CHATBOT IN PYTHON

Phase 4: Development Part 2

Start building the project by performing different activities like feature engineering, model training, evaluation etc as per the instructions in the project.

Introduction:

At the most basic level, a chatbot is a computer program that simulates and processes human conversation (either written or spoken), allowing humans to interact with digital devices as if they were communicating with a real person.

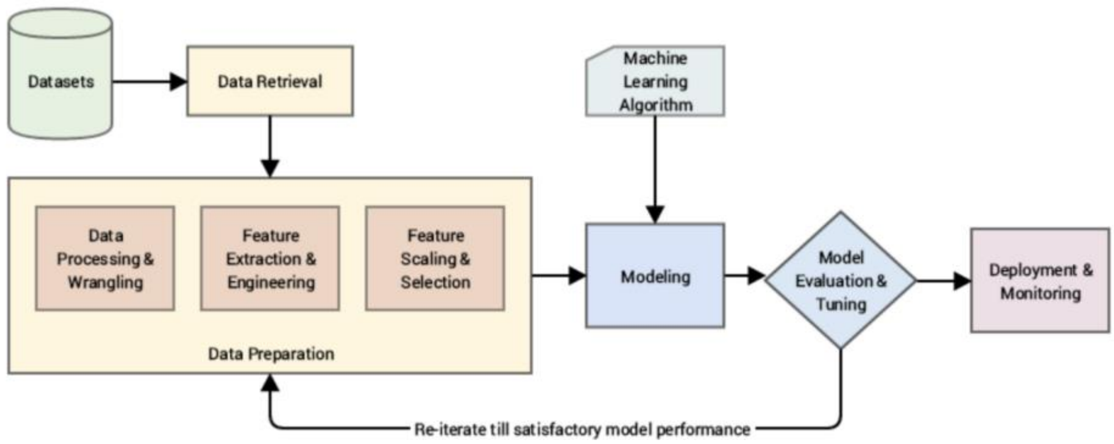
A major area where chatbots have long been used is in customer service and support, such as with various sorts of virtual assistants. Companies spanning various industries have begun using the latest generative artificial intelligence technologies to power more advanced developments in such areas.

FEATURE ENGINEERING:

Feature engineering is the process that takes raw data and transforms it into features that can be used to create a predictive model using machine learning or statistical modeling, such as deep learning.

it is the process of selecting, extracting, and transforming the most relevant features from the available data to build more accurate and efficient machine learning models.

Feature engineering techniques:



Feature engineering is the process of improving a model's accuracy by using domain knowledge to select and transform raw data's most relevant variables into features of predictive models that better represent the underlying problem. Feature engineering and selection aim to improve the way statistical models and machine learning (ML) algorithms perform.

The preprocessing steps that transform raw data into features make up the feature engineering pipeline. These features are used in predictive models and other machine learning algorithms. Predictive models comprise an outcome variable and one or more predictor variables. The feature engineering process is what creates, analyzes, refines, and selects the predictor variables that will be most useful to the predictive model. Some machine learning software offers automated feature engineering.

Feature engineering in machine learning includes four main steps: feature creation, transformation, feature extraction, and feature selection. During these steps, the goal is to create and select features or variables that will achieve the most accurate ML algorithm.

Feature creation:

Feature creation, sometimes just called feature engineering, is the process of training a machine learning model by using existing data to construct new features

Feature extraction:

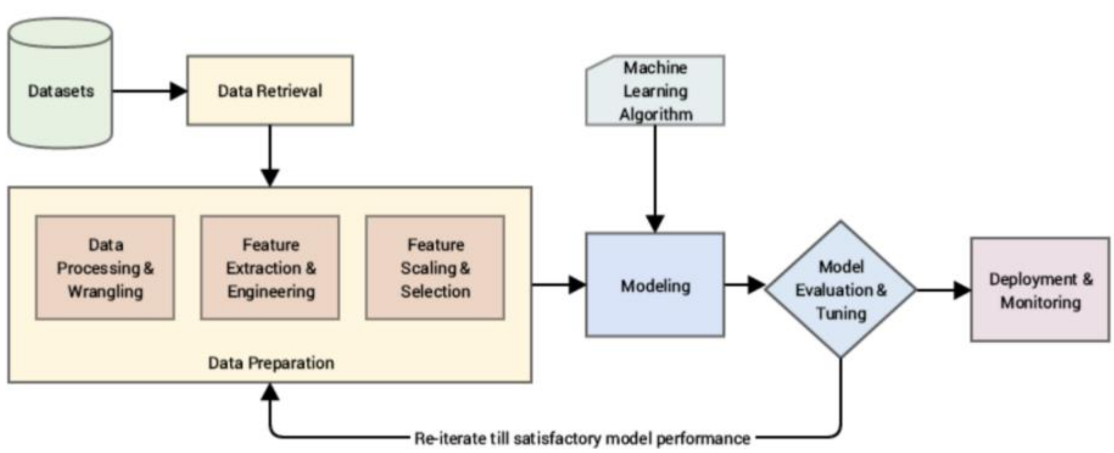
It involves extracting and creating new variables automatically from raw data. The goal of feature extraction is to reduce data volume to a more manageable modeling set automatically. Some feature extraction techniques include cluster analysis, edge detection algorithms, principal components analysis, and text analytics.

Feature extraction is used when predictive modeling algorithms cannot directly model observations because they are too voluminous in their raw state. For example, audio, image, tabular, and textual data may have millions of attributes algorithms may not be effective for this unstructured data, unsupervised learning can be very useful.

Feature extraction creates features from the existing ones and then discards the original features to reduce the total number of features in a dataset. The new, reduced set of features can summarize the original set of features and the information they contain. The newer, smaller dataset can much more easily be modeled.

For tabular data, feature extraction methods might include unsupervised clustering methods and projection methods such as principal component analysis. For image data, techniques might include edge or line detection. All feature extraction methods solve for the issue of dimensional data that is unmanageably high and work automatically.

During predictive model development, feature selection is the process of selectively reducing the number of input variables. This is always desirable to reduce the computational costs of modeling, and it often enhances model performance.



Feature Engineering Automation:

Feature engineering for machine learning might include: identifying new sources of data, applying new business rules, or reshaping data. Typically, this is an extended manual process that relies heavily on expertise, manipulation of data, intuition, and domain knowledge. The tedium and resource-intensive nature of the process can limit the final features—as can mere human subjectivity.

Automating this process creates many hundreds or even thousands of candidate features automatically from a dataset. The data scientist can then select the best options and use them for training data.

Automated feature engineering is in no position to replace data scientists – its main strength lies in reshaping data.

In this way, it allows data scientists to engage more with tasks that demand experience, creativity, and business domain feature knowledge. Automating feature engineering allows data scientists to focus on delivering robust models into production, interpreting complex data, creative feature engineering, and other more valuable parts of the machine learning pipeline.

MODEL TRAINING:

Artificial intelligence (AI) training is the process of teaching an AI system to perceive, interpret and learn from data. That way, the AI will later be capable of inferencing—making decisions based on information it's provided.

This type of training requires 3 important components: a well-designed AI model; large amounts of high-quality and accurately annotated data; and a powerful computing platform.

Properly trained, an AI's potential is nearly limitless. For example, AI models can help anticipate our wants and needs, autonomously navigate big cities, and produce scientific breakthroughs.

It's already happening. You experience the power of well-trained AI when you use Netflix's recommendation engine to help decide which TV show or movie you want to watch next.

Or you can ride with AI in downtown Phoenix, Ariz. It’s home to the robotaxi service operated by Waymo, the autonomous-vehicle developer owned by Google’s parent company, Alphabet.

And let’s not forget ChatGPT, the current belle of the AI ball. This year has seen its fair share of fascination and fawning over this new generative AI, which can hold a remarkably human conversation and regurgitate every shred of information the internet offers—regardless of its accuracy.

AI can also be used for nefarious purposes, such as creating weapons, methods of cybercrime and tools that some nation states use to surveil and control their citizens. As is true for most technologies, it’s the humans who wield AI who get to decide whether it’s used for good or evil.

3 steps to train AI:

AI training is technically demanding. But years of research aided by the latest technology are helping even novice developers harness the power of original AI models to create new software like indie video games.

The process of training enterprise-level AI, on the other hand, is incredibly difficult. Data scientists may spend years creating a single new AI model and training it to perform complex tasks such as autonomous navigation, speech recognition and language translation.

Assuming you have the programming background, technology and financing to train your desired type of AI, the 3-step process is straightforward:

> **Step 1: Training.** The AI model is fed massive amounts of data, then asked to make decisions based on the information. Data scientists analyze these decisions and make adjustments based on the AI output’s accuracy.
>
> **Step 2: Validation.** Trainers validate their assumptions based on how the AI performs when given a new data set. The questions they ask include: Does the AI perform as expected? Does the AI need to account for additional variables? Does the AI suffer from overfitting, a problem that occurs when a machine learning model memorizes data rather than learning from it?
>
> **Step 3: Testing.** The AI is given a novel dataset without the tags and targets initially used to help it learn. If the AI can make accurate decisions, it passes the test. If not, it’s back to step 1.

Future of AI Training

New AI training theories are coming online quickly. As the market heats up and AI continues to find its way out of the laboratory and onto our computing devices, Big Tech is working feverishly to make the most of the latest gold rush.

One new AI training technique coming to prominence is known as Reinforcement Learning (RL). Rather than teaching an AI model using a static dataset, RL trains the AI as though it were a puppy, rewarding the system for a job well done.

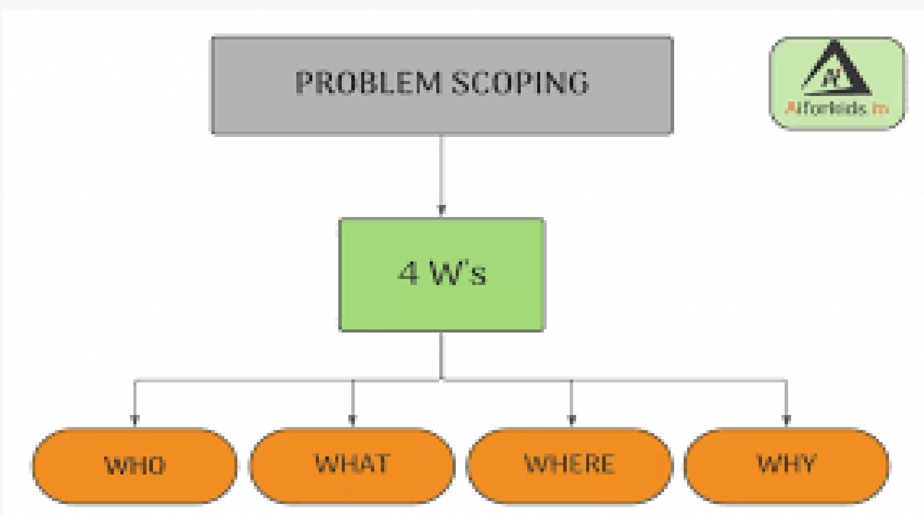
Instead of offering doggie treats, however, RL gives the AI a line of code known as a “reward function.” This is a dynamic and powerful training method that some AI experts believe will lead to scientific breakthroughs.

Advances in AI training, high-performance computing and data science will continue to make our sci-fi dreams a reality. For example, one AI can now teach other AI models. One day, this could make AI training just another autonomous process.

Will the next era of AI bring about the altruism of Star Trek or the evil of The Matrix? One thing’s likely: We won’t have to wait long to find out.

EVALUATION:

Evaluation is the method of understanding the reliability of an API Evaluation and is based on the outputs which are received by feeding the data into the model and comparing the output with the actual answers.



In the 1960s, AI research gained momentum with the development of the first AI programming language, LISP, by John McCarthy. Early AI systems focused on symbolic reasoning and rule-based systems, which led to the developmentof expert systems in the 1970s and 1980s. Evaluation provides a systematic method to study a program, practice, intervention, or initiative to understand how well it achieves its goals. Evaluations help determine what works well and what could be improved in a program or initiative.

Evaluation is a process that critically examines a program. It involves collecting and analyzing information about a program's activities, characteristics, and outcomes. Its purpose is to make judgments about a program, to improve its effectiveness, and/or to inform programming decisions (Patton, 1987).