UDACITY

Back to Machine Learning Engineer Nanodegree

Capstone Proposal

REVIEW

HISTORY

Requires Changes

3 SPECIFICATIONS REQUIRE CHANGES

Dear student

Great start on this proposal! I've noted a few areas where you should add a bit more detail, but I think that you've picked a great project and you're definitely on the right track. I think you'll see that most of these things shouldn't take long to update. Almost there...keep going!

Cheers!

Project Proposal

Student briefly details background information of the domain from which the project is proposed. Historical information relevant to the project should be included. It should be clear how or why a problem in the domain can or should be solved. Related academic research should be appropriately cited. A discussion of the student's personal motivation for investigating a particular problem in the domain is encouraged but not required.

Great job giving the reader an introduction to the problem domain!

Still required:

- Please cite or link to an academic paper where machine learning was applied to this type of problem.

Suggested:

- If you include a link to your data source in this section, you can directly lift it into the 'Project Overview' section of your capstone report.

Student clearly describes the problem that is to be solved. The problem is well defined and has at least one relevant potential solution. Additionally, the problem is quantifiable, measurable, and replicable.

Nice job here! I think that this is very clear.

The dataset(s) and/or input(s) to be used in the project are thoroughly described. Information such as how the dataset or input is (was) obtained, and the characteristics of the dataset or input, should be included. It should be clear how the dataset(s) or input(s) will be used in the project and whether their use is appropriate given the context of the problem.

Nice start! Here are a couple of things to be sure to include in this section next time you submit your project:

- How many examples (specifically) are there in the dataset that you'll be using?
- How are the classes in the dataset balanced (how many examples of each class)?
- How will you split the data into training/validation/testing sets? Will you do anything to maintain class balances across each subset?

Student clearly describes a solution to the problem. The solution is applicable to the project domain and appropriate for the dataset(s) or input(s) given. Additionally, the solution is quantifiable, measurable, and replicable.

A benchmark model is provided that relates to the domain, problem statement, and intended solution. Ideally, the student's benchmark model provides context for existing methods or known information in the domain and problem given, which can then be objectively compared to the student's solution. The benchmark model is clearly defined and measurable.

Good choice! Logistic regression is a common default implementation that shouldn't be too hard to beat with your proposed solution.

Student proposes at least one evaluation metric that can be used to quantify the performance of both the benchmark model and the solution model presented. The evaluation metric(s) proposed are appropriate given the context of the data, the problem statement, and the intended solution.

AUC is a great metric for binary classification problems. Additionally, it should be capable of handling class imbalances (if there are any).

Student summarizes a theoretical workflow for approaching a solution given the problem. Discussion is made as to what strategies may be employed, what analysis of the data might be required, or which algorithms will be considered. The workflow and discussion provided align with the qualities of the project. Small visualizations, pseudocode, or diagrams are encouraged but not required.

> As part of this task, we will complete all of the data preprocessing, if needed, and subset the entire dataset to create the independent and response variables needed to train the baseline model. To accomplish this task, we will need to first complete any preprocessing that is needed to have a complete dataset based on our findings from Task 1.

Can you provide a bit more detail about what pre-processing steps you're considering? For instance:

- Do you think that you'll need to normalize or scale any of the data?
- Will you do anything to detect outliers?
- Are there missing values to deal with? If so, how will you handle data points with missing values?

> For the baseline model, we'll use a logistic regression model with the standard defaults as given in the sk-learn library. Once the baseline model is developed, we will create the learning/validation curves and the ROC plot and determine the area under the curve. We will document the findings as it relates to the baseline model based on the shapes of the curves and the AUC value that we obtain from this task. Once the baseline model is developed, we will employ the grid search method to optimize the model's parameters and the value of the AUC value. If we are able to find a model with a higher AUC value than the initial run we will use the optimized version of the model as the baseline model.

Please note that your solution to the problem should be different than the benchmark model. Since the point of a benchmark model is to provide a clear threshold that determines whether the project has succeeded, it's important that you pick a different model for your solution so that you can make an objective comparison between the benchmark and the solution to the problem. Odds are very good that you should be able to easily beat a default implementation of logistic regression with many of the other (more complex) SKLearn models (random forests, svc, adaboost etc.).

Suggested:

- The XGBoost and LightGBM models could be good supervised learning approaches to try here.
- If you create multiple supervised learning models, you could try combining them all together into a custom ensemble model:

http://blog.kaggle.com/2016/12/27/a-kagglers-guide-to-model-stacking-in-practice/
https://www.kaggle.com/arthurtok/introduction-to-ensembling-stacking-in-python

Proposal follows a well-organized structure and would be readily understood by its intended audience. Each section is written in a clear, concise and specific manner. Few grammatical and spelling mistakes are present. All resources used and referenced are properly cited.

The template format is followed and the proposal is well written.

RESUBMIT

DOWNLOAD PROJECT

Best practices for your project resubmission

Ben shares 5 helpful tips to get you through revising and resubmitting your project.

Watch Video (3:01)

Rate this review