***We implemented a module that corrects mistakes or errors of the words in the given text. The way we do that was by using a dictionary and map the mistaken work to most close one in the dictionary. However, there is a problem, in some scenarios it can not decide the correct one between the second and first option.***

***We can solve this by considering what context of the given word was placed on. So based on the words around that specific mistaken word we identify that which work it can chose from corresponding dictionary. In a way rating every word based on the context and chose the highest score between them as a perfect option to replace the mistaken word.***

***We want a solution for this part, as a result we collect a huge amount of data to train a model on it to understand the context. What we intend is using a bad of words module with window size of 5 and do not consider the order of words given. As a expert in NLP field, What you suggest we do in this scenario? Are our approach with bag of words correct or is any better solution or approach to this?***

***In Persian language if we use ParsBERT, we can collect the embeddings of the returned words from the dictionary for example our dictionary has 6 suggested replaced words in place of the given one. So I propose this:***

***We can collect the embedding of all words returned from the dictionary and then by choosing a window size of 5 we can perform a cosine similarity measurement between the mistaken word and the surroundings. After performing this then we can chose the highest similarity score as the replaced word.***

***Do you think that this approach is practical? Is there any better approach for this?***