Standard Machine Learning Language

Kelechi Ikegwu   IKEGWU2@ILLINOIS.EDU
226 Astronomy Building, MC-???
1002 W. Green St.
Urbana, IL 61801

Micheal Hao   ???@ILLINOIS.EDU
???

Robert Brunner   BIGDOG@ILLINOIS.EDU
226 Astronomy Building, MC-221
1002 W. Green St.
Urbana, IL 61801

Abstract

Standard Machine Learning Language (SML) is a language agnostic framework that integrates a query-like language to simplify the development for a variety of state-of-the-art machine learning pipelines. Emphasis was placed on ease of use and abstracting the complexities of machine learning from the end user encouraging it's use in professional and academic settings from a variety of disciplines. SML's architecture is discussed, followed by multiple interfaces that one could use to interact with SML. We then apply SML to a few research problems and compare the complexities for multiple problems. Lastly we perform a case study on SML. The source code and documentation for SML is open sourced and publicly available on github (?).

1. Introduction

Machine Learning has simplified the process of solving a vast amount problems in a variety of fields by learning from data. In most cases machine learning has become more attractive than manually creating programs to solve these same issues. However they're a lot of nuisances (that a novice may be unfamilar with) involved when developing machine learning pipelines (?) and if they are not taken into consideration one may not receive satisfactory results. In addition to this, an experience user may not want to deal with [REASONS HERE]. In this paper we introduce Standard Machine Learning Language (SML) which helps to combat these two use cases.

The overall objective of the SML is to provide a level of abstraction which simplifies the development process of machine learning pipelines. Consequently this enables researchers and industry professionals without a background in this area to use machine learning to solve problems. We developed a query like language to which serves as an abstraction from writing actual machine learning code see Figure 1 for an example.

2. Related Works

Related Works Stuff

```
READ ".../path/to/data.csv" AND
SPLIT (train = 0.8, test = 0.2) AND
CLASSIFY (predictors = [1,2,3,4], label = 5, algorithm = svm)
AND PLOT
```

Figure 1: Example of a SML Query. TODO: FIX COLOR SYNTAX

3. Grammar

Grammar Stuff

4. SML's Architecture

Architecture Stuff

4.0.1 MODEL PHASE

model phase

4.0.2 APPLY PHASE

apply phase

4.0.3 METRICS PHASE

It's often useful to visualize the data that one works with; it's also benefical to see the performance metrics of your machine learning algorithm to better understand ones data. By default if you specify the 'PLOT' keyword in a query, SML will execute the metrics phase. Figure 2 displays a block diagram of the metric phase of SML. SML performs a dictionary lookup with the average complexity of O(1) to find specific terms that are in the query. For the example in Figure 2 we specified 'PLOT' which instructs SML to create visualizations, with the 'READ' keyword SML will create a lattice plot containing Kernel Density Estimates. Given an algorithm type such as Classification SML generates plots such as ROC Curves and Validation and Learning Curves. For a comprehensive list for the type of plots that SML can generate visit ().

4.0.4 PARSER

parser

4.0.5 CONNECTOR

connector

5. Interface

They're multiple interfaces available for working with SML. We've developed a web tool that's publicly available which allows the user to interact with SML. There's also a REPL
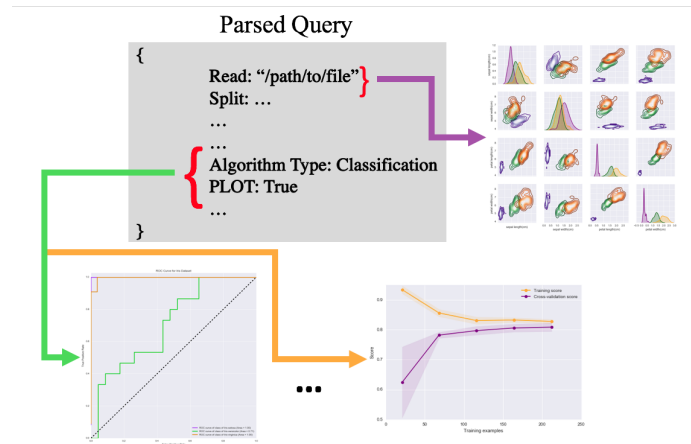
Figure 2: Block Diagram of Metric Phase Architecture

environment available that allows the user to interactively use SML. Lastly, users have the option to import SML into an existing pipeline to simplify the development process.

6. Use Cases

use case stuff

7. Case Study

case study stuff

8. Future Work

future work stuff

9. Conclusion

conclusion stuff

Acknowledgments

acknowledgments go here

Appendix X...

Appendix goes here if needed.