Supporting Ad Hoc Experimentation in Keras through Auxillary Analytics Systems

Andrew McNutt, Will Brackenbury, Ahsan Pervaiz

University of Chicago, Department of Computer Science

ABSTRACT

In recent years, systems such as Jupyter Notebooks have enabled the rapid, ad-hoc, iterative development of deep learning models. As models are increasingly developed in this manner, however, factors such as reproducibility become more of a concern: mis-transcribed results, differing representations of data, and reliance on manual record-keeping all contribute to this. To address this, we provide Ad-Hoc Nota-Bene (AHNB), a system integrated with Keras and Jupyter Notebooks to help scientists perform repeatable deep learning experiments with minimal overhead. We demonstrate the effectiveness of the system's approach through resilience and scalability testing, as well as a small-scale user study.

INTRODUCTION

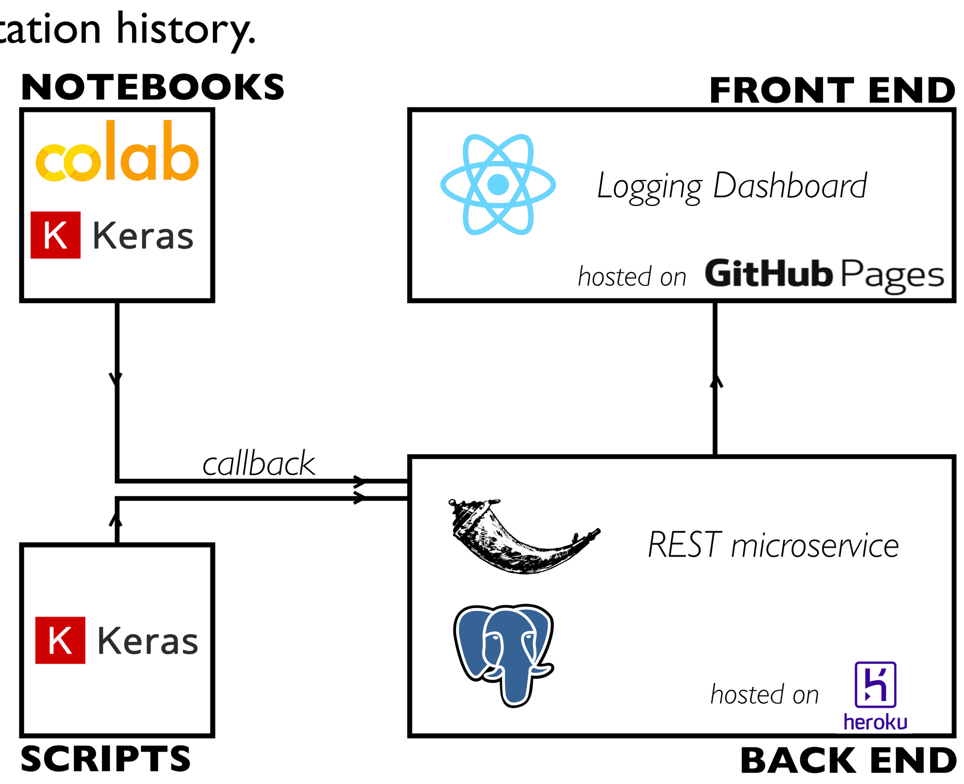
Over the last decade, the rising ubiquity of "big data" has dramatically increased the need for analysts who can extract, clean, and utilize this data. Many have noted this: the number of data scientist roles on LinkedIn has grown over 650% since 2012, and the U.S. Bureau of Labor Statistics estimates data science will create 11.5M more roles by 2026. These roles, however, differ in important ways from data-focused roles previously in industry. Well-known data workflows such as monthly reporting or large-scale Six Sigma studies require different sets of tools from the more exploratory, ad-hoc nature of data scientist workflows.

This type of workflow is particularly common in early stages of pipeline development for practitioners of "deep learning", the colloquial term for statistical modeling with multi-layer artificial neural networks. The aforementioned tools, in combination with Keras and back-end frameworks like Tensorflow and PlaidML, form the bedrock of many common activities for data scientists. They will likely perform all these activities in a notebook environment, potentially deleting lines of code where appropriate, perhaps adding descriptive text.

Prior work has shown, however, that the common use cases in these activities often result in potentially lost or mis-transcribed research data. While researchers and industry have developed some tools to combat this trend and to increase the explainability of results, we focus on the specific use case of optimizing hyperparameters for deep learning models. In this case, known tools either generalize across models and therefore miss many deep-learning-specific optimizations, or limit themselves to a specific platform for development. To this end, we introduce Ad-Hoc Nota-Bene (AHNB), an environment-agnostic tool designed to assist deep-learning practitioners in the model hyperparameter optimization process without any manual overhead.

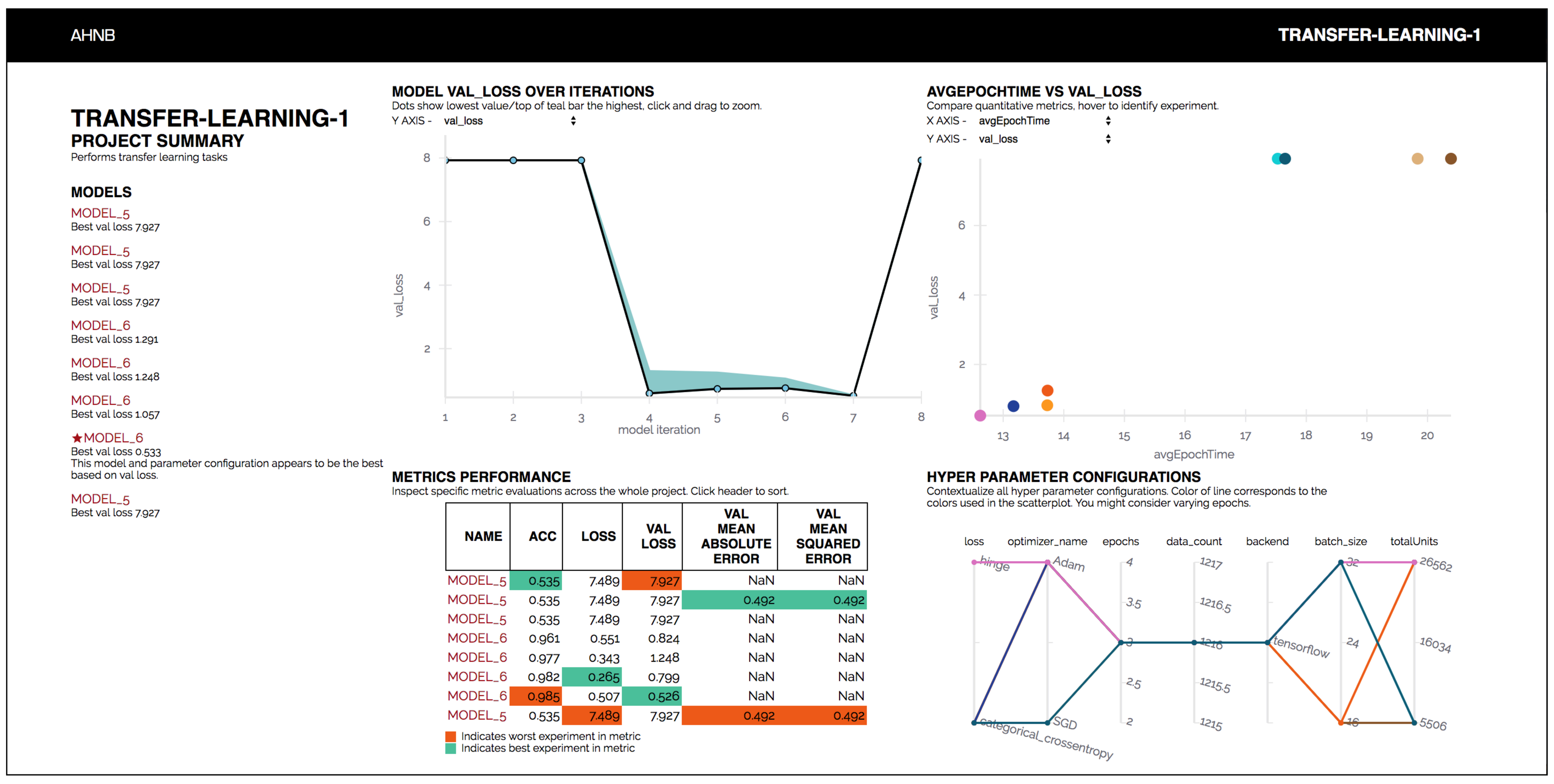
IMPLEMENTATION

Our prototype consists of three distinct components: the Keras plugin or callback, a rest microservice that receives logs and statistics collected by the callback, and a logging dashboard that allows users to visualize their experimentation history.

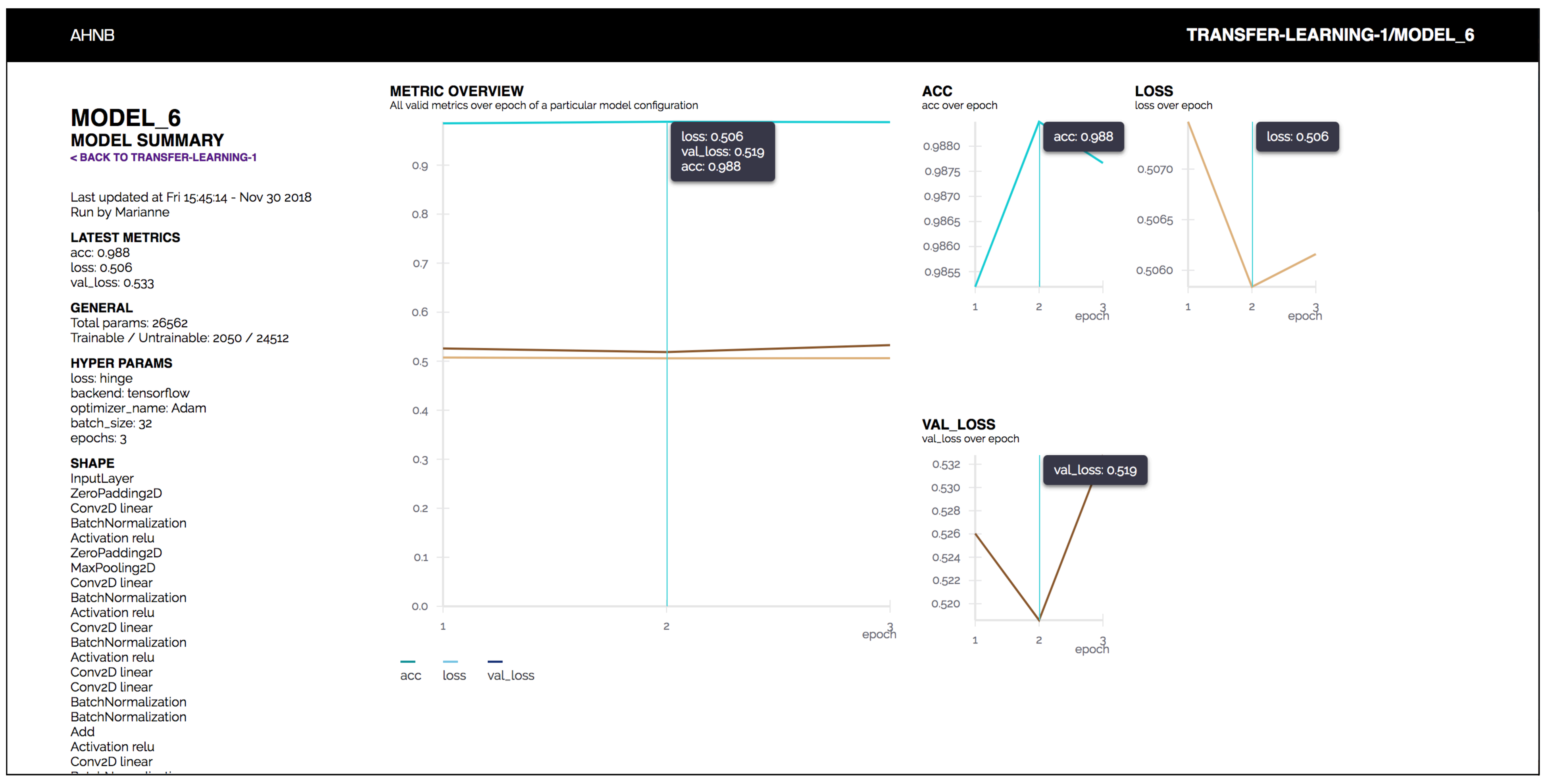


This modular data pipeline architecture makes it easy for users to slot our callback into either traditional python based machine learning environments or to computational notebooks. On initialization our callback collects the model configuration and any relevant non-model parameters (e.g. what hardware and backend is being used). Data is passed to a rest microservice, which prioritizes fast writes with acknowledgement of the limitations of the free tier on Heroku. Specfically we implement our web application in Flask on top of a PostGRES database (selected over comparable database options for it's low cost and fast writes). Finally data is presented to the user in the form a react based web app served from github pages (see above).

INTERFACE



The project summary view presents the user with a quick overview of the state of their projects experimental history as well as the foundations on which to create further explorations. We provide two types of analysis: contextualization of model results between models (table, scatterplot, timeseries) and contextualization of hyperparameters (parallel coordinates).



In the model summary view gives a detailed report about the performance of a particular experiment. The textual summary surfaces the top level metrics and a model summary, so that the user can quickly understand how one model might differ from another, and generate actionable insights therein. The performance of the model is described through a series of interactive time series charts.

EVALUATION

For initial validation of our tool, and to gain insight for the next phase of its development, we wrote and piloted a user study among 3 participants drawn in a convenience sample of persons with data science knowledge known to the researchers. These persons have taken courses in machine learning or data science at top U.S. research institution, and have used these skills in research projects.

We asked participants in a semi-structured interview format a series of 12 questions, 10 of which corresponded to the System-Usability-Scale methodology. In addition to free responses, these include Likert Scale (Strongly Agree/Agree/Neutral/Disagree/Strongly Disagree) responses to questions or assertions like:

"I think that I would like to use this sytem frequently."
" I found the various functions in this system were well integrated."

The size of our sample, and the method of gathering it, are a significant limitation. Aside from the issues of sampling bias, we anticipate that answers were colored by response bias, given that the interviewer was known to the subjects. It is likely ratings of our system would skew more favorably due to this effect. The results of this study are still pending.

CONCLUSION & FUTURE WORK

We view AHNB as a key stepping stone in any a number of significant directions. The relevance of AHNB and tools like it are here to stay for the foreseeable future, and until tools are developed that meet the needs of most organizations, this research area will remain active. In future work we would like to cover

Improved Experiment Organization
Enterprise Deployment
Browser Extension
Non-python Support
Automated Versioning
Automated Model Linting

See our paper for additional details