Quiz : LSTM Based Classification

Please answer the following questions.

* Indicates required question

Nom

Your answer

Adresse e-mail

Your answer

How do LSTMs handle the problem of vanishing gradients differently than traditional RNNs in the context of sentiment classification? * 1 point

- (●) By incorporating forget, input, and output gates to manage long-term dependencies.
- ( ) By using higher learning rates during training.
- ( ) By using ReLU activation functions.
- ( ) By reducing the size of the input data.

What are the specific roles of stop words removal, tokenization, and lemmatization in the preprocessing pipeline for sentiment analysis? Explain how each step contributes to the effectiveness of an LSTM model. 1 point

- (●) Stop words removal eliminates common words to reduce noise; tokenization splits text into manageable units; lemmatization converts words to their base form, ensuring consistency and reducing dimensionality.
- ( ) Stop words removal increases the size of the dataset; tokenization groups words into sentences; lemmatization replaces words with synonyms.
- ( ) Stop words removal enhances the semantic meaning of the text; tokenization removes punctuation; lemmatization creates synonyms.
- ( ) Stop words removal changes the structure of sentences; tokenization adds new words to the dataset; lemmatization breaks down words into characters.

Clear selection

Considering a tokenizer, what challenges might arise when fitting a tokenizer on a large and diverse dataset, and how can these challenges be mitigated? * 1 point

- ( ) Overfitting on rare words; use subword tokenization methods.
- ( ) Underfitting on common phrases; increase the vocabulary size.
- (●) Memory constraints and large vocabulary size; apply a maximum vocabulary limit and handle out-of-vocabulary tokens.
- ( ) Difficulty in tokenizing numerical data; convert numbers to text form.

When integrating GloVe embeddings into an LSTM model for classification, what are the potential limitations of using pre-trained embeddings, and how might these limitations affect model performance? * 1 point

- ○ GloVe embeddings have too high dimensionality; this increases the training time significantly.
- ○ Pre-trained embeddings are always more accurate than learned embeddings; they should not be modified.
- ○ Using GloVe embeddings increases model overfitting; this necessitates a larger training dataset.
- ◉ GloVe embeddings may not capture domain-specific nuances; this can lead to lower accuracy in specialized datasets.

How does the dimensionality of the embedding matrix influence the performance of an LSTM mode, and what considerations should be made when choosing the embedding size? 1 point

- ○ Higher dimensionality always improves performance; choose the largest possible embedding size.
- ◉ Dimensionality affects the balance between computational cost and the ability to capture word semantics; choose a size that balances these factors considering the dataset size and computational resources.
- ○ Lower dimensionality increases overfitting; always use a higher dimensionality.
- ○ The dimensionality does not significantly affect performance; any size can be used.

Clear selection

In the context of training an embedding layer from scratch using LSTM, what are the pros and cons compared to using pre-trained embeddings like GloVe? * 1 point

- (●) Training from scratch can lead to embeddings tailored to the specific dataset but requires more data and computational resources; pre-trained embeddings provide a good starting point but may not capture domain-specific nuances.
- ( ) Pre-trained embeddings always outperform learned embeddings; they should never be replaced.
- ( ) Learned embeddings are more accurate but harder to interpret; pre-trained embeddings are easier to interpret but less accurate.
- ( ) There is no significant difference in performance between learned and pre-trained embeddings; either can be used interchangeably.

Do you have any feedback on the programming session?

Your answer

Submit Clear form