Assignment – 3

Name: Srilalitha Lakshmi Anusha Chebolu
Date: 02/12/2025

New Insights:

Tokenization is a crucial step in NLP, where text is converted into smaller units (tokens) for processing. Among the popular tokenization methods, Byte Pair Encoding (BPE), WordPiece, and SentencePiece are widely used in transformer-based models like BERT, GPT, and T5.

Byte Pair Encoding (BPE)

BPE is a subword tokenization method that starts by treating each character as a separate token. It then merges the most frequently occurring adjacent characters into subwords. This process continues iteratively until a predefined vocabulary size is reached. The main advantage of BPE is that it can effectively handle out-of-vocabulary (OOV) words by breaking them down into smaller, recognizable subwords. This is particularly useful for languages with rich morphology. BPE is used in models like GPT-2 and RoBERTa. One of its strengths is that it balances efficiency and flexibility—words that appear frequently remain whole, while rarer words get split into smaller subword units. However, it doesn't consider linguistic meaning when merging subwords, and its fixed vocabulary size may limit generalization in certain cases.

WordPiece

WordPiece is another subword tokenization method, but instead of merging characters based solely on frequency, it uses a probabilistic approach to determine which subword combinations maximize the likelihood of the training corpus. This makes it more context-aware compared to BPE. WordPiece is commonly used in transformer models like BERT, DistilBERT, and ALBERT. It improves generalization by ensuring that words with shared roots or meanings have similar subword representations, which helps the model understand word relationships better. Like BPE, WordPiece effectively handles rare words by breaking them into smaller, meaningful subunits. However, it requires more computational power due to its probabilistic nature and can still suffer from the fixed vocabulary limitation, meaning new words might still get split into many pieces unnecessarily.

SentencePiece

SentencePiece differs from BPE and WordPiece in that it does not require pre-tokenization, meaning it works directly on raw text rather than relying on whitespace separation. This makes it particularly useful for languages like Japanese and Chinese, where word boundaries are not clearly defined. SentencePiece can be trained using either BPE or Unigram Language Modeling, with the latter allowing for more flexibility by selecting subwords based on probability. This method is widely used in models like T5, XLNet, and MarianMT. A major advantage of SentencePiece is that it handles multilingual text more effectively and does not rely on spaces to define token boundaries. However, it can generate more tokens than BPE or WordPiece, leading to longer sequences and higher computational costs. It is also slightly more complex to train and optimize compared to the other two methods.

Relevant Literature:

How BERT model works? **https://arxiv.org/abs/1810.04805**

Fine Tuning and Feature Based Approach:

- The fine-tuning approach involves adding a simple classification layer to the pre-trained model, with all parameters jointly fine-tuned on a downstream task.
- In fine-tuning, models like Generative Pre-trained Transformer (GPT) introduce minimal task-specific parameters and are trained on downstream tasks by simply fine-tuning all pre-trained parameters.
- For fine-tuning BERT, the model is first initialized with pre-trained parameters, then fine-tuned using labeled data from the downstream task. Each task has separate fine-tuned models, even if they start from the same pre-trained parameters.
- The feature-based approach leverages knowledge from a pre-trained model by using it to create fixed representations of input data without modifying the model itself.
- These embeddings serve as input for a lighter model, such as logistic regression or a basic neural network, which is designed for a specific task (e.g., sentiment analysis, named entity recognition).
- This new model (often referred to as a classification layer) does not modify the pre-trained model's parameters but learns from the fixed representations produced by it.

Pre-Training Tasks in BERT:

1. Masked Language Modeling (MLM)

- Some input tokens are randomly masked, and the model is trained to predict those masked tokens. This process is referred to as Masked LM (MLM) or the Cloze Task.
- While MLM enables a bidirectional pre-trained model, it creates a mismatch between pre-training and fine-tuning because the [MASK] token does not appear during fine-tuning.
- To address this, 15% of tokens are selected for prediction:
    1. 80% of the time, the token is replaced with [MASK].
    2. 10% of the time, it is replaced with a random token.
    3. 10% of the time, it remains unchanged.
- The model is then trained to predict the original token using cross-entropy loss.

2. Next Sentence Prediction (NSP)

- In 50% of cases, sentence B is the actual next sentence after sentence A (labeled as IsNext).
- In the other 50%, sentence B is randomly selected from the corpus (labeled as NotNext).