Assignment 2: Byte Pair Encoding

Instructions:

- The aim of this assignment is to give you an initial hands-on regarding Urdu Unicode processing.
- You can use any programming language but for your ease Python is highly recommended.
- Feel free to read chapter 2 of course book to get better understanding of BPE algorithm. (https://web.stanford.edu/~jurafsky/slp3/)
- You are not allowed to use code (not even a chunk) from the internet.
- Use a good Unicode text editor (such as BabelPad) to view Urdu files. Here is a link to the Unicode Urdu code page.
- Make a report in PDF format, clearly mentioning question/ part number against your answers.
- **This is an individual based assignment. Submit your report and code as one compressed file (.zip) on Google Classroom. The name of file should be your roll number i.e. <23L-110--->.zip (This is important)**
- Deadline to submit this assignment is: **Friday 3rd March, 2024 11.59 p.m.**
- **LATE SUBMISSIONS PENALTY!**

  20% after deadline (first day), 30% next day, not acceptable after that.

Problem No. 1:

For the first part of this problem you are given 50 unsegmented Urdu sentences and a dictionary (wordlist) for lookup. You are required to implement BPE algorithm to segment these unsegmented sentences. Write the segmented sentences to a new text file.

Note: You may pre-process dictionary to simplify the lookup process. Clearly mention your approach and pre-processing applied (if any) in not more than 100 words in your report.

File Details:

You are given the following files:

1. wordlist.txt (Dictionary for lookup in BPE)
2. 50_nospaces.txt (Unsegmented Urdu sentences)

Problem No. 2:

For the second part of this assignment, you have to calculate average no. of characters per word in a running Urdu text. Take a large enough Urdu text corpus (at least 10K tokens) from the internet. You can scrap blogs, news website etc. (but not poetry). Clearly mention the link in your report. Calculate no. of characters per token and report the average in your report.