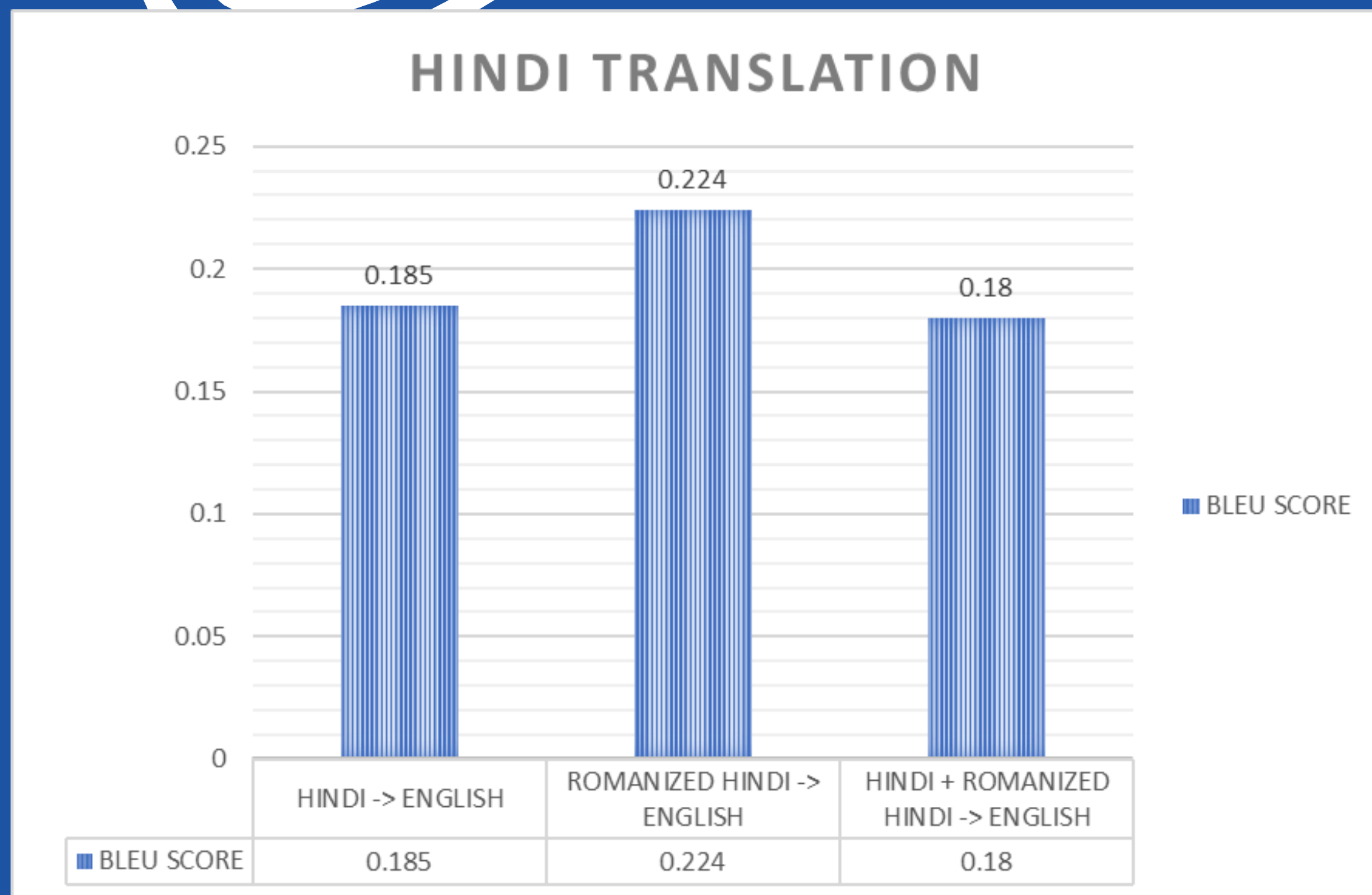


Translation

Hindi to English

Metric - BLEU Score

Romanized Hindi outperforms Hindi
by ≈ 0.04

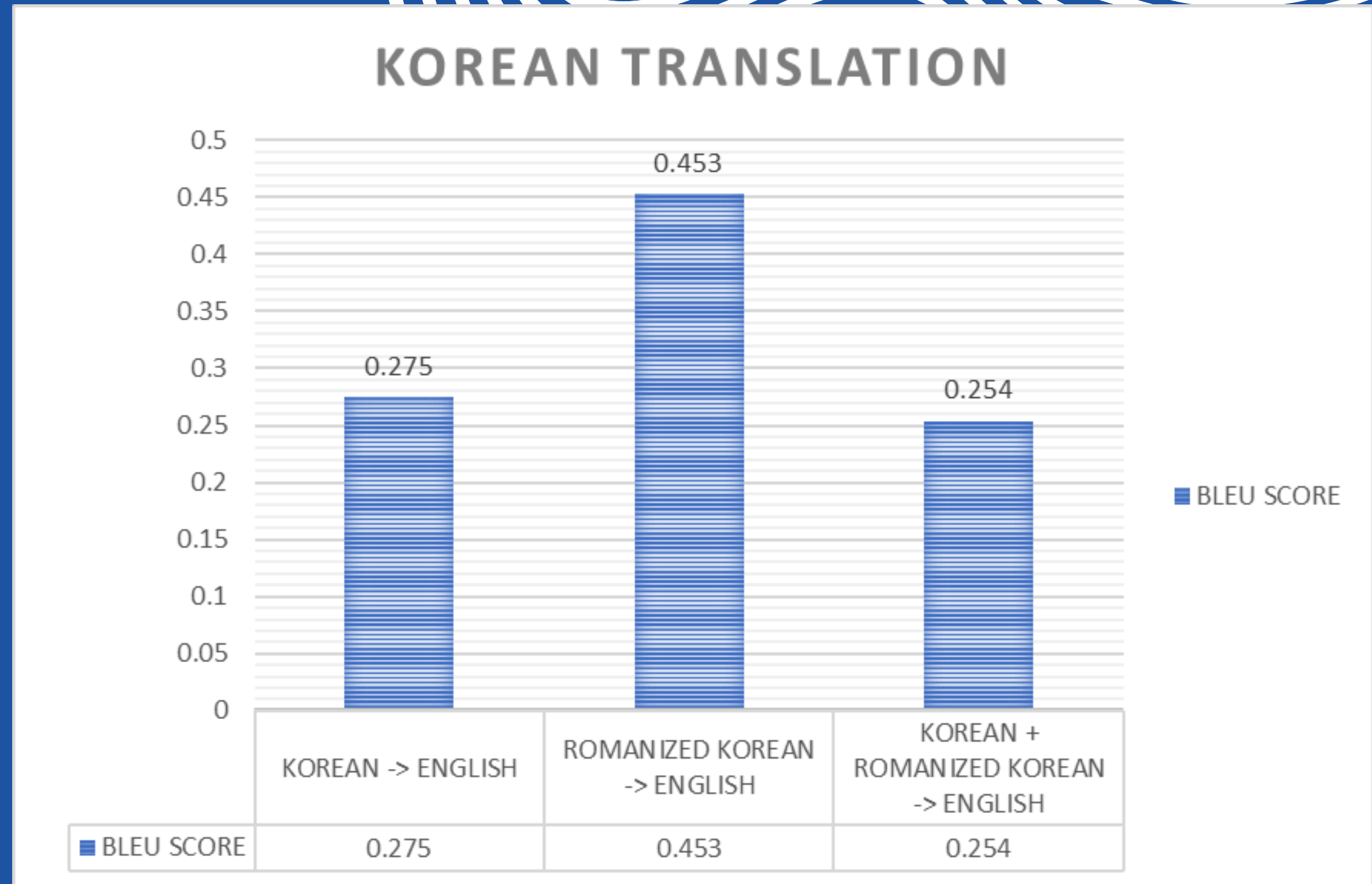


Translation

Korean to English

Metric - BLEU Score

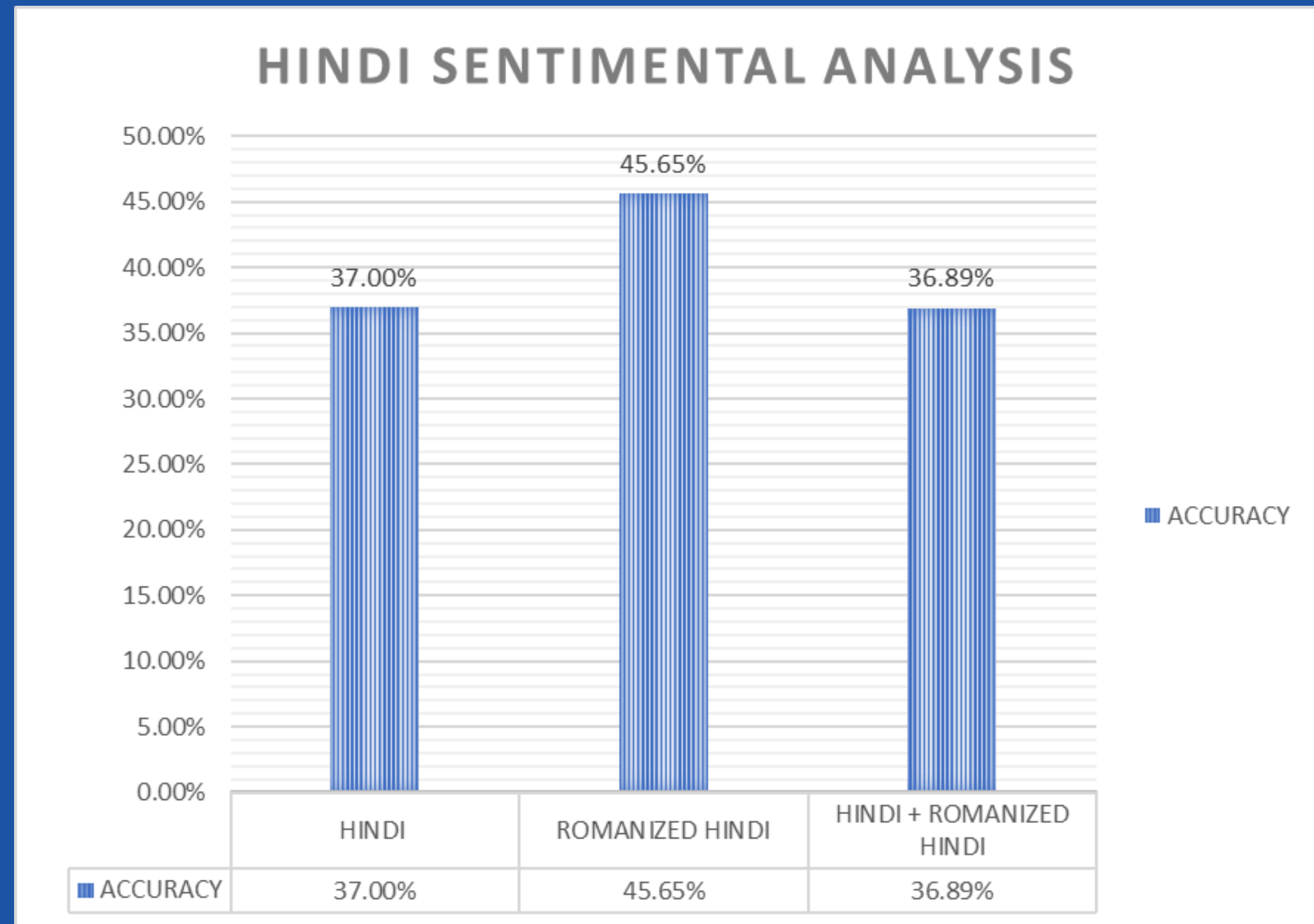
Romanized Korean also
outperforms Korean by ≈ 0.18



Sentiment Analysis Hindi

Metric - Accuracy

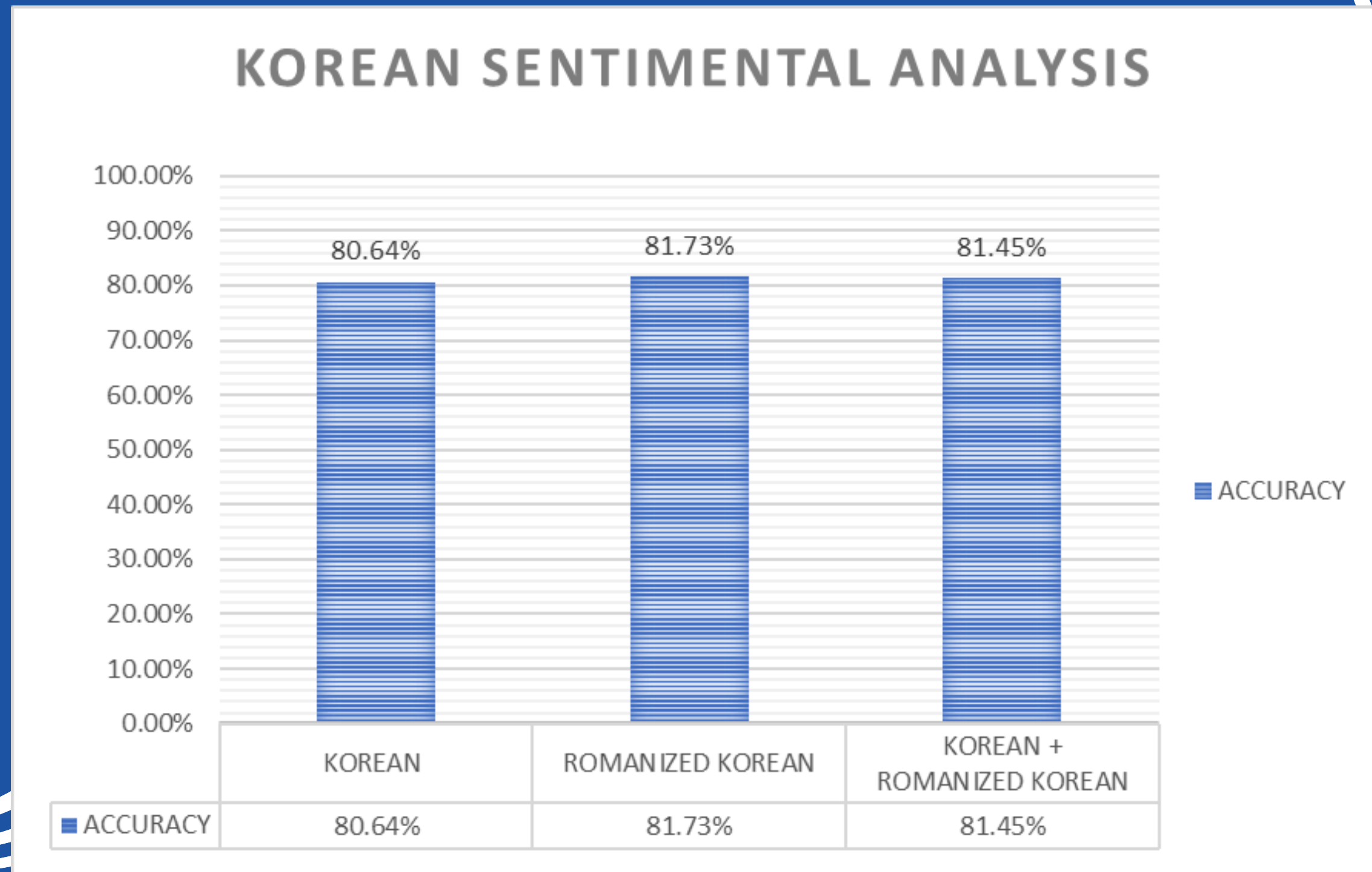
Romanized Hindi improves performance by 8.65%



Sentiment Analysis Korean

Metric - Accuracy

Romanized Korean outperforms
Korean by symbol $\approx 1\%$





Challenges and Future Work

Challenges

- Limited computational resources
- Small dataset

Future Work

- Experimentation with different hyperparameters to improve model results
- Exploring evaluation metrics like COMET scores.

Conclusion

- **Low rank adapters** can be, and are proven effective to fine-tune large language models.
- **For translation, entirely romanized prompts perform marginally better** than romanized + native script and native script. This may be due to encoding and tokenization differences.
- **Native +Romanized script performs worse than pure Romanized.** This might be due to the more complex input, and the model needed more time to learn.
- **For sentiment analysis, entirely romanized prompts perform better** than native and native + romanized inputs. This reinforces the notion that models trained on English perform better with romanized data.
- Even with limited computational resources, we have established that **romanization does have a positive impact on the performance of LLMs** on non-Latin languages.