

Your Project Title and Author Names

November 18, 2024

1 Project Overview (10 points)

Paragraph 1: The first paragraph is a funnel. You start with the broadest scope and quickly narrow down to the specific version of the problem you are tackling. First state the problem area you are in working in the broadest terms and motivate it. That is state why this problem area is important. Usually you can use applications or other impact of solutions for this problem area. Then state why this isn't an easy thing to do. What is the basic challenge in this problem space? In other words you can state why this problem area hasn't been solved. You can then make a statement about the scope you are considering and state your overall goal within this problem space. You should very clearly state the overall goal of your project and describe the task you are trying to solve in terms of inputs and outputs.

Paragraph 2: I can this the gap paragraph. What has been done about this type of problem before? There should be some related work. Sometimes there may not be work that solves your exact problem but that is ok. You should find the closest set of solutions and describe them at a high-level here. What type of approaches has been attempted for this problem? Then, provide a line or two describing why the existing solutions are inadequate. End this paragraph with a line that states at a high-level what needs to be done.

Paragraph 3: Give your proposed ideas here. What are the ideas/methods you are trying to address this problem? Now you have stated the gap(s) in the previous paragraph. Your goal here should be to give a high-level description or intuition for what ideas are likely to work in addressing the gap and why.

Paragraph 4-6: Detailing the ideas. Use this and one or two subsequent paragraphs to explain the ideas in a some more detail. For each idea you will specify the steps involved (or components that needed to be developed). If you are using existing implementations or previous papers for the ideas, cite them here. Make sure that you tie back to how each idea addresses the gap in terms of the specifics.

Paragraph 7: State how you will evaluate the ideas. Describe the main evaluation questions that need to be addressed to demonstrate/assess the usefulness of your ideas. State what systems you will use to implement, datasets you will use to evaluate, and what evaluation measures you will be using. Identify what kind of analyses you will be doing in addition to the main evaluation measures which will help you understand the utility of your ideas under different settings.

Paragraph 8: Summarize your main findings here. What is the high-level finding from your experiments.

2 Ideas (10 points)

Describe in your own words the ideas you tried for the task. Please describe your ideas and how you implement them in detail. Take a page if you need for each for describing each idea. Use figures if it helps to describe your idea. See

2.1 Idea 1

Describe idea 1 here.

2.2 Idea 2

Describe idea 2 here.

2.3 Idea 3

Describe idea 3 here.

3 Experimental Setup (10 points)

In this section you will describe what models you used, the datasets used, and list the evaluation metrics.

3.1 Models

Describe each model you used at a high-level. What type of model is it? An RNN? A transformer? How many layers do you have in your model? Was it pre-trained? Or trained from scratch.

If you trained any models, describe the training settings. How many epochs did you train the model for, how long . List the main hyper-parameters (e.g. learning rate, batch size, optimizer used etc.).

Method	Precision	Recall	F1
DistilBERT	0.40	0.30	0.34
GPT-3(ZS)	0.50	0.60	54
GPT-3(FS)	0.55	0.65	0.59

Table 1: Comparison of the three methods on the Fixed test set of the PerSent dataset: GPT-3 in the few-shot achieves the best performance across all measures. GPT-3(ZS) is the zero shot use of GPT-3 and GPT-3(FS) is the few-shot method with K=3 examples.

3.2 Dataset

What is the dataset you are using? What is the input/output format for each instance in your dataset. How many training instances are there? How many test instances did you use?

3.3 Evaluation Metrics

Say how you evaluated your model. Human evaluation or automatic evaluation? What are the metrics you use to compare the different models. Explain each metric.

If you use human evaluation for the output, describe what criteria you used to evaluate the outputs. If you use manual evaluation you only need to do this on a small number of examples (say 25 examples for each method).

4 Results (40 points for implementation and results)

In this section describe the main results you obtained using tables or plots. Show the best result you were able to obtain under each idea.

For example, if you used fine-tuning DistilBERT, Zero Shot GPT-3, and Few Shot GPT-3 as the main ideas for the PerSent task, your table should look something like Table ??

1. You can report results on training, validation (development) and the test sets in separate tables. Note that the main result you should explain is the one for the test set.
2. If you tried different hyper-parameter settings, different prompt methods, or any variations for each method show results for those in separate tables.

5 Analysis and Discussion (25 points)

Take one model from the ideas you tried above and do a manual analysis of the outputs of this model. Your goal here is to try and understand what kinds of inputs

this model succeeds and fails on. You are free to explore how to do this analysis. This section should include the following:

1. Identify some types of failures of the model.
2. Come up with three hypotheses for the kinds of inputs the model fails or succeeds on.
3. For each hypothesis include an example input and the model's output. Use the example to make your point.
4. If your hypothesis is true then try modifying the input to show that the model actually behaves in the way you anticipated. It is ok if it doesn't. This means your hypothesis isn't correct or it is not easy to test your hypothesis in this way.

6 Code

In this section, please provide a **google drive link to your packaged code** or give us a link to your github repo hosting these.

The code itself should be structured with a README that clearly specifies the following:

1. List the original source for your code base. Include the URL to the original source.
2. The list of files that you modified and the specific functions within each file you modified for your project.
3. A list of commands that provide how you train and test your baseline and the systems you built.
4. If you trained any models please include the trained models and links to the data you trained the models on.
5. If you used prompts please put the prompts in a text file and include them.
6. A list of the major software requirements that are needed to run your system. (E.g. Tensorflow 2.3, Python 243.12, CUDA abd2.0, nltk-2401.11, allen-nlp 5.0).

These descriptions should be adequate enough to help anyone who wants to run your system.

If we need additional information, we will ask you for it when grading.

Please include the above information in this report in a succinct fashion (the actual README can be detailed).

7 Learning Outcomes

List what you learnt from doing this project.

8 Contributions

List what each member contributed to this project. Note that the entire team will get the same grade.