Q1

Optimization algorithm need to know the direction of parameters efficiently, computing the gradients is crucial since it tell us in which direction to update our parameters. L_1 regularization promotes sparsity, which can be leveraged for efficiency in both storage and computation. However focusing on the total loss doesn't directly exploit this sparsity. In contrast, gradient-based optimization method can be adapted to more efficiently handle sparse data and parameters.

Q2 When we concatenate the embeddings of two previous words to form the model input, we are essentially providing the model with the context it need to predict the next word. If we were to concatenate these embeddings in the wrong order. The model will receive an incorrect context. For example, if the original text is " Cat eat fish", and we mistakenly concatenate the embeddings in the order " eat cat", the model will be trained on incorrect sequences. which would degrade its ability to learn the true patterns in the data.

Q3

Gradient vanishing: Occurs when the gradients become very small, effectively preventing the weights from changing their values. This makes it very hard for the network to learn and converge to a good solution.

Gradient exploding: Occurs when the gradients become too large. leading to very large updates to the network's weight. This can cause the learning process to diverge. Where the model fails to converge.

Vanishing/exploding gradient phenomena occurs in RNN because as the gradient are propagated backwards through time, they can become extremely small/large due to the repeated multiplication of gradient in each time step.
This happens in RNN because they have recurrent connection that allow them to store information from previous time steps.

LSTMs maintain a separate cell state along with a hidden state for each time step. it utilize three types of gates(input, forget, and output gate) to regulate the flow of information. The key to preventing the vanishing gradient problem lies in the LSTM's ability to maintain constant error flow through its cell state. This is achieved by the forget gate, which can learn to keep important information unchanged over many time steps. Because the cell state can carry gradient across many time steps without multiplying by potentially small numbers, LSTMs can mitigate the vanishing gradient problem, allowing for more effective learning for long-range dependencies.