QUESTION 6:

TOKENIZERS:

Indic -BERT tokenizer is specifically trained for languages of the Indian subcontinent, including Hindi. It is fine-tuned to better understand the composition and structure of these languages, which should lead to more accurate tokenization, especially for tasks like word grouping in Hindi. **mBERT tokenizer** is trained on a multilingual corpus, including Hindi. But it is not specifically optimized for Hindi, instead trained on a diverse range of languages and can handle multilingual text reasonably well. So, it should be able to capture word groups effectively in Hindi. **BPE (Byte Pair Encoding) tokenizer** might not be able to make right word grouping because it iteratively replaces the most frequent pair of consecutive characters with a new symbol, it can handle only some level of morphological differences but for Hindi morphological differences is more complex. **Unigram tokenizer** splits text into individual words based on whitespace or punctuation. So, it may work decently in most languages.

	UNIGRAM -1000	BPE-1000	BPE-2000	mBERT - 1000	mBERT - 2000	Indic-BERT -1000	Indic-BERT -2000	WHITE SPACE
PRECISION	0.021170610211706103	0.02981029810298103	0.02981029810298103	0.020531400966183576	0.020531400966183576	0.009917355371900827	0.009917355371900827	0.05731225296442688
RECALL	0.09239130434782608	0.11956521739130435	0.11956521739130435	0.09239130434782608	0.09239130434782608	0.03260869565217391	0.03260869565217391	0.15760869565217392
F-SCORE	0.034447821681864235	0.047722342733188726	0.047722342733188726	0.03359683794466403	0.03359683794466403	0.015209125475285173	0.015209125475285173	0.08405797101449276

COMPARISON:

1- According to the values of precision obtained from question-5 i am getting the best precision for white space tokenizer. This is because white space tokenizer is separating every single word which is matching with all **punctuation word groups** and all **conjunction word groups** as these are single words. But other models are breaking the words even which makes it difficult to form the right word group. Indic-Bert is supposed to responde best but actually it's giving syllables instead of words so right match is not possible.

Recall gives the true positives out of all actual positive instances in the data, and is also best for white space tokenizer. And hence, its F-score is also highest.

2- Also we can observe that the metric values for the same models (e.g. BPE) on different vocab sizes (1000 and 2000) are the same. From this we interpret that the token formed by them is very much the same. So, vocab size is not helping much in improving the right word group division for the sentences.