Question: On which dimension does the examples in few shot influence the performance the most?

Or, first, which one is more important for a example in few shot given the test example? Do we need more similiar Content or Semantic structure? If it’s semantic structure, how so? In which way do we want the examples and our tested data to be semantically similar?

Basic setup:

When choosing good examples for few shot, we want to choose the ones that are closer to the test data we have (Liu2021), the performance is proven to be generally better when they are closer. But in which way do we want the examples to be closer to our test data: if we use knn to choose the nearest neighbors by euclidean distance, maybe there are some dimensions in this vector space matters more than the others. We want to evaluate the performance when choosing k nearest neighbors on considering only m(1 to k, 1 is easier during my experiment, or I can try PCA first from the embedding since the interpretable dimensions are combined together into different dimensions, and there are just too many of them using any transformer model’s embedding) dimension as few shot examples, to see how much does these examples improve in comparison with the baseline: the few shot performance when randomly choosing examples from a prepared set.

Then interpret this dimension use cluster use any method (e.g cluster sentences using data from some particular dimensions and see their similarities, or difference between different clusters; the sentences can be chosen from anywhere, not necessarily from the dataset we use previously)

Specific plan1:

Assuming we only use one model and some dataset, we first need a way to transform the examples into a vector in a space of n dimensions. N hopefully is low b/c we want to interpret these dimensions(do PCA). Our hope is to find a dimension that contains information about certain **syntactically** patterns of given datasets, not on the content side.

If the hypothesis that “the content of the example is the key” is true, then the the variance of the ideal dimension to find good examples, must be very high, and the examples chosen will certainly have similar content as the test example; then it might prove the point that being familiar with the content of the test example is more important. **If it’s true cross all the dataset then there’s a crucial conclusion we can draw: for all tasks similar content in examples improves performance in few shot.**

If we find some common ideal dimensions to find good examples that are ideal for all test datas, these dimensions are possibly **syntactically** meaningful for few shot learning. Then we will need a linguitical way of interpretting these special dimensions. **If there’re some special dimensions that matters a lot for all dataset, then such dimensions surly contains info about syntax that strongly influence all tasks’ performance.** If for each dataset, there’re different special dimensions for them, then we can interpret this dimensions and try to figure out why some dimensions are particularly important for some task.

The worst but a very likely case woud be that the similar content and similar **syntact** matters almost equally, (or the difference is not significant enough), that will give us the a blurred conclusion, after which we can’t really go anywhere. Try the best as I can to interpret the difference there, or try to use the second plan.

Specific plan 2:

Try the same experiment but use another model that exclude the examples that has the very similar content with the test data first, so that the knn is forced to pick the ones that are syntactically similar to the test data, then just see if it improves the performance. (at least the result will not be heavily influenced by the similarities of content using this method)

After that, if it doesn’t improve the performance of the baseline at all, awesome, then similar content is the only helper, which is very unlikely.

If it does, then we should try the similar things in plan1, do knn using only several dimension, then interpret the most usful ones. But it must be difficult to do at least for me since I know nothing about linguistic. (do PCA first so it’s easier to interpret)

Limitation here would be that we want to make sure that the same content examples is fully excluded, so we can’t just use simpler models like knn. But any transformer based model can do this as classification task; if possible, just pick the decision boundary and exclude the data that we don’t want based on their word embedding, this is much faster and also doable.

Concerns:

During the process of KNN we want the input part only not the labels, otherwise the retrieval model will have the higher chance of picking the ones with same answer

Do this experiment on appropriate datasets (making sure the dataset can have at least above random performance) For easy dataset, find knn as example might lead the model to just copy the answer. But if it’s too difficult, even if the performance is above random guess result, it might just learn some very **specific** lower level feature, in this case even if we located a ideal dimension that’s not about the content, the info there might be specifically important to this one test data, so it’s not about syntax either, so it’s less possible for us to find the ideal dimensions .

Limitations:

We should try as many models as possible, to see if it’s true that closer example is always better for all model

(Liu2021)What Makes Good In-Context Examples for GPT-3