Style Guide

Deep Learning Diseases

Facebook

Style Guide

In our development process, as suggested in the lecture “Scrum Best Practices”, we have adopted Google's Python style guide as the primary standard for our coding practices. By utilizing Google's well-established guidelines, we ensure consistency, readability, and adherence to industry best practices in our Python codebase. This style guide (PEP8) covers various aspects, including naming conventions, code formatting, documentation, and more.

The following are the main guidelines or suggestions for our style guide:

- Consistency
  - Use meaningful and descriptive names for variables, functions, and classes.
  - Start variable and function names with a lowercase letter and use underscores for multi-word names (e.g., my_variable, calculate_average).
  - Use CamelCase for class names (e.g., MyClass, StudentRecord).

- Code Formatting
  - Follow consistent indentation using four spaces for each level of indentation.
  - Avoid excessive or unnecessary comments. Focus on writing self-explanatory code.

- Imports
  - Import packages and modules at the top of the Python file.
  - Place each import statement on a separate line.
  - Avoid using wildcard imports (from module import *).

- Documentation
  - Include clear and concise comments for functions and classes to provide clear explanations of their purpose, parameters, and return values.
  - Document complex code sections or algorithms with inline comments to aid understanding.

    - Use inline comments sparingly and focus on explaining why, not what, the code does.

- Organization
    - Organize code into logical sections or functions based on their purpose or functionality.
    - Keep related functions or methods together within classes.

- Readability and Maintainability
    - Write code that is easy to read, understand, and maintain.
    - Use appropriate variable names and avoid ambiguous or non-descriptive names.
    - Modularize code.
    - Avoid unnecessary code duplication by utilizing functions or reusable code blocks.

Style Guide (specific to Google Colab)

This style guide is intended for using Google Colab for Python development. It aims to provide guidelines for writing clean, readable, and efficient code in the Google Colaboratory environment. Of course, all of the above mentioned style guidelines apply to this as well (these are only additions).

- Imports
    - Import packages and modules only when needed in the following cell.
    - Install the modules before importing them in the next cell.

- Code organization
    - Include markdown cells at the beginning of the Colab notebook to provide an overview of the notebook's purpose, its main components, and any setup or installation instructions.
    - Include comments within code cells to provide explanations for complex or non-obvious code segments.

- Documentation
    - Use markdown cells to provide clear section headings and subheadings to improve readability.