

Q1

TensorFlow- TensorFlow is an open source machine learning platform made by Google that helps build systems that can learn from data. It's able to build models to train neural networks that classify images, analyse text to detect spam, and also helps with time series data. It uses keras as the design models by stacking building blocks. Tensor then trains and optimizes the data. And export, monitor, and deploy the models.

PyTorch- PyTorch is an open source machine learning library that is created by Facebook. It's mainly used for academia for research and prototyping. It stands out for its dynamic computation graph and Pythonic design. PyTorch is best suited for deep learning tasks where flexibility, readability, and rapid experimentation are important

Q2

Describe two case use of jupyter notebooks in ai development

Jupyter Notebooks provides an interactive, flexible, and visual environment for working with code, data, and models. It's ideal for exploring data, training data and visualizing the results. Developers use them to prototype ideas and document their work clearly.

Q3

How does spaCy enhance NLP tasks compared to basic Python string operations

SpaCy enhances NLP tasks by providing advanced, fast, and accurate language processing tools, which go far beyond basic Python string operations. Compared to basic Python string operations, which only handle simple tasks like splitting, replacing, or finding substrings, spaCy offers powerful, linguistically-informed features such as tokenization, part-of-speech tagging, named entity recognition (NER), and syntactic parsing. These features make it easier to extract meaningful information from text and build intelligent NLP applications.

Part 2

Compare Scikit-learn and TensorFlow in terms of target applications (e.g. classical ML vs deep learning).

Ease of use for beginners and community support

Scikit-learn and TensorFlow are both powerful machine learning libraries, but they cater to different types of tasks and user needs.

Scikit-learn is primarily used for classical machine learning algorithms such as linear regression, decision trees, random forests, and clustering. It's extremely beginner-friendly, with a simple and consistent interface that makes it easy to learn and apply. It's ideal for smaller datasets and projects where deep learning isn't necessary.

TensorFlow, in contrast, is designed for building and training deep learning models, including convolutional neural networks (CNNs), recurrent neural networks (RNNs), and transformers, making it suitable for tasks like image classification, speech recognition, and natural language processing.

Although TensorFlow has a steeper learning curve than Scikit-learn, its high-level API (`tf.keras`) significantly lowers the barrier for beginners. TensorFlow is also built for

scalability and performance, offering tools to deploy models to mobile, web, and production environments with GPU and TPU acceleration.

In terms of community support, both libraries have large and active user bases. Scikit-learn has strong support within the data science and academic communities, while TensorFlow, backed by Google, has a vast ecosystem with extensive documentation, tutorials, research papers, and integration with other tools like TensorBoard, TFX, and TensorFlow Lite. In summary, Scikit-learn is ideal for classical ML tasks and fast prototyping, while TensorFlow is better suited for advanced AI and deep learning projects at scale.