**How Does ChatGPT Work?**

In this video, we explore the workings of ChatGPT, an incredibly fast-growing app.

Launched on November 30, 2022, ChatGPT amazingly reached 100 million monthly active users within just two months. How does this achievement compare to other apps? Imagine it took Instagram two and a half years to reach the same milestone.

Clearly, ChatGPT is the fastest-growing app in history.

Now, let's delve into how ChatGPT works.

At the core of ChatGPT lies an LLM, which stands for Large Language Model. The current LLM utilized by ChatGPT is known as GPT-3.5. While there is not much technical information available yet about the latest GPT-4 model, ChatGPT has the potential to incorporate it as well.

So, what exactly is a Large Language Model? Essentially, it is a type of neural network-based model that undergoes training with massive amounts of text data to comprehend and generate human language. By analyzing the statistical patterns and relationships between words in the training data, the model learns to predict subsequent words one at a time.

An LLM is often characterized by its size and the number of parameters it possesses. The largest variant of GPT-3.5 consists of a staggering 175 billion parameters spread across 96 layers, making it one of the most extensive deep-learning models ever created.

To facilitate processing, the input and output of the model are organized using tokens. Tokens are numerical representations of words or more precisely, segments of words. Numbers are preferred for tokens as they can be processed more efficiently than words.

GPT-3.5 underwent training using a substantial portion of Internet data, with the source dataset containing 500 billion tokens. Another way to perceive this is that the model was trained on hundreds of billions of words. The objective of the training was to predict the next token given a sequence of input tokens, enabling the model to generate text that follows grammatical rules and semantically resembles the internet data it was trained on.

However, without proper guidance, the model can also generate outputs that are false, toxic, or reflect harmful sentiments. Despite this drawback, the model already exhibits usefulness, but in a controlled manner. By providing carefully engineered text prompts, the model can be "taught" to perform natural language tasks.

This practice has given rise to a new field known as "prompt engineering." To enhance safety and enable chatbot-like question-and-answer capabilities, the model undergoes further fine-tuning to become the version used in ChatGPT. This fine-tuning process is called Reinforcement Learning from Human Feedback (RLHF).

OpenAI has explained how they employed RLHF on the model, but it can be challenging for non-machine learning experts to comprehend. To aid in understanding, let's draw an analogy. Envision GPT-3.5 as a highly skilled chef who excels in preparing culinary delights.

The chef is initially trained with a large dataset of recipes and techniques. Feedback from real people is collected to create a new dataset. The process includes creating a comparison dataset where multiple dishes are prepared and ranked based on taste and presentation.

A reward model is then created, guiding the chef's understanding of customer preferences. The model is trained using Proximal Policy Optimization, similar to the chef comparing different dishes and improving their skills.

This iterative process is repeated based on customer feedback, resulting in the chef becoming better at satisfying customer preferences.

Another example is looking at GPT-3.5 as a master trainer who excels at making professionals even better through personalized training. Like a chef learning recipes, GPT-3.5 starts by absorbing loads of training info, becoming an expert in the field it's training for.

To improve, it uses feedback from real learners, like a trainer adjusting methods based on learner input. This feedback loop helps GPT-3.5 refine how it trains. Similarly, GPT-3.5 is fine-tuned with RLHF by gathering feedback, creating a reward model, and iteratively improving the model's performance using PPO. This allows GPT-3.5 to generate tailored responses for specific user requests.

When using ChatGPT, the prompt is fed into the model, taking into account the context of the conversation. Conversational prompt injection ensures context awareness.

Primary prompt engineering provides instructions for a conversational tone, and moderation API is used to warn or block unsafe content.

The generated result is also likely to be moderated before returning to the user.