FINAL-PROJECT EVALUATION

Incorporating Context into Subword Vocabularies

(https://aclanthology.org/2023.eacl-main.45/)

Hemendra Meena(200050052,4th,CSE)

Nagesh Kumar(200050082,4th,CSE)

Vir Wankhede(210050166,3rd,CSE)

30 November, 2023

Problem Statement

Make a subword tokenizer whose objective is to make tokens that keep the context intact without compromising on encoding efficiencies.

Input : Sentence

Output : Context-based tokens

Dataset Used

- Datasource, Wikipedia dump :-
  - English - https://dumps.wikimedia.org/enwiki/20231101/enwiki-20231101-pages-articles16.xml-p20460153p20570392.bz2
  - Hindi - https://dumps.wikimedia.org/hiwiki/20231101/hiwiki-20231101-pages-articles.xml.bz2
  - Marathi - https://dumps.wikimedia.org/mrwiki/20231120/mrwiki-20231120-pages-articles.xml.bz2

Introduction

- Most of the currently used subword tokenizers are trained based on word frequency statistics over a corpus. This results in lack of consideration of co-occurrence and context.
- The SAGE model is a modified subword tokenizer whose objective is to make tokens that keep the context intact without compromising on encoding efficiencies and domain robustness.

SAGE Algorithm

- Tokenization - The initial vocabulary (V) is created using a basic tokenizer (T) on the given corpus (C). The size of this vocabulary is initially set to be n times larger than the desired final size (V).
- Iterative Refinement-
  - The procedure then enters a loop that continues until the size of the vocabulary (|V|) becomes equal to or less than the desired final size (V).
  - Within each iteration, the procedure alternates between updating the embedding table (E_V) and refining the vocabulary based on contextual information.
- Embedding Update -
  - Every l * m iterations, the embedding table (E_V) is updated using a Word2Vec algorithm. This helps to enhance the understanding of word relationships in the vocabulary.

SAGE Algorithm

- Likelihood Calculation -
  - The total likelihood of the current vocabulary is calculated based on the contextual information in the corpus.
- Vocabulary Update-
  - The procedure then evaluates and prunes the vocabulary iteratively.
  - Every m iterations, the likelihoods for removing each word from the vocabulary are calculated.
  - A bottom set of candidate words to be pruned (Vbot) is determined based on the calculated likelihoods.
  - The pruning is done by removing a subset of words (P) from the bottom set, and the vocabulary is updated accordingly.
  - The iteration counter (i) is incremented.

Parameters

- Vocabulary size :-
  - Initial Vocabulary Size - 4000
  - Final Vocabulary Size - 3000
- Other Parameters (as given in paper):-
  - Tokens to prune in each Iteration - 50
  - Token to consider in Iteration - 1000
  - Iterations until Reranking - 10

Work we have done

We have used SaGe to generate vocabulary on different language, viz. English, Hindi and Marathi, to see its results and effectiveness in tokens generated.

We have analysed these using different parameters as mentioned in paper

Results

- For English :-

- BPE- _Ta us en _continued _to _pre ach _to _ en orm ou s _c ro w d s _in _the _o p en .
- SAGE - _T a us en _continued _to _pre ach _to _en orm ous _c row d s _in _the _open
- The token ‘ous’ is chosen in SAGE because it a commonly used affix.
- BPE choses the token ‘ou’ which is not contextually relevant.
- Similar is done for word ‘crowds’, where BPE chooses tokens as ‘ro’ which has no meaning whereas SAGE chooses ‘row’ which
- Also the ‘_Ta’ token in BPE which has no meaning is replaced in SAGE by ‘_T a’, i.e. broken down to characters.

Results

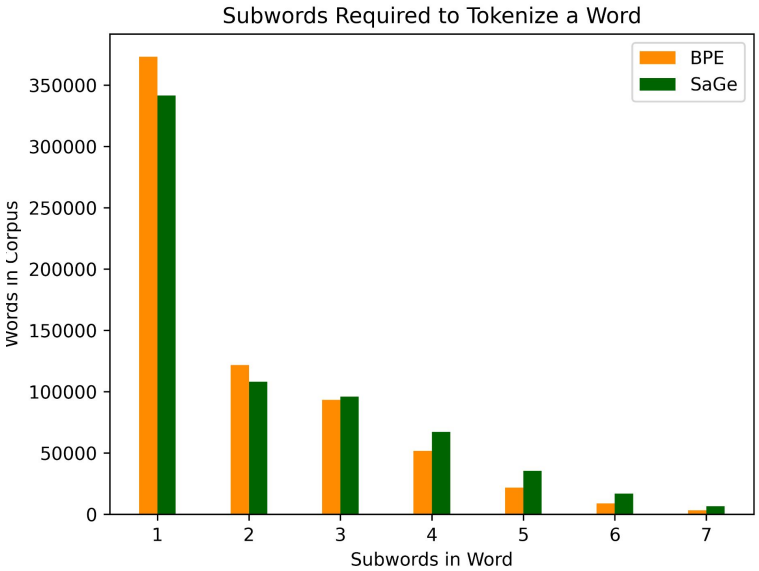
- For Hindi :-

- BPE - _किसी _भी _रोम ांच क _या त्रा _में _कै ंप िंग _एक _महत्वपूर्ण _स्थान _रख ती _है ।
- SAGE - _किसी _भी _रोम ांच क _यात्रा _में _कै ंप िंग _एक _महत्वपूर्ण _स्थान _रख ती _है ।
- BPE splits '_या त्रा' as 2 separate contextually incoherent tokens whereas '_यात्रा' is considered to be a singular contextually coherent token in SAGE.

Results

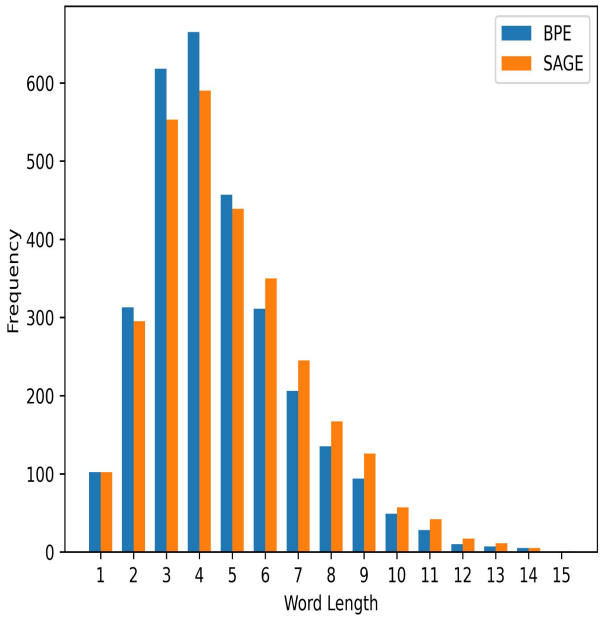
- For Marathi :-
- BPE - _रा न डे _यां ना _आ ध ु न िक _भार ताच्या _इतिहास ात _महाराष्ट्रातील _समाज सु धारण ा _च ळ वळ ीत _महत्त ् वा चे _स् थ ान _आहे
- SAGE - _रा न डे _या ं न ा _आ ध ु न िक _भारताच्या _इतिहास ात _महाराष्ट्रातील _स म ा ज सु ध ार ण ा _च ळ व ळ ीत _महत्त ् वा च े _स् थ ान _आहे .
- BPE splits into _भार ताच्या where both the tokens do not preserve much context but SAGE generates _भारताच्या as one complete token.
- Sometimes SAGE does falter compared to BPE, for example BPE generates _समाज सु धारण which are meaningful tokens but SAGE is unable to tokenize this and generates heavily split _स म ा ज सु ध ार ण ा
- We infer from this that SAGE, completely dismantles a word if it is unable to find context, unlike BPE which may produce ambiguous tokens.

Results (English)



- These results establish that SAGE completely dismantles a word if it's contextual meaning is unknown unlike BPE which splits it into 2-3 ambiguous subwords which may prove to harmful the language model.

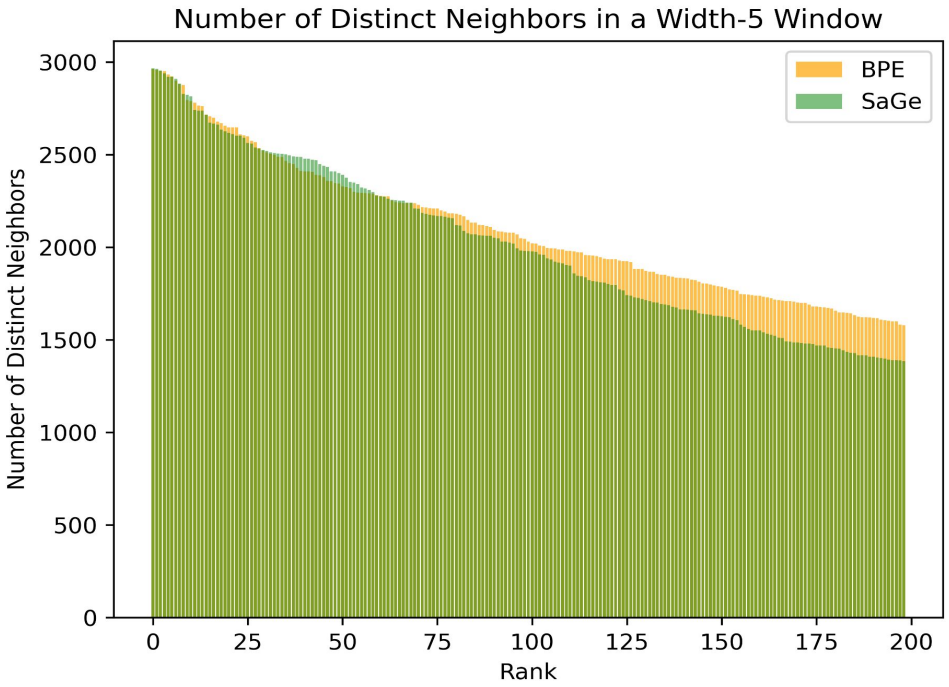
Results (English)



We can see that SAGE has more token of large length compared to BPE.

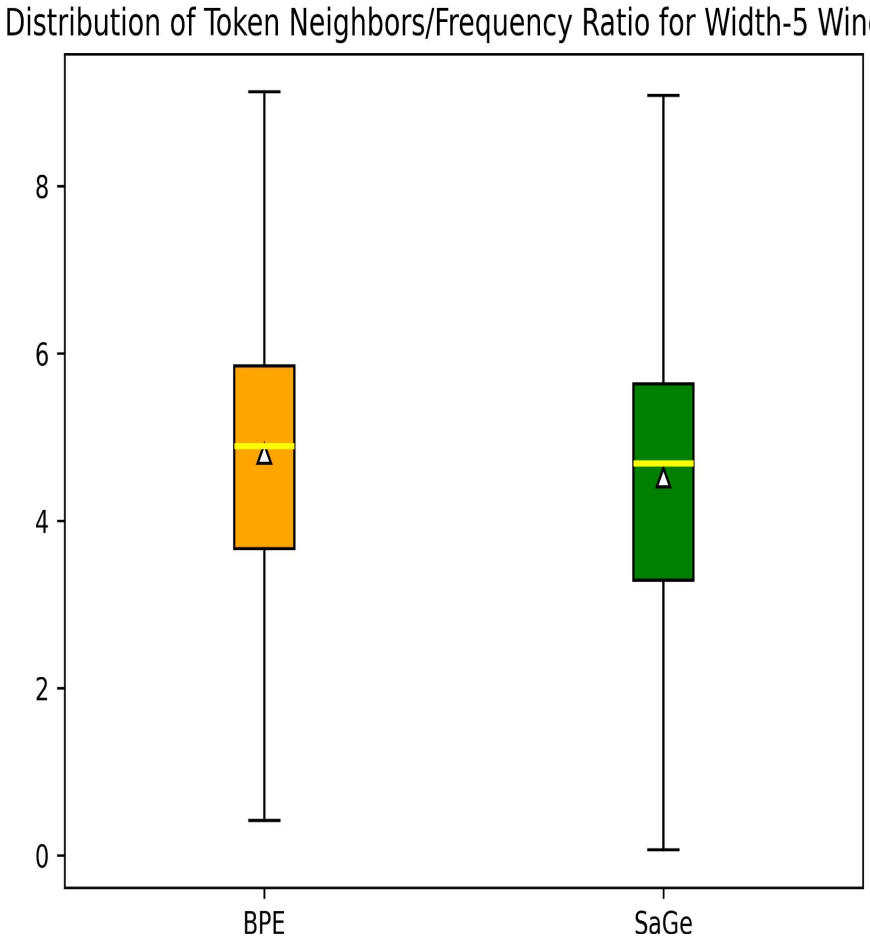
This implies that SAGE doesn't necessarily split a word into tokens always and tokenizes large subparts, or the word entirely in order to capture context.

Results (English)



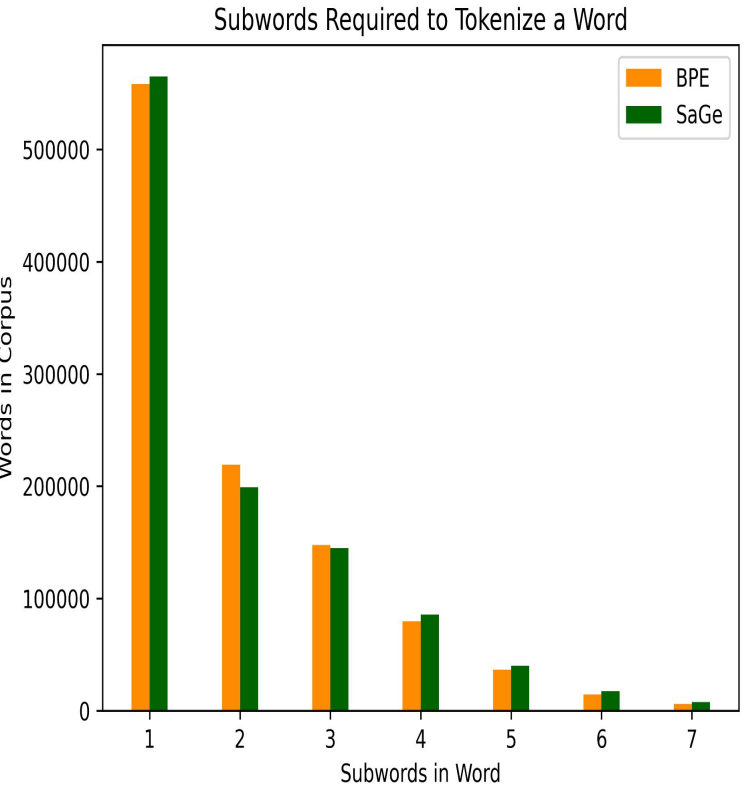
- Contextual coherence here refers to the extent to which SAGE enhances the meaningfulness of tokens
- Plot tries to represents contextual coherence through the number of distinct neighbors each token encounters throughout the training corpus, ranked from high to low
- Single character tokens, which are contextually neutral, occupy top places with similar number of distinct neighbors in both the tokenizers
- But after around fifth ranked tokens there is dip in number of distinct neighbors represented by tokens generated by SaGe, showing that they have more contextual meaning.

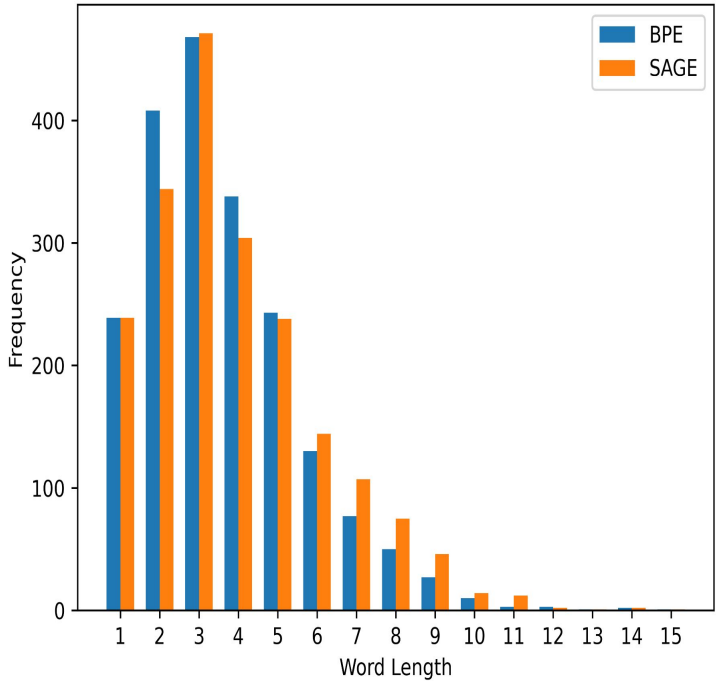
Results (English)



- Since the claim of contextual coherence as represented in previous plot can be be attributed to a frequency artifact of tokens.
- This figure clears that by presenting a normalized analysis of the distribution of Neighbors/Frequency Ratio.
- The figure clearly shows that SaGe provide lower ratios.

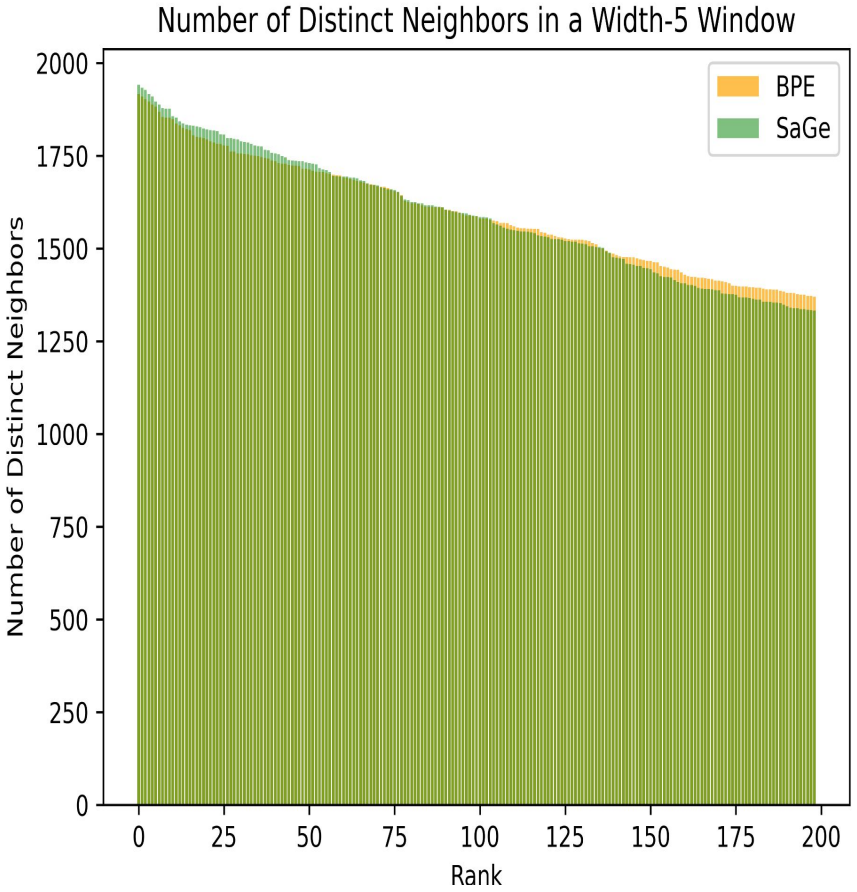
Results (Hindi)





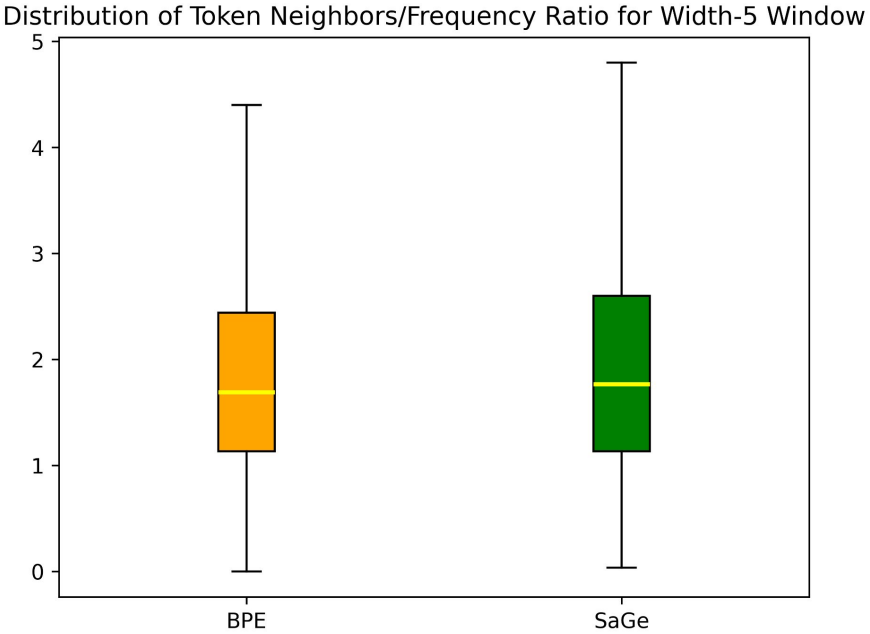
Similar results to English are observed in these figures

Results (Hindi)



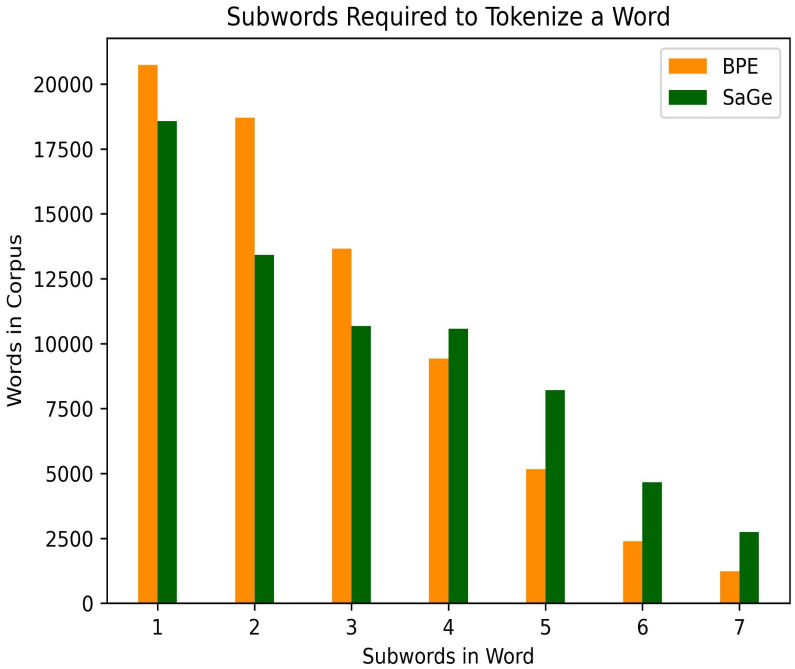
- Though there is similar dip in SaGe like in English but it is observed much later and much lesser (dip in number of distinct neighbours).
- This is due to the presence of much more single character type tokens in Hindi compared to English

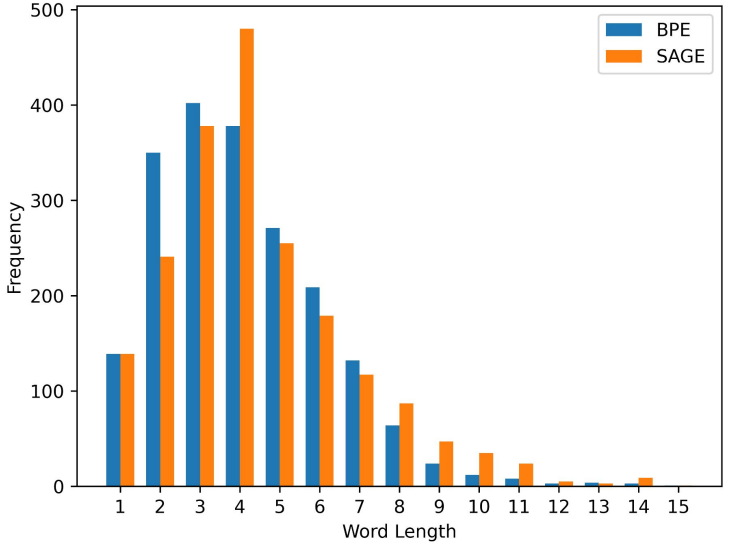
Results (Hindi)



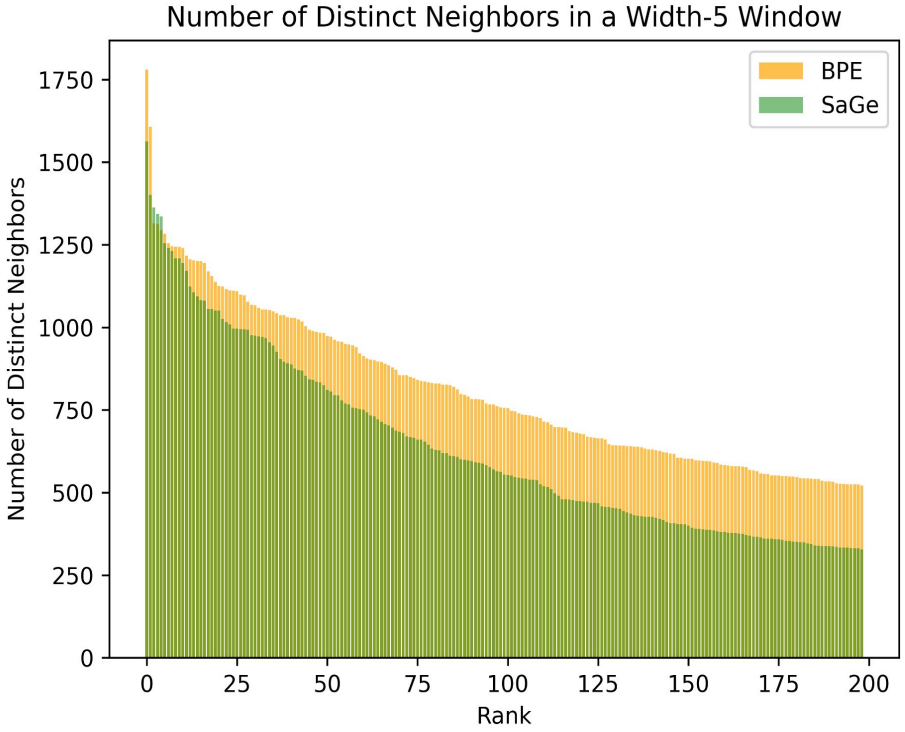
- Due to similar reason as in previous figure the results of this figure also vary from those of the English.
- As such the Distribution ratio is seen similar in both case BPE and SaGe.

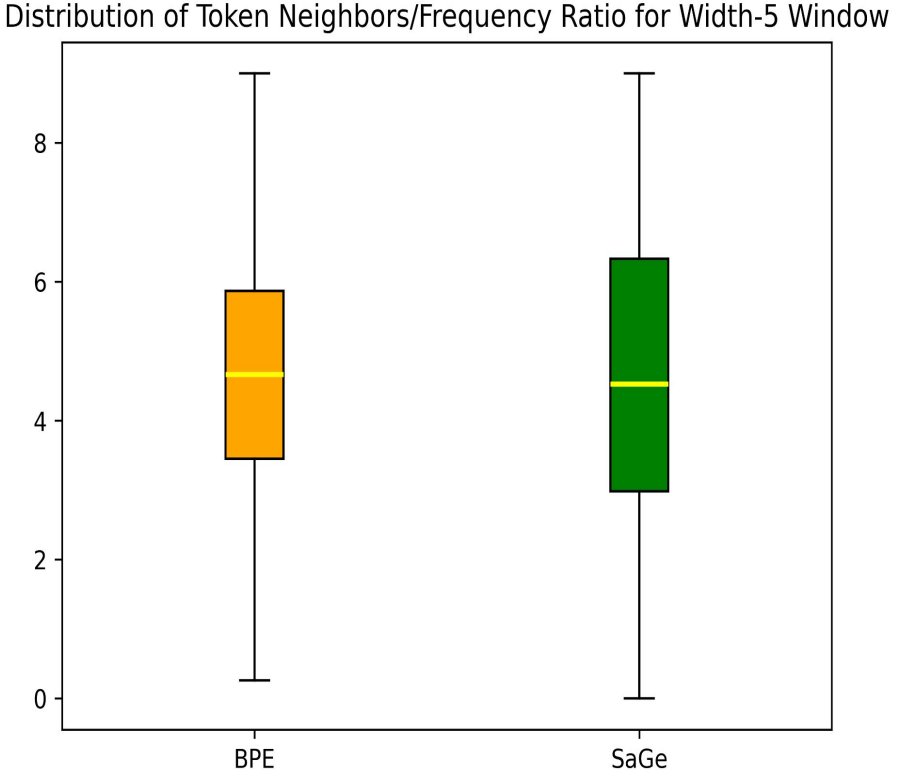
Results (Marathi)





Results (Marathi)





Conclusion

- Through the analysis based on experiments, it is clear that SaGe provides more context based tokens and replace non meaningful tokens with characters. This can be very helpful for the Language Models to train.
- Though there lies problem that the large vocabulary generation with SaGe requires high computational power.
- There is a need to make contextual tokenizer like SaGe faster and more computational efficient.

THANKS FOR ATTENDING!