CS378: Final Project - Data Artifacts
https://github.com/sarimaleem/cs378_fp

Sarim Aleem
ska2222

Abstract

TODO: write an abstract describing the core motivation and findings from your project

1 Introduction (5pt)

TODO: This section should include three paragraphs. The first paragraph is to briefly describe task and data. The second paragraph should describe the results of your analysis, and the third paragraph describing your fix and main experimental take aways.

The model that was used was [1]

2 Task/Dataset/Model Description (15pt)

TODO: Describe the task/dataset/model you are working on. Clearly define your task with mathematical notations. Describe your learning algorithm. You must formally specify the loss function, stopping criteria, training data, etc used for the model you are analyzing in the next section. Remember, every notation you use must be defined.

3 Performance Analysis (25pt)

The model was trained for 3 iterations on the SNLI training dataset. Each iteration has 550152 training examples, the batch size for the training examples was 32. All training was done on google colab in the cloud, and training took about 25 minutes.

Overall Accuracy Statistics

An initial evaluation of the model shows that it has an accuracy of 88%. We also did further evaluation to show statistics about which labels it misclassified most using a confusion matrix.

Table 1: Development Set Evaluation Metrics

loss	0.315
accuracy	0.8865
runtime	20.2629
examples/s	485.715
steps/s	60.751

Model Metrics for 9842 examples

Table 2: Confusion Matrix

		Predicted			
		0	*1*	*2*	**Total**
True	*0*	2996	252	81	3329
	1	213	2787	235	3235
	2	79	257	2942	3278
	Total	3288	3296	3258	**9842**

0=Entailment, 1=Neutral, 2=Contradiction

Table 3: Percent mispredictions

		Predicted		
		0	*1*	*2*
True	*0*	89.99	7.78	2.47
	1	6.39	86.15	7.16
	2	2.37	7.94	89.74

0=Entailment, 1=Neutral, 2=Contradiction

In general, the model seems to do well with Separating Entailment and Contradiction. For example, it only incorrectly identifies 2.47% of entailment examples as contradiction and 2.37% of contradiction examples as entailement. However, the model struggles more to understand the relationship between, entailment and neutral, as well as contradiction and neutral.

TODO: Describe clearly how you conducted your analysis of the model performance, and report the outcomes. You should provide (1) some examples of both **specific errors/behavior** from the model and (2) analysis or discussion of **the general class of mistakes the model makes**. For characterizing the general class, try to come up with rules that can identify a set of challenging examples (e.g., "examples containing *not*") and try to visualize in charts, graphs, or tables what you believe to be relevant statistics about the data. For example, how frequent is the class of examples where such error pattern apply? This part of the report should be at least one page.

4 Describing Your Fix (20pt)

TODO: You should describe modifications you have made to improve performances. We will evaluate based on how reasonable is your fix – if your fix and motivation does not link, your modificiation is unreasonable. For example, perhaps you tried to modify neural network training in a way that is totally unconnected to your stated goal, or your modification was erroneous. Describe the baseline model to be compared to your approach as well here. Again, clearly define all notations. Someone who is reading your report (with reasonable background, e.g., your classmates in this course) should have a reasonable idea how to implement it themselves. Describe hyperparameters, including how they are selected, if you have hyperparameters involved.

5 Evaluating Your Fix (25pt)

TODO: Your writeup should address how effective is your fix, how broadly applicable is your fix, etc. Providing a single number (overall accuracy) is necessary but not sufficient here. For example, if your change made the model better on challenging NLI examples, you could try to quantify that on one or more slices of the data, give examples of predictions that are fixed, or even use model interpretation techniques to try to support claims about how your improved model is doing its "reasoning." (You can look at the papers listed above to get a sense of how to do such fine-grained evaluation). You should report results from a baseline approach (your initial trained model) as well as your "best" method. If doing your own project, baselines such as majority class, random, or a linear classifier are important to see. **Ablations**: If you tried several things, analyze the contribution from each one. These should be *minimal* changes to the same system; try running things with just one aspect different in order to assess how important that aspect is. This part of the report should be at least one page.

6 Related Work (5pt)

TODO: Briefly discuss prior research papers related to your approach. This will likely some papers in the project description document. How is your approach different from existing studies?

7 Conclusion (5pt)

TODO: Brief conclusion summarizing findings from both numerical results and qualitative analysis.

(Optional) AI Assistance

TODO: If you have used any AI toolkit (either for writing assistance or code assistance) for your final project, please describe it here.

References

[1] Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. Electra: Pre-training text encoders as discriminators rather than generators, 2020.