Leveraging Machine-Generated Data for Joint Intent Detection and Slot Filling in Bangla: A Resource-Efficient Approach

A H M Rezaul Karim
George Mason University, VA, USA
akarim9@gmu.edu

Özlem Uzuner
George Mason University, VA, USA
ouzuner@gmu.edu

Abstract

Natural Language Understanding (NLU) is crucial for conversational AI, yet low-resource languages lag behind in essential tasks like intent detection and slot filling. To address this gap, we translated the widely-used English SNIPS dataset to Bangla using LLaMA 3, creating a dataset that captures the linguistic complexities of the language. With this translated dataset, we compared both independent and joint modeling approaches using transformer architecture. Results demonstrate that a joint approach based on multilingual BERT (mBERT) achieves superior performance, with **97.83%** intent accuracy and **91.03%** F1 score for slot filling. This work advances NLU for Bangla and provides insights for developing robust models in other low-resource languages. [1]

1 Introduction

Natural Language Understanding (NLU) is an important part of artificial intelligence (AI), powering applications from home assistants to conversational agents, text analysis, and language translation (Vanzo et al., 2019; Liu et al., 2021; Carvalho et al., 2019; Bender and Koller, 2020; Stahlberg, 2020; Bast et al., 2016). Although we see a significant advance in languages with abundant resources, low- to medium-resource languages face substantial challenges in NLU development. The Bangla language is spoken by almost 280 million people and still remains notably underrepresented in this domain (Ethnologue, 2024). The rich morphology, complex sentence structure, and compound characters of Bangla make it challenging for NLU tasks.

Intent detection and slot filling represent core NLU tasks that are important for building effective language understanding systems. Intent detection identifies the user's purpose while slot filling extracts specific details such as time, location, or quantity. If a user says, *"What's the weather in New York tomorrow afternoon?"* intent detection identifies the goal as *"GetWeather,"* and slot filling pulls out details like *location* ("New York") and *time* ("tomorrow afternoon"). According to the findings of Grishman and Sundheim, these tasks share similarities with Named Entity Recognition (NER) in extracting structured information from text but they go beyond entity identification by requiring the system to understand the user's goal and dynamically extract task-specific details. These tasks have been extensively studied for English (Weld et al., 2022; Niu et al., 2019; Qin et al., 2021; Liu and Lane, 2016; Goo et al., 2018), and some progress has been made for several low-resource languages, including Bangla (Dao et al., 2021; Akbari et al., 2023; Stoica et al., 2021; Sakib et al., 2023). Although there are a few prominent studies on Bangla NLU (Bhattacharjee et al., 2021; Hossain et al., 2020; Alam et al., 2021) research remains limited, mainly due to the lack of large annotated datasets. Tackling this data deficiency is essential for expanding the representation of Bangla in AI and improving NLU systems for diverse languages.

Our work focuses on two primary objectives:

1. We develop a high-quality Bangla NLU dataset using English-to-Bangla translation models and Large Language Models (LLM). This work demonstrates how automated methods can effectively generate resources for underrepresented languages.

2. We evaluate separate and joint modeling for Bangla intent detection and slot filling tasks. Our evaluation compares these approaches with established methods from English NLU research.

Through these objectives, our aim is to establish a foundation for NLU systems in Bangla while providing insight that can benefit other underrepresented languages.

[1] The dataset and the code can be found here: https://github.com/AHMRezaul/Joint_IDSF_Bangla.

Proceedings of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL 2025), pages 208–216
January 19, 2025. ©2025 International Committee on Computational Linguistics

2 Related Work

Conversational AI has advanced intent detection and slot filling. Early models like Hidden Markov Models and Conditional Random Fields (Bhargava et al., 2013; Shen et al., 2011), treated these tasks separately, limiting generalization capabilities. The introduction of Recurrent Neural Networks, mainly Long Short-Term Memory networks, improved performance by modeling language sequences (Mesnil et al., 2013; Sreelakshmi et al., 2018).

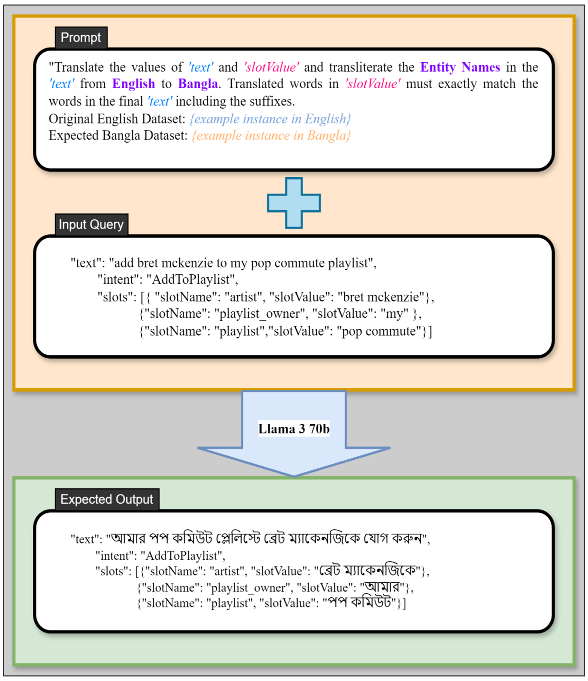
After recognizing the dependency between intent detection and slot filling tasks, joint modeling was adopted (Zhang et al., 2018; Weld et al., 2022; Qin et al., 2021). The slot-gated model (Goo et al., 2018) advanced this approach by using intent predictions to guide slot filling. JointBERT (Chen et al., 2019) further improved performance through transformer-based joint optimization. While effective in resource-rich languages, applying them to low- and medium-resource languages such as Bangla has been challenging due to limited datasets (Sakib et al., 2023). Efforts to address this gap included translating English datasets into languages such as Vietnamese, Persian, and Romanian (Dao et al., 2021; Akbari et al., 2023; Stoica et al., 2021) and applying the established NLU methodologies.

Recent advances in machine translation (MT) models such as Multilingual T5 (Xue et al., 2020), XLM-ProphetNet (Qi et al., 2021), and BanglaT5 (Bhattacharjee et al., 2023; De bruyn et al., 2022) offer promising solutions for translating benchmark datasets to low- to medium-resource languages. Additionally, LLMs including Mistral (Jiang et al., 2023), LLaMA 2 (Touvron et al., 2023), LLaMA 3 (Meta, 2024), GPT-3.5 (Brown et al., 2020), and GPT-4 (Achiam et al., 2023) show potential for generating datasets in resource-scarce languages (Xu et al., 2023; Mahfuz et al., 2024).

Our work translates a benchmark English dataset to Bangla using traditional MT techniques and LLMs, showing that LLMs excel in capturing context and generating quality data. We use this translated dataset to develop and evaluate a Bangla model for intent detection and slot filling, outperforming previous efforts.

3 Methodology

This section provides a comprehensive overview of the dataset generation process and the models implemented in this project.



"Translate the values of *'text'* and *'slotValue'* and transliterate the **Entity Names** in the *'text'* from **English** to **Bangla**. Translated words in *'slotValue'* must exactly match the words in the final *'text'* including the suffixes.
Original English Dataset: *{example instance in English}*
Expected Bangla Dataset: *{example instance in Bangla}*

```
"text": "add bret mckenzie to my pop commute playlist",
     "intent": "AddToPlaylist",
     "slots": [{ "slotName": "artist", "slotValue": "bret mckenzie"},
             {"slotName": "playlist_owner", "slotValue": "my" },
             {"slotName": "playlist","slotValue": "pop commute"}]
```

```
"text": "আমার পপ কমিউট প্লেলিস্টে ব্রেট ম্যাকেনজিকে যোগ করুন",
     "intent": "AddToPlaylist",
     "slots": [{"slotName": "artist", "slotValue": "ব্রেট ম্যাকেনজিকে"},
             {"slotName": "playlist_owner", "slotValue": "আমার"},
             {"slotName": "playlist", "slotValue": "পপ কমিউট"}]
```

Figure 1: *A few-shot prompting approach with the input query and the expected output for the LLaMA 3 model.*

3.1 Dataset

3.1.1 SNIPS Dataset

We used the English SNIPS dataset (Coucke et al., 2018) to generate the Bangla dataset. SNIPS is a popular dataset for training and testing NLU models, especially for tasks like intent detection and slot filling. It consists of 13,084 training, 700 testing, and 700 validation samples, covering 7 intent classes and 72 slot values. Each intent and slot is carefully labeled to cover a large spectrum of user interactions, making it a useful resource for developing language models.

3.1.2 Dataset generation

The dataset generation process was completed in two steps: (1) machine translation for the Training and Validation sets, and (2) manual translation for the Test set. This combined approach ensured resource efficiency while maintaining high accuracy for evaluating real-world performance.

A. Training and Validation sets:
For Training and Validation sets, various methods were explored to generate the Bangla dataset from the English SNIPS dataset.

BanglaT5 model: Initially, the **BanglaT5** model (Bhattacharjee et al., 2023) was chosen for machine translation due to its high BLEU score compared to other English-to-Bangla models.

However, it struggled with entity names (e.g., Artist, Location, Movie, Song), translating them instead of transliterating, which altered the sentence meaning and made the results unusable. To address this, we applied the **BNTRANSLIT** model (Sarkar, 2021), designed for English-to-Bangla transliteration. We transliterated entity names before translating the sentences. Unfortunately, this approach also produced suboptimal results, as the overall translation quality remained insufficient.

LLaMA-3: Finally, the **LLaMA-3-70B-Instruct** model (Meta, 2024) was employed using a carefully crafted prompt that transliterated entity names while translating the rest of the sentence. We adopted a few-shot approach, providing five examples from the original English dataset along with their manually translated counterparts. Figure 1 shows the prompt, specifying that entity names should be transliterated, and the rest of the sentence translated into Bangla. After refining the prompt, the model delivered highly accurate translations, nearing manual translation quality. However, minor issues persisted, such as untranslated English words and extraneous information adding noise to the dataset. These issues were resolved during post-processing through automated rule-based methods, identifying and removing irrelevant content and correcting mismatches between slot values and translated text. The results were then manually verified to ensure the accuracy and consistency of the final dataset. The final dataset was annotated using the Beginning-Inside-Outside (BIO) notation.

B. Test set:

The test set was manually translated and annotated to ensure accuracy when evaluating real-world performance. Four doctoral students fluent in English and Bangla participated in this process. Initially, two annotators translated the English SNIPS test set into Bangla and annotated slot values using the BIO format. To ensure consistency, two annotators independently annotated 10% of the samples and discussed their results to agree on a unified annotation method. They then applied this agreed-upon method to annotate the remaining 90% of the dataset, achieving a **0.83 Cohen's Kappa** score A.2 for the entire dataset. Following this, two additional independent reviewers conducted sequential reviews of the entire dataset, further enhancing its quality by identifying and removing any remaining errors or biases.

Dataset	Stat.	Train	Valid.	Test
SNIPS (English)	Intents	13084	700	700
	Slots	60412	3221	3276
Generated (Bangla)	Intents	12850	685	694
	Slots	54747	2865	3105

Table 1: *Comparison of data distribution between generated Bangla dataset and the original English SNIPS dataset.*

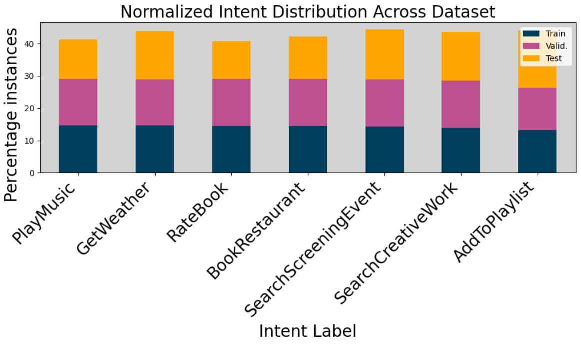


Figure 2: *A normalized distribution of Intent classes in the generated Bangla train, validation, and test sets.*

3.1.3 Dataset Analysis

The generated Bangla dataset contains 7 intent classes but 80 (vs. 72 in English) unique slot labels. The increase in the number of slot labels is due to single-word slot values in English often translating into multi-word slot values in Bangla. As a result, many slots that previously required only a beginning (B) tag in the English dataset now require both beginning (B) and inside (I) tags in the Bangla dataset. Table 1 compares the number of intent and slot instances between the original English and generated Bangla datasets.

The Bangla Train and Validation sets have slightly fewer instances of intents and slots than the English version as seen in Table 1, primarily due to post-processing after the LLaMA 3 translation. Instances with errors that could not be easily fixed were excluded to maintain the integrity of the machine-translated corpus, as the focus was on assessing LLM performance without manual intervention.

The slight reduction in the number of slot values can be attributed to two main factors:

Alignment Issues: Some slot values failed to align with the translated text, even after post-processing, resulting in unannotated slots.

Linguistic Differences: In Bangla, certain multi-word slot values in English are condensed into single or fewer words, causing a reduction in the number of slot values compared to the English dataset.

Figure 2 shows a balanced distribution of intent classes (normalized for better visualization) across the training, validation, and test sets, reducing bias. However, there is an uneven distribution of slot labels demonstrated in figure 3, with rare slots potentially challenging the model's prediction accuracy as discussed in the appendix A.3.

Overall, the dataset effectively supports training and evaluation for diverse intents and slot labels.

3.2 Models

We evaluated three transformer-based models on our Bangla dataset: BERT Base (baseline), Multilingual BERT (mBERT), and Bangla BERT (Devlin, 2018; Bhattacharjee et al., 2021). Bangla BERT handles Bangla-specific processing, while mBERT offers multilingual adaptability. Both separate and joint training approaches were tested, following the benchmarking methodology of the English SNIPS dataset. Detailed specifications are in Appendix A.1.

4 Experiments and Analysis

The Bangla dataset was used to fine-tune the models with optimized hyperparameters: batch size of 32 for training and 64 for evaluation, maximum sequence length of 160, learning rate of 5e-5, and dropout rate of 0.1 gave the best performance.

4.1 Training details

We divided the experiments into two parts for each of the BERT variants (BERT Base, mBERT and Bangla BERT): (1) separate fine-tuning for intent detection and slot filling, and (2) joint fine-tuning using different BERT variants as the backbone. For the joint setup, we adopted the JointBERT configuration (Chen et al., 2019) with the mentioned hyperparameters and applied similar settings to the separate models. Models were trained across varying epochs $[1, 5, 10, 20, 30, 40]$, and the best performances from these runs were recorded.

4.2 Result and Discussion

Table 2 presents intent detection accuracy and slot filling F1 score at token level. It also illustrates the accuracy on a sentence level for the joint approach; this metric measures the percentage of instances where both intent class and slot labels were correctly predicted. The results clearly indicate that a joint approach outperforms the separate approaches. Notably, the multilingual BERT (mBERT) model surpasses Bangla BERT, a model specifically pre-trained in Bangla, in both joint and separate task settings.

Model		Intent Accuracy	Slot F1	Sentence Accuracy
Separate Training	BERT Base	95.53	86.13	-
	mBERT	**96.97**	**90.64**	-
	Bangla BERT	95.96	89.96	-
JointBERT	BERT Base	96.97	84.83	69.30
	mBERT	**97.83**	**91.03**	**79.39**
	Bangla BERT	97.69	89.42	76.65

Table 2: *Results for intent detection and slot filling tasks (%). Best scores for separate and joint models are bolded, with the overall best score underlined.*

This outcome suggests that mBERT's pre-training on a diverse multilingual corpus enables it to generalize effectively across languages, providing an advantage when dealing with the complexities of the Bangla language. Although Bangla BERT has shown superior performance in downstream tasks like sentiment analysis and hate speech detection (Sarker, 2021), mBERT outperforms it in the slot filling task, which is closely associated with Named Entity Recognition (NER) (Grishman and Sundheim, 1996). This is consistent with previous research, where mBERT outperformed Bangla BERT in the Bengali NER task using the *'Bengali NER'* dataset (Rahimi et al., 2019). The broad linguistic knowledge in pre-training of mBERT appears to offer an advantage in tasks that rely on accurate entity recognition.

Additionally, we observe the highest sentence-level accuracy with mBERT. This measures how often both the intent class and all slot labels are predicted accurately. This metric provides a holistic view of the model's performance.

Appendix A.4 presents a detailed error analysis of the best-performing joint model, highlighting common errors and identifying areas for potential improvement.

5 Conclusion

This study offers a comprehensive evaluation of joint intent detection and slot filling for Bangla, a resource-scarce language. To overcome the lack of available data, we generated a Bangla dataset from the benchmark English SNIPS dataset using the LLaMA 3 model and applied well-established NLU methodologies. Using a manually curated test set, we confirmed that joint modeling outperformed separate approaches, with the mBERT variant achieving better results than the language-specific Bangla BERT.

Our research also highlights the potential of LLMs in generating training data for low- to medium-resource languages. By leveraging existing benchmark datasets, LLMs can produce datasets that are effective for real-world applications. This approach provides a scalable solution for training high-performing models.

6 Limitations

We manually translated and annotated the SNIPS test set. However, we encountered resource constraints that limited our ability to manually curate the entire dataset. So, we relied on LLaMA 3 to generate training and validation data, utilizing its machine translation and entity recognition capabilities. While we recognize that a manually curated dataset would likely result in better fine-tuning and improved model performance, the resource limitations made machine translation a more practical and feasible option for this study. This experience also suggests that LLM-generated datasets can effectively support model fine-tuning for specific tasks.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Masoud Akbari, Amir Hossein Karimi, Tayyebeh Saeedi, Zeinab Saeidi, Kiana Ghezelbash, Fatemeh Shamsezat, Mohammad Akbari, and Ali Mohades. 2023. A persian benchmark for joint intent detection and slot filling. *Preprint*, arXiv:2303.00408.

Firoj Alam, Arid Hasan, Tanvirul Alam, Akib Khan, Janntatul Tajrin, Naira Khan, and Shammur Absar Chowdhury. 2021. A review of bangla natural language processing tasks and the utility of transformer models. *arXiv preprint arXiv:2107.03844*.

Hannah Bast, Björn Buchhold, Elmar Haussmann, et al. 2016. Semantic search on text and knowledge bases. *Foundations and Trends® in Information Retrieval*, 10(2-3):119–271.

Emily M Bender and Alexander Koller. 2020. Climbing towards nlu: On meaning, form, and understanding in the age of data. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 5185–5198.

Aditya Bhargava, Asli Celikyilmaz, Dilek Hakkani-Tür, and Ruhi Sarikaya. 2013. Easy contextual intent prediction and slot detection. In *2013 ieee international conference on acoustics, speech and signal processing*, pages 8337–8341. IEEE.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Ahmad, and Rifat Shahriyar. 2023. Banglanlg and banglat5: Benchmarks and resources for evaluating low-resource natural language generation in bangla. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 714–723.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Uddin Ahmad, Kazi Samin, Md Saiful Islam, Anindya Iqbal, M Sohel Rahman, and Rifat Shahriyar. 2021. Banglabert: Language model pretraining and benchmarks for low-resource language understanding evaluation in bangla. *arXiv preprint arXiv:2101.00204*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *Preprint*, arXiv:2005.14165.

Arthur Carvalho, Adam Levitt, Seth Levitt, Edward Khaddam, and John Benamati. 2019. Off-the-shelf artificial intelligence technologies for sentiment and emotion analysis: a tutorial on using ibm natural language processing. *Communications of the Association for Information Systems*, 44(1):43.

Qian Chen, Zhu Zhuo, and Wen Wang. 2019. Bert for joint intent classification and slot filling. *Preprint*, arXiv:1902.10909.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, et al. 2018. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv:1805.10190*.

Mai Hoang Dao, Thinh Hung Truong, and Dat Quoc Nguyen. 2021. Intent detection and slot filling for vietnamese. *arXiv preprint arXiv:2104.02021*.

Maxime De bruyn, Ehsan Lotfi, Jeska Buhmann, and Walter Daelemans. 2022. Machine translation for multilingual intent detection and slots filling. In *Proceedings of the Massively Multilingual Natural Language Understanding Workshop (MMNLU-22)*, pages 69–82, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Ethnologue. 2024. Ethnologue 200: Languages of the world. Accessed on April 4, 2024.

Chih-Wen Goo, Guang Gao, Yun-Kai Hsu, Chih-Li Huo, Tsung-Chieh Chen, Keng-Wei Hsu, and Yun-Nung Chen. 2018. Slot-gated modeling for joint slot filling and intent prediction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 753–757.

Ralph Grishman and Beth M Sundheim. 1996. Message understanding conference-6: A brief history. In *COLING 1996 volume 1: The 16th international conference on computational linguistics*.

Md Zobaer Hossain, Md Ashraful Rahman, Md Saiful Islam, and Sudipta Kar. 2020. Banfakenews: A dataset for detecting fake news in bangla. *Preprint*, arXiv:2004.08789.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Bing Liu and Ian Lane. 2016. Attention-based recurrent neural network models for joint intent detection and slot filling. *arXiv preprint arXiv:1609.01454*.

Xingkun Liu, Arash Eshghi, Pawel Swietojanski, and Verena Rieser. 2021. Benchmarking natural language understanding services for building conversational agents. In *Increasing Naturalness and Flexibility in Spoken Dialogue Interaction: 10th International Workshop on Spoken Dialogue Systems*, pages 165–183. Springer.

Tamzeed Mahfuz, Satak Kumar Dey, Ruwad Naswan, Hasnaen Adil, Khondker Salman Sayeed, and Haz Sameen Shahgir. 2024. Too late to train, too early to use? a study on necessity and viability of low-resource bengali llms. *arXiv preprint arXiv:2407.00416*.

Grégoire Mesnil, Xiaodong He, Li Deng, and Yoshua Bengio. 2013. Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding. In *Interspeech*, pages 3771–3775.

Meta. 2024. Llama 3.

Peiqing Niu, Zhongfu Chen, Meina Song, et al. 2019. A novel bi-directional interrelated model for joint intent detection and slot filling. *arXiv preprint arXiv:1907.00390*.

Weizhen Qi, Yeyun Gong, Yu Yan, Can Xu, Bolun Yao, Bartuer Zhou, Biao Cheng, Daxin Jiang, Jiusheng Chen, Ruofei Zhang, et al. 2021. Prophetnet-x: Large-scale pre-training models for english, chinese, multi-lingual, dialog, and code generation. *arXiv preprint arXiv:2104.08006*.

Libo Qin, Tailu Liu, Wanxiang Che, Bingbing Kang, Sendong Zhao, and Ting Liu. 2021. A co-interactive transformer for joint slot filling and intent detection. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8193–8197. IEEE.

Afshin Rahimi, Yuan Li, and Trevor Cohn. 2019. Massively multilingual transfer for NER. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 151–164, Florence, Italy. Association for Computational Linguistics.

Fardin Ahsan Sakib, AHM Karim, Saadat Hasan Khan, and Md Mushfiqur Rahman. 2023. Intent detection and slot filling for home assistants: Dataset and analysis for bangla and sylheti. *arXiv preprint arXiv:2310.10935*.

Sagor Sarkar. 2021. Bntranslit.

Sagor Sarker. 2021. Evaluation of bangla-bert on classification task. GitHub repository.

Yelong Shen, Jun Yan, Shuicheng Yan, Lei Ji, Ning Liu, and Zheng Chen. 2011. Sparse hidden-dynamics conditional random fields for user intent understanding. In *Proceedings of the 20th international conference on World wide web*, pages 7–16.

K Sreelakshmi, PC Rafeeque, S Sreetha, and ES Gayathri. 2018. Deep bi-directional lstm network for query intent detection. *Procedia computer science*, 143:939–946.

Felix Stahlberg. 2020. Neural machine translation: A review. *Journal of Artificial Intelligence Research*, 69:343–418.

Anda Stoica, Tibor Kadar, Camelia Lemnaru, Rodica Potolea, and Mihaela Dînşoreanu. 2021. Intent detection and slot filling with capsule net architectures for a romanian home assistant. *Sensors*, 21(4):1230.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Andrea Vanzo, Emanuele Bastianelli, and Oliver Lemon. 2019. Hierarchical multi-task natural language understanding for cross-domain conversational ai: Hermit nlu. *arXiv preprint arXiv:1910.00912*.

A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*.

Henry Weld, Xiaoqi Huang, Siqu Long, Josiah Poon, and Soyeon Caren Han. 2022. A survey of joint intent detection and slot filling models in natural language understanding. *ACM Computing Surveys*, 55(8):1–38.

Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2023. A paradigm shift in machine translation: Boosting translation performance of large language models. *arXiv preprint arXiv:2309.11674*.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2020. mt5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint*.

Chenwei Zhang, Yaliang Li, Nan Du, Wei Fan, and Philip S Yu. 2018. Joint slot filling and intent detection via capsule neural networks. *arXiv preprint arXiv:1812.09471*.

A Appendix

A.1 Implemented Models

A.1.1 BERT and its Variants

BERT (Bidirectional Encoder Representations from Transformers) (Devlin, 2018) revolutionized NLP by using deep bidirectional representations and self-attention mechanisms (Vaswani, 2017). We utilized three key BERT variants: BERT Base, which is trained on lower-cased English text, ideal for tasks where case sensitivity is less critical; Multilingual BERT (mBERT), trained on over 100 languages, making it suitable for cross-lingual tasks; and Bangla BERT (Bhattacharjee et al., 2021), specifically trained on Bangla text, making it more effective at handling the unique linguistic and cultural aspects of Bangla. [2].

These models were chosen to evaluate performance on Bangla language tasks. BERT Base was used to assess how the base English model, which established the original benchmark on the English SNIPS dataset, performs on Bangla data and to measure improvements with other variants. mBERT provided insights into cross-lingual transfer learning, while Bangla BERT leveraged its Bangla-specific training to address linguistic nuances.

A.1.2 JointBERT Modeling

The JointBERT model (Chen et al., 2019) combines intent detection and slot filling into a single architecture using a BERT backbone. It classifies intent based on the $[CLS]$ token and assigns slot labels to each token in the input sequence. By jointly modeling both tasks, JointBERT enhances contextual understanding and improves accuracy in both intent detection and slot filling, making it highly suitable for conversational AI tasks in the Bangla language.

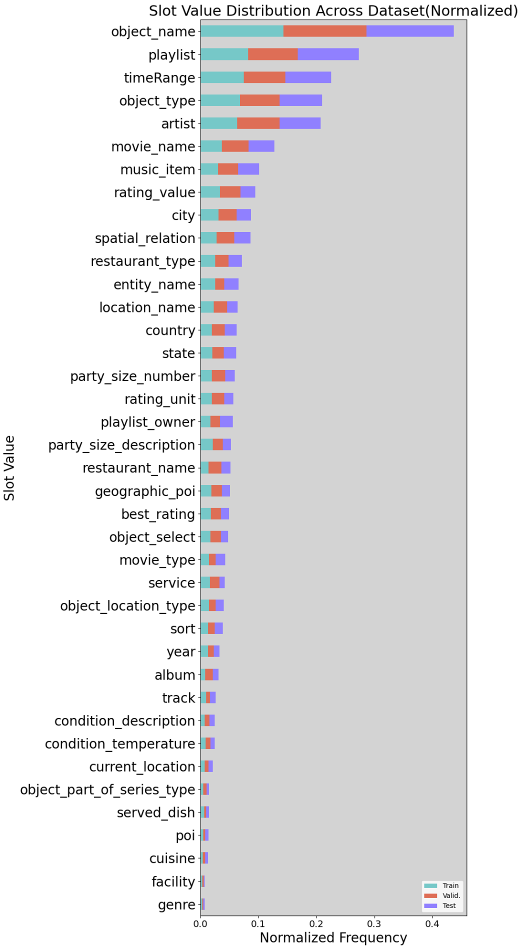
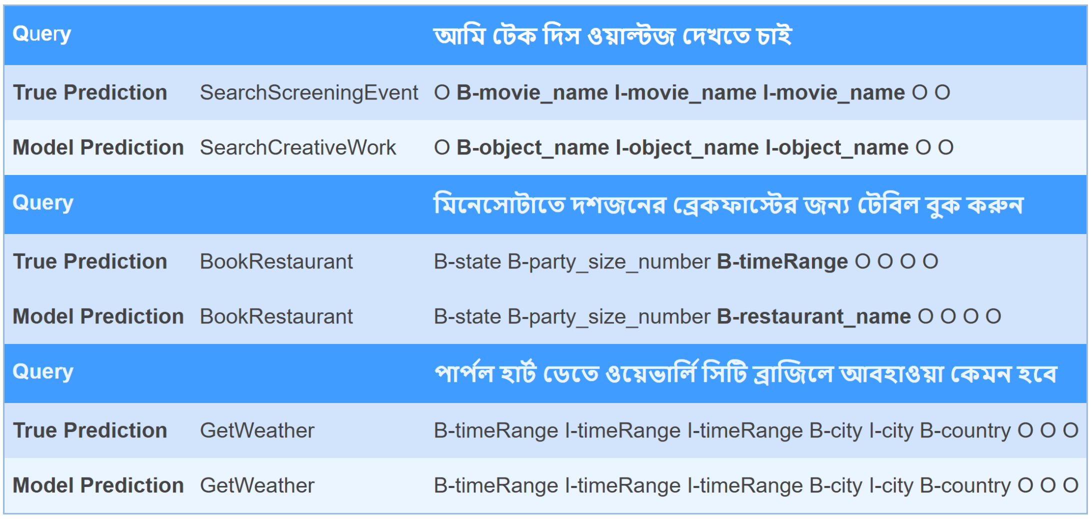


Figure 3: *A normalized distribution of different slot labels across the train, valid. and test sets demonstrate an imbalance of different slot labels.*

A.2 Inter-annotator Agreement

In this research, Cohen's Kappa (Cohen, 1960) was used as a key metric to assess inter-annotator agreement, ensuring the quality and reliability of the manually translated and annotated test set. Cohen's Kappa assesses the level of agreement among annotators, considering the likelihood of agreements occurring by chance. A score of 1 signifies complete agreement, whereas 0 indicates no more agreement than what might be anticipated by chance. In this instance, **0.83** Cohen's Kappa score indicates a high level of agreement between the annotators,

[2] Huggingface BERT Base, mBERT, Bangla BERT



Query		**আমি টেক দিস ওয়াল্টজ দেখতে চাই**
True Prediction	SearchScreeningEvent	O **B-movie_name I-movie_name I-movie_name** O O
Model Prediction	SearchCreativeWork	O **B-object_name I-object_name I-object_name** O O
Query		**মিনেসোটাতে দশজনের ব্রেকফাস্টের জন্য টেবিল বুক করুন**
True Prediction	BookRestaurant	B-state B-party_size_number **B-timeRange** O O O O
Model Prediction	BookRestaurant	B-state B-party_size_number **B-restaurant_name** O O O O
Query		**পার্পল হার্ট ডেভে ওয়েভার্লি সিটি ব্রাজিলে আবহাওয়া কেমন হবে**
True Prediction	GetWeather	B-timeRange I-timeRange I-timeRange B-city I-city B-country O O O
Model Prediction	GetWeather	B-timeRange I-timeRange I-timeRange B-city I-city B-country O O O

Figure 4: *Instances of predicted intent class and the slot labels by the JointBERT(mBERT) model compared with the true predictions. 1) Both intent and slot value predictions are wrong, 2) Only a single slot value is incorrectly predicted, 3) Everything is predicted correctly.*

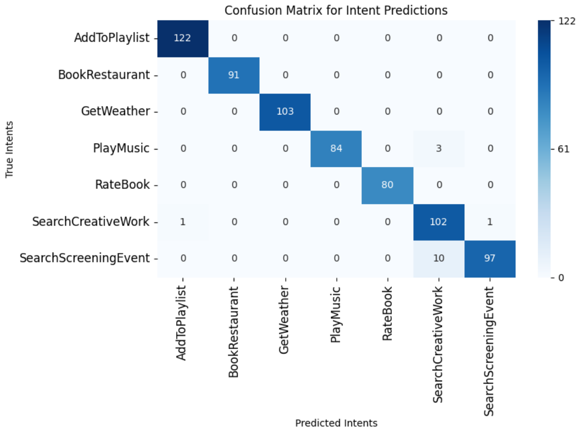


Figure 5: *Confusion matrix highlighting shared vocabulary-induced misclassifications of Intent classes by JointBERT model with mBERT.*

Error Type	**No. of errors**
Missing slot value in prediction (entirely or partly)	34
Predicted slot value matches an 'O' label	11
Predicted slot has correct label but incorrect boundary	23
Predicted slot has the correct boundary but incorrect label	70
Total errors	**138**

Table 3: *The number of different error types noticed for the JointBERT model with mBERT on the Test set.*

reflecting the consistency and reliability of the annotations.

Inter-annotator agreement is crucial for verifying the accuracy of translations and annotations in a dataset, especially when it involves subjective tasks like labeling slot values and intents. By applying this metric, we can ensure that different annotators interpret the data consistently, directly affecting the dataset's quality and the performance of models trained on it.

A.3 Distribution of Slot Labels

Figure 3 shows the normalized distribution of slot labels across the generated train, validation, and test sets. It is clear that the frequency of different slot labels varies significantly, which can introduce bias during fine-tuning. More frequent slot labels are likely to be predicted more often than less frequent ones. This bias is evident in a predicted instance shown in Figure 4, where the slot label *'movie_name'* is incorrectly labeled as *'object_name'*. The distribution indicates that *'object_name'* appears more frequently than *'movie_name'* across all datasets, which likely causes the model to favor the more frequent label. However, achieving a balanced dataset with an equal distribution of slot labels is difficult in the real world.

Although the model correctly identifies slot boundaries, it struggles to distinguish between labels, possibly because of the lack of semantic information about the entity, such as whether the entity is a movie name. Providing the model with this additional context could improve label accuracy.

A.4 Error Analysis

The confusion matrix for the JointBERT model using the mBERT variant in figure 5 shows a recurring pattern of confusion between the *‘SearchScreeningEvent’* and *‘SearchCreativeWork’* intents. This likely occurs because of the overlap in vocabulary across these intents, where terms related to screening events and creative works appear in similar contexts, leading to misclassification. An example instance of figure 4 also highlights this misclassification of intent classes.

Table 3 highlights the types of slot prediction errors. Out of 138 instances of incorrect slot predictions, about half involve the model correctly identifying the slot boundary, but mislabeling the slot values itself. These errors often occur in categories like *‘city’*, *‘country’* and *‘state’*, or between *‘movie_name’* and *‘object_name’*, and *‘track’* and *‘playlist’*. This can be because of the model’s reliance on recognizing patterns from its training phase without understanding the semantic meaning of an *entity name*. Additionally, the imbalance in slot label frequencies skews predictions towards more common slot labels, such as predicting *‘object_name’* instead of *‘movie_name’*.

Another instance of figure 4 shows that even though *‘timeRange* is a common slot label, the model still predicted it to be *‘restaurant_name*. This can be because the slot *‘restaurant_name’* appears more frequently with the other predicted slots from this instance than the *‘timeRange’* slot.

The second most common type of error involves the model missing certain slot values, especially those that have been transliterated. This can cause confusion regarding their semantic meaning. Additionally, the model sometimes predicts the correct slot label but struggles with boundary detection, particularly for multi-word slot values where part of the entity name is mistaken as a portion of the sentence. Lastly, some common words are incorrectly tagged as slot values due to their high frequency as a slot value in the training data, leading the model to incorrectly assign a label.