AI INBOUND CALLING AGENT

Project Advisor

Dr. Muhammad Saad Razzaq (Internal)

Project Manager

Dr. Muhammad Ilyas

Natural Language Understanding (NLU)

1. Introduction:

Natural Language Understanding (NLU) is a subfield of Natural Language Processing (NLP). It focuses on enabling machines to comprehend and interpret human language in a meaningful way. While NLP broadly involves the interaction between computers and human language, NLU specifically targets the understanding aspect, including the ability to recognize user intent, extract relevant information, and interpret context in natural language. Unlike simple keyword-based processing, NLU aims to capture the semantic, syntactic, and pragmatic meaning behind text or speech, enabling machines to respond intelligently and appropriately in dynamic conversational settings.

At its core, NLU performs two major functions:

1.1 Intent Detection:

The first and primary function of NLU is to understand what the user wants. This involves classifying a user's input into predefined categories known as *intents*. An intent represents the underlying goal or purpose behind an utterance. For instance, the sentence *"Admission kab start ho rahe hain?"* would be classified under the intent `admission_open_date`. Intent detection is a crucial step in conversational AI systems because it guides the system's response strategy and determines the subsequent actions or queries required to satisfy the user's request.

1.2 Entity Extraction:

The second key function involves extracting specific information or parameters from the user's input, referred to as *entities*. Entities are the critical pieces of information that provide context to the intent. For example, in the utterance *"BS programs ki fee kitni hoti hai?"* ,`program: BS` is an entity. By identifying entities, NLU allows systems to transform unstructured natural language into structured data that can be processed or acted upon programmatically.

2. Key Terminologies in NLU:

During the study of NLU, several foundational terminologies are crucial for understanding its operation:

- **Utterance:** A single instance of user input, whether spoken or typed, representing a discrete query or statement.
- **Intent:** The goal, objective, or action that the user wants to achieve through their utterance.
- **Entity:** Specific data points extracted from an utterance that provide details about the user's request.

These terminologies form the basis for building NLU models and designing conversational AI systems that can handle real-world user queries.

3. Types of Conversations in NLU:

NLU systems are designed to manage different types of conversational patterns. Based on existing literature and practical observations, conversations handled by NLU can generally be categorized into the following:

3.1 Topic Discovery:

This involves identifying the general domain or subject matter of a conversation. For example, a query about university courses would be classified under the "education" domain.

3.2 Intent Recognition:

After the domain is identified, intent recognition determines the precise action the user wants to perform within that domain. This is critical for mapping user queries to system responses or backend operations.

3.3 Conversation Switching:

Real-world conversations are rarely linear. Users often switch topics or intents mid-conversation. NLU systems must detect these shifts and adjust context dynamically to maintain a coherent dialogue, ensuring that responses remain relevant and context-aware.

4. Approaches to NLU:

Several approaches to NLU are:

4.1 Rule-Based Approaches:

Early NLU systems relied on manually defined rules and patterns to map user input to intents and entities. While straightforward, these systems are inflexible and cannot easily handle variations in unforeseen queries.

4.2 Framework-Based Approaches (e.g., Rasa):

Open-source frameworks like Rasa provide pipelines for intent classification and entity extraction using machine learning techniques. They allow combining rule-based and statistical methods for improved performance.

4.3 Transformer-Based Language Models:

Modern NLU systems leverage deep learning, particularly transformer architectures.

- **mBERT (Multilingual BERT):** Supports multiple languages and can perform cross-lingual tasks.
- **XLM-RoBERTa:** Optimized for cross-lingual understanding, especially effective for low-resource languages and code-mixed text.

Studies indicate that XLM-RoBERTa outperforms mBERT for languages like Urdu and for datasets containing Roman script variations.

Approach	Description	Advantages	Limitations
Rule-Based	Uses manually defined rules and patterns for intents and entities	Simple, interpretable, no training data needed	Cannot handle variations or unseen queries
Framework-Based (Rasa)	Combines ML and rule-based methods; provides pipelines for NLU	Flexible, customizable, handles some variations	Requires dataset for training, setup complexity
Transformer-Based (XLM-R)	Uses pretrained multilingual transformers; fine-tuned for tasks	Handles context, multilingual data, flexible	Computationally expensive, requires labeled data for fine-tuning

5. NLU Dataset Considerations:

A key component of NLU is access to high-quality data. Studying NLU revealed the importance of creating domain-specific datasets:

- **Intents:** Selecting relevant user goals (e.g., admission-related queries).
- **Utterances:** Providing multiple examples of user input for each intent.

- **Bilingual Data:** Including both English and Roman Urdu utterances improves system performance for multilingual users.
- **Structured Format:** Representing data in CSV or JSON format for model training and evaluation.

6. Transformer-Based Language Models in NLU:

Modern NLU systems increasingly rely on transformer-based language models for their ability to capture context, manage multilingual data, and handle complex sentence structures. **XLM-RoBERTa (XLM-R)** is a multilingual transformer optimized for cross-lingual understanding and is effective for low-resource languages, such as Roman Urdu.

Roman Urdu as a Low-Resource Language

A low-resource language is one with limited labeled datasets, and computational resources for NLP tasks. Roman Urdu is considered low-resource because:

- There is no large-scale, publicly available dataset for text or speech in Roman Urdu.
- Users often mix English and Roman Urdu, increasing variability.
- Morphological and syntactic rules are inconsistent, making standard NLP techniques less effective.

Low-resource languages pose challenges for traditional machine learning models, which depend on large amounts of labeled data.

How XLM-R Handles Roman Urdu

XLM-R is pretrained on **100+ languages**, allowing it to:

1. **Leverage cross-lingual knowledge:** Knowledge from related languages (like Urdu or Hindi) helps understand Roman Urdu.
2. **Handle code-mixing:** Its subword tokenization allows sentences with English and Roman Urdu to be processed seamlessly.
3. **Adapt via fine-tuning:** Even with small Roman Urdu datasets, XLM-R can be fine-tuned to perform **intent detection** and **entity extraction** accurately.

7. XLM-R Modes:

XLM-R can operate in **three main modes**, depending on the task:

7.1 Masked Language Modeling (MLM):

  - This is the **default pretraining mode** of XLM-R.

- The model learns to predict masked (hidden) words in a sentence using context from surrounding words.
- Example: *"I want to [MASK] admission"* → Model predicts "apply".
- MLM helps the model capture general language patterns, semantics, and grammar.

7.2 Sequence Classification:

- This mode is used for **classifying entire sequences** into predefined categories, such as **intent detection** in NLU.
- Example: *"How can I check my application status?"* Classified as `check_status`.
- Sequence classification is obtained **after fine-tuning** the pretrained XLM-R model on a labeled dataset specific to the task.

7.3 Token Classification:

- This mode is used for **classifying individual tokens**, such as identifying entities in a sentence.
- Example: *"I want to apply for Computer Science in Fall 2025"* Tokens `Computer Science` labeled as `program`, `Fall 2025` labeled as `semester`.
- Like sequence classification, this mode requires **fine-tuning** on a task-specific dataset.

.

8. Fine-Tuning:

Fine-tuning is the process of taking a pretrained language model (like XLM-R) and adapting it to a specific downstream task. While pretraining allows the model to learn general language representations, fine-tuning enables it to specialize in tasks such as intent detection, entity extraction, or sentiment analysis.

- **Process of Fine-Tuning:**
  - Start with the pretrained XLM-R model.
  - Feed it a labeled dataset (e.g., sentences labeled with intents or tokens labeled as entities).
  - Train the model for a few epochs on this dataset, adjusting weights slightly to perform well on the specific task.
  - Evaluate performance and adjust hyperparameters as needed.

- **Why Fine-Tuning Matters:**
  - Pretrained models understand general language structure but cannot directly perform task-specific predictions without adaptation.

- Fine-tuning bridges this gap, enabling XLM-R to perform **sequence classification for intent detection** and **token classification for entity extraction** accurately in your dataset.

Summary:

Studying NLU provided deep insights into the mechanisms by which machines can interpret human language. It highlighted the importance of **intent detection**, **entity extraction**, handling **ambiguities**, managing **conversation types**, and selecting the right **approaches and models**. Through the study of transformer-based architectures, particularly XLM-RoBERTa, and practical dataset creation, a strong foundation in NLU has been established. This knowledge forms the basis for implementing sophisticated conversational AI systems capable of understanding and responding accurately to diverse, multilingual user queries.

References:

1. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. NAACL.

2. Conneau, A., Khandelwal, K., Goyal, N., et al. (2020). *Unsupervised Cross-lingual Representation Learning at Scale*. ACL.

3. Rasa. (2023). *Rasa Open Source Documentation*. https://rasa.com/docs/

4. Jurafsky, D., & Martin, J. H. (2021). *Speech and Language Processing* (3rd ed.).

5. Goldberg, Y. (2017). *Neural Network Methods for Natural Language Processing*. Morgan & Claypool Publishers.