N-Gram Based Diary Generation in Roman Urdu

Implementation Overview

We implemented unigram, bigram, and trigram models to generate diary-like text in Roman Urdu. The dataset consists of daily routine diary entries. The key steps included:

1. **Data Preprocessing**
    - Loaded and tokenized text, converted to lowercase, and removed unnecessary punctuation.
2. **N-Gram Model Training**
    - **Unigram Model**: Word frequency distribution.
    - **Bigram Model**: Conditional word pair probabilities.
    - **Trigram Model**: Three-word sequence probabilities.
3. **Sentence Generation**
    - Selected starting words from real diary sentences.
    - Used probability-based word selection for coherence.
    - Implemented smooth transitions between sentences.

Challenges Faced

- Handling punctuation while preserving time expressions.
- Ensuring logical sentence flow in bigram and trigram models.
- Addressing data sparsity in trigram generation.

Model Comparison

Model	Perplexity (Lower is Better)	Coherence
Unigram	**370.02** (Highest)	Very poor, random words with no structure
Bigram	**185.66** (Moderate)	Some fluency but lacks long-term context
Trigram	**266.94** (Better than unigram)	Most natural sentence flow, but data sparsity is a challenge

Conclusion

Trigram-based generation produced the most readable diary entries despite higher perplexity than bigrams. Further improvements can include smoothing techniques and a larger dataset.