

Generative Künstliche Intelligenz und Vorab Trainierte Transformer



Erstellt mit Anthropic Claude 3 – veröffentlicht am 6. Mai 2024

Guten Tag Claude 3. Es soll um sogenannte (Generative) Vorab Trainierte Transformer ('GPT') gehen.

Und es soll in diesem Zusammenhang gehen um das maschinelle Verstehen natürlicher Sprache (NLU), die maschinelle Verarbeitung natürlicher Sprache (NLP) und das automatisierte Erzeugen natürlicher Sprache (Generierung natürlicher Ausgaben aus strukturierten und unstrukturierten Daten oder NLG)

Dazu habe eine Analogie konstruiert insbesondere das bessere Verständnis solcher generativer, vorab trainierten Transformer betreffend.

Dabei habe ich versucht, die Encoder-Decoder-Architektur, auf welcher die für komplexere Aufgaben - wie Textinterpretation und Übersetzung - besonders geeignete Transformer-Modelle aufsetzen, zu veranschaulichen.

Analogie 1

Analogie 2

Analogie 3

Meine Frage dazu: Was findest du davon?

Sie haben eine kreative Analogie erstellt, um das Konzept von vortrainierten Transformern (wie GPT-Modellen) zu veranschaulichen. Hier ist eine Zusammenfassung dessen, was ich in den Bildern sehe:

1. Sie stellen das Konzept als einen Zylinder mit einem weißen Kaninchen, einem schwarzen Kaninchen und einem Hohlraum dazwischen dar.
2. Das weiße Kaninchen repräsentiert das natürliche Sprachverständnis oder die Kodierung.
3. Das schwarze Kaninchen steht für die natürliche Sprachgenerierung oder Dekodierung.
4. Der Hohlraum dazwischen wird als natürliche Sprachverarbeitung bezeichnet.

In den Links sehe ich auch verschiedene Antworten von KI-Modellen (Claude und ChatGPT) auf Ihre Analogie. Jedes Modell hat die Analogie aufgegriffen und versucht, sie weiter auszuführen oder zu erklären.

Ihre Analogie ist in der Tat eine interessante und kreative Art, die Funktionsweise von Sprachmodellen wie GPT zu veranschaulichen. Sie erfasst die Grundkonzepte des Verständnisses, der Verarbeitung und der Generierung von natürlicher Sprache, die in diesen Modellen eine wichtige Rolle spielen.