

Weekly Progress Report

Date: [11/06/2023]

Project: Practical Deep Learning for Coders

This weekly progress report outlines the activities and achievements for the week. The focus was on working with Jupyter notebooks, fast.ai library, and deep learning concepts.

- **Fastai:** A learning courses and resources that offers practical deep, it provides a hands-on approach to learning deep learning, emphasizing practical implementation and real-world applications, working with Jupyter notebooks, fastai, and deep learning involves hands-on coding, understanding concepts related to neural networks, and addressing challenges specific to data analysis and model training.
- **Jupyter Notebook:** Got to know about visualizations, explanatory text, data analysis, scientific computing, and machine learning tasks. ex. Python, R, PyTorch.
- **Deep Learning:** machine learning that focuses on training artificial neural networks with multiple layers to learn representations of data. exposers on various domains, including computer vision, natural language processing, speech recognition.
- **Challenges:** When working with Jupyter notebooks, facing challenges debugging code errors, managing dependencies, handling large datasets efficiently, and ensuring reproducibility. It's important to carefully organize and document code in notebooks to maintain readability and facilitate collaboration. Additionally, deep learning itself can present challenges such as selecting appropriate architectures, dealing with overfitting or underfitting, and optimizing model performance.

Reference:- https://www.youtube.com/watch?v=0oyCUWLL_fU&t=15085s

- **Introduction to Jupyter Notebooks:**

Explored the functionalities and benefits of Jupyter notebooks.

Installed Jupyter notebook using the Google Colab platform.

Learned how to run code cells, add explanatory text, and visualize outputs.

Explored various markdown and code cell features to enhance documentation and code readability.

- **Setup and Installation:**

Installed necessary packages and libraries using pip, including fastbook, duckduckgo_search, jupyter_contrib_nbextensions, and nbdev.

Configured the fastbook library to facilitate the learning process.

Set up the Azure search key for image searching.

Image Searching and Data Acquisition:

Utilized the fast.ai and duckduckgo_search libraries to search for and download images.

Learned how to search for images using specific keywords, such as "grizzly bear."

Explored different image sources, such as Bing and DuckDuckGo, for retrieving relevant images.

Downloaded a set of images related to different types of bears (grizzly, black, and teddy) for further analysis.

- **Data Preprocessing and Visualization:**

Processed the downloaded images using various techniques, such as resizing and cropping.

Used the PIL (Python Imaging Library) to open and manipulate images.

Verified the integrity of the downloaded images and handled any failed images.

Created a DataBlock object to organize and preprocess the image data.

Visualized a batch of images using the DataLoaders object to ensure the correctness of the preprocessing steps.

- **Model Training and Evaluation:**

Constructed a deep learning model using the resnet18 architecture.

Split the data into training and validation sets using a random splitter.

Fine-tuned the model using transfer learning and the fast.ai framework.

Evaluated the model's performance using error rate and confusion matrix metrics.

Analyzed and interpreted the model's top losses to gain insights into the model's behavior.

- **Deployment and Interactivity:**

Exported the trained model for deployment using the `learn.export()` function.

Created an interactive interface to classify user-provided images.

Integrated widgets and buttons to enable image uploads and classification.

Utilized the trained model to predict the class and probability of the uploaded image.

Displayed the prediction and probability to the user using interactive widgets.

- **Challenges Faced:**

Debugging code errors and resolving package dependencies during installation.

Ensuring the availability and quality of the downloaded images for training.

Understanding and applying the `fast.ai` library and its associated functions effectively.

Managing and handling the interactive interface components for image classification.

- **Lessons Learned:**

Jupyter notebooks provide an interactive and flexible environment for deep learning projects.

The `fast.ai` library offers powerful tools and abstractions to simplify deep learning tasks.

Image preprocessing is crucial for model training and requires careful handling and verification.

Transfer learning and fine-tuning can greatly improve model performance and training efficiency.

Creating interactive interfaces enhances user experience and facilitates model deployment.

- **Next Steps:**

Explore additional deep learning architectures and techniques for further experimentation.

Extend the image classification project to handle a wider range of object categories.

Dive deeper into advanced topics such as natural language processing or generative models.

In conclusion, this week's progress was focused on getting started with Jupyter notebooks, utilizing the fast.ai library for deep learning tasks, and working through the process of image classification. The challenges faced were successfully addressed, and valuable lessons were learned throughout the week's activities. The next steps will involve further exploration and expansion of the project.

Please find attached a PDF file containing a summary of the progress made during this week, including code snippets, explanations, and visualizations.

Throughout the week, several challenges were encountered and successfully addressed. These challenges included debugging code errors and resolving package dependencies during the installation process. Additionally, ensuring the availability and quality of the downloaded images for training posed a challenge. Understanding and effectively utilizing the fast.ai library and its associated functions required focused efforts. Lastly, managing and handling the interactive interface components for image classification demanded careful attention, Engaging in the project activities.

Thank you.

Sincerely, Shadab.A. Sheikh

WEEKLY PROGRESS REPORT

Project: Practical Deep Learning for Coders

•Introduction to Jupyter Notebooks

- Explored interactive coding, data analysis, and documentation in Jupyter notebooks.
- Utilized Google Colab for code execution, text explanations, and visualizations.

•Setup and Installation

- Installed required packages using pip.
- Configured fastbook and other libraries for a streamlined learning experience.

PROJECT: PRACTICAL DEEP LEARNING FOR CODERS

- Image Searching and Data Acquisition
 - Used fast.ai and duckduckgo_search libraries for image searching and downloading.
 - Obtained images of grizzly, black, and teddy bears for analysis.
- Data Preprocessing and Visualization
 - Preprocessed images using resizing, cropping, and integrity verification.
 - Visualized image data using DataBlock and DataLoaders.
- Model Training and Evaluation
 - Constructed a deep learning model with the resnet18 architecture.
 - Fine-tuned the model using transfer learning.
 - Evaluated model performance using metrics and analyzed top losses.
- Deployment and Interactivity
 - Exported the trained model and created an interactive interface.
 - Enabled users to upload and classify their own images.
 - Displayed predictions and probabilities using interactive widgets.
- **Challenges Faced:** Debugging code errors, managing image availability, learning fast.ai library.
- **Lessons Learned:** Understanding Jupyter notebooks, managing dependencies, utilizing fast.ai efficiently.