Activeloop: Implementing Auto Dataset Tuning

- by Suhaas Neel

Introduction

Project Name: Implementing Auto Dataset Tuning
Organization: Python Software Foundation
Suborganization: Activeloop

Name of Applicant: Suhaas Neel
University : Jawaharlal Nehru University(JNU), New Delhi

Time Zone: GMT + 5:30 (India)
Email : suhaasneel22@gmail.com
GitHub : https://gitHub.com/neel2299
LinkedIn : https://www.linkedin.com/in/suhaas-neel-a40296158/
CV : Resume Google Drive Link

About Project

Project Statement:
Auto Dataset Generation using Hub & hub.transform Generate experiments that generate datasets to improve the overall accuracy of the model.

Project Solution Introduction:
The idea of this project is to get a plugin for auto-tuning deep datasets for implementations in PyTorch/Tensorflow. This will entail researching and implementing different techniques aimed at a variety of datasets/tasks. Given there already are some existing libraries like auto-albument, our API should ideally be simpler and at the same time perform similar to the current

SOTA(State of The Art) libraries. Hub is aimed at being a go-to solution for handling large datasets used by giants like Google, and it would be highly appreciated if we could integrate this feature into Hub.

Project Solution:
Repeating mundane tasks in feature engineering is not a very rewarding experience, moreover, if it is something that can rather easily be automated, [1]*.

For image datasets, we have the option of getting the augmented data (including things for training. This helps the model generalize much better than before. Apart from giving the option of Augmenting data to the users of Hub, we can also provide an option to automatically generate novel unseen data using Conditional Generative Adversarial Networks (CGAN).
These will be available with hub.transform while we will make a separate API for tuning the hyperparameter (that is the fine-tuning of these transformations)
For text and numeric datasets we can follow a variety of methods including word2vec, one-hot encoding Tf-Idf standardizing, normalizing, ranks, capping

Solution Approach and timeline:

Community Interaction Period

- Fine tune the structure of the plugin based mechanism for Hub.transform
- Familiarize myself with hyperparameter tuning of data augmentation techniques for different datasets.
- Interact and work closely with the community, follow the PRs of my seniors in the community. (Doing so is both fun and makes me feel more significant about my place in the Hub project as a whole.)
- Try to test native alternatives to albumentations like torchvision.transforms.
- Read up on various image segmentation techniques and talk to mentor about its utility for Hub.
- Explore alternatives to Conditional Generative Adversarial Networks[2].

- Talk about some implementation details (for example - storage implementation for non-contextual and contextual transformations of text data.)
- Talk about Hub’s User-base with the mentors.

Week 1 and Week 2

- Implement image dataset tuning including augmentations with the libraries chosen in the community interaction period.
- Make util files for basic other image transformations like crop, rotation, rescaling the images, shear, etc. Implementing these for Hub will help users do basic image processing tasks on Hub itself.
- Make the documentation and tests

Week 3 and Week 4

-
- For tabular data implement features like auto detect categorical and numerical data using data samples to identify feature types.
- Start implementing text data transformations like word2vec, one-hot encoding Tf-Idf etc, and explore contextual embeddings like co-occurrence matrix for purely text data and tabular data seperately.
- For numerical data we can add support for algorithms like standardizing, normalizing, ranks, capping etc. The implementation details for tabular data will be formulated along with the mentor.

Week 5

- Catch up week - Complete any documentation and tests that are needed
- Complete any left out features
- Fine Tune the structure and model support (Chat with the mentor as to what model/architecture to use for tuning different dataset transform hyperparameters) of hyperparameter tuning deciding on a separate class hub.tune for example

Milestone - Complete the basic plugins for hyperparameter tuning up ahead

Week 6 and Week 7

- Implement the skeleton of the class and decide how models from different frameworks/libraries fit in using options among GridSearch, RandomSearch and bayesian optimization.
- Build the models for image datasets first and integrate it with the hub.tune class.
- Add a nice-looking output for ranking the strategy.
- Add necessary metrics to test the above

Week 8 and Week 9

- Build models for audio_datasets and integrate it.
- Start cleaning up documentation and tests.
- Work on making higher-level Hub APIs
- Include metrics including training loss/accuracy, recall, precision, validation loss, validation accuracy, IoU etc.

Week 10

- Cleanup Week, add tests and documentation.
- Keep the test coverage close to 100%.
- Make the API fully functional by tackling any left-over work from week 6 - week 9
  Milestone - Making Hub tune API ready to roll out

Week 11 and Week 12

- Make the Conditional Generative Adversarial Networks (CGAN). Make it accessible through hub.transform.
- Make a search hyperparameter method for CGANs.
- Implement any new ideas that I get during my time at Hub and implement them from week 11.

Final Milestone - The final plugin will include various transformation for images, text and numeric datasets accessible independently through hub.transform. The plugin will also include hub.tune and hyperparameter tuning features along with it.

Personal Details and Projects

Personal Details

I am currently pursuing an integrated (B.Tech+M.S) program at Jawaharlal Nehru University (JNU), New Delhi specializing in Electronics and Communications engineering for my bachelor's and Economics for my master's. Along with this, I am also pursuing a part-time Bachelor's in Data Science from IIT Madras which is something I am focussing more on right now. I will complete my program at JNU before Aug 2024 a semester after I complete my part-time Bachelor's from IIT Madras.

The **programming languages I am most comfortable** are python and C++. I am someone who normally likes to do something only when I feel that it has a chance to generate some impact. So fairly enough, my projects include things that can either establish something or have potential to be useful to somebody or myself.

Projects

I coauthored a research paper with my Mini-Project professor (i10-Index: 56, H index: 30) that explored the relation between Covid-19 and reverse migration and mobility. I was responsible for making plots and data stunts that were needed. I used stuff like **agglomerative clustering and disparity filters** to get the community structure and the backbone structures respectively in the Indian Migration and Mobility networks. I also scraped the Indian railway network data connecting data from a variety of websites to model the mobility network in the first place. This is when I came to love the field of Data Science [LINK].

Another project is when I was curious whether I could simplify my chess journey by **automating** the tedious part of reading chess e-books. That is get the FEN encoding for chess positions so I can straight off run to analyze the position. [LINK]

- For this I wrote a script to detect chess boards from ebook page images.
- I made a small dataset by gamifying the tagging process and making it simple using pygame (I know I could have used google API to get the image data but then I wanted some data, particularly for e-books so I used my scripts to get it and then tag it).
- Finally used a model from a kaggle notebook using transfer learning and fine tuned the model to my dataset.

Finally for something that **others apart from me** used.
I made a discord bot to display player rankings on online sites like chess.com in our club across different time formats. This was really simple but made 2 of the clubs I headed really active. And as you can see I am really passionate about chess.

Further Improvements after GSoC timeline
I wish to be involved in more issues and features in future. I would hence do my best to complete my own tasks as soon as possible so I can move on to other Issues. And if there is still work left in some other GSoC projects after the timeline, I would love to try my hand at them as well. While writing the proposals, I have become interested in many of these projects and would love to see them unfold or be directly involved.

Why Hub?
I never imagined I would be applying to get into GSoC this summer. It was only through some random article on “Towards Data Science” medium blog about productionizing your model, that I came across Hub.

I am interested in applications of machine learning and data engineering and my reasons for choosing hub out of other open-source tools in the domain is simply because Hub looked like a comparatively new library where contributions and people are needed to take it forward. I plan to stick with hub and become one of the major contributors.

Finally, to stress on how Hub will help me, after my love of data drew me in, just by going through the documentation and examples given in activeloop/examples, I feel like I know alot of things that I previously didnt. Given it is a platform for datasets, it has the potential to simplify using datasets for machine learning by providing
support for automated dataset exploration. Something like pandas_profiling and even more. Even this particular proposal has a lot of potential as in changing the workflow of the average data science team.
In terms of my long-term goals. It will help me well on my way to becoming a data architect :).

Contribution to Hub
I worked on the pretty-printing hub datasets and tensors. This looked like a really good first issue and the idea of having people see the product of your code **whenever they use Hub** was really encouraging. [LINK]
Another issue I have been working on is (hub.auto for audio and video htypes). I have yet to make a PR (have an idea for the solution ready though) as I have been reading up on Hub's codebase and other stuff needed for the GSoC proposals. [LINK]. Apart from these contributions I have also read the parts relating to this proposal and some others trying to dig into older PRs from older versions before Hub 2.0 to get an idea of how Hub previously worked.

Other Commitments:
Not for the next three months after may when I will have my end terms for my full-time degree and my current internship will end. The part-time degree in data science can be easily stopped and courses retaken next term given its flexible structure.