Day 3: Tokenization in Urdu & Pashto NLP

Sami Uddin Shinwari
Subject Specialist - IT | NLP Researcher
MS in Data Science (FAST NU Peshawar)
BS in Computer Science (UET Peshawar)

December 23, 2025

What is Tokenization?

- Tokenization = splitting text into words (tokens)
- First step of every NLP pipeline
- Errors here affect all downstream tasks

Why Whitespace Works in English

- English uses spaces consistently
- Example:

This is a sentence

Tokens: This | is | a | sentence

Why Whitespace Fails in Urdu

- Space is not a reliable word boundary
- Writers often omit or insert spaces

Urdu Examples:

غلطالفاظ (no space)

غ لط الفاظ (extra space)

Problem: Both confuse tokenizers.

Urdu Sentence Example

یہ ایک جملہ ہے

Whitespace Tokens:
یہ | ایک | جملہ | ہے

Issue: Compound words break easily in real data.

Pashto Tokenization Challenges

- Similar Arabic-based script
- Inconsistent spacing in informal text
- Low-resource $\rightarrow$ no strong tokenizer

Pashto Example:
دایوهبیلګهده

Correct Form:
دا یوه بیلګه ده

Why This Matters

- Wrong tokens $\rightarrow$ wrong POS tags
- NER performance drops
- TF-IDF & embeddings become noisy

What Researchers Have Found

- Urdu tokenization requires rule-based methods
- Pashto needs whitespace correction models
- Subword tokenizers (BPE, SentencePiece) help

Better Tokenization Approaches

- Regex-based tokenization
- Morphological rules
- CRF-based segmentation
- Transformer subword tokenizers

Key Takeaways

- Whitespace tokenization is NOT enough
- Urdu & Pashto need language-aware tokenizers
- Tokenization quality defines NLP success

#Day3 #Tokenization #UrduNLP #PashtoNLP

References

- Rehman et al. (2011). Challenges in Urdu Tokenization. ACL.
- Becker & Riaz (2012). Urdu Morphology Study.
- Khan et al. (2023). Pashto Text Processing Challenges.