KI - Einsichten - Teil 2 **KI-Technik auf einen Blick** (Februar 2024)

Was Ihr schon immer über die Künstliche Intelligenz wissen wolltet.

Kurz und wesentlich - und vorab mit einem Hinweis für euch und eure Gäste:

Unterkunft mit Direkt-Kontakt deutschlandweit: www.finde-unterkunft.de

Und nun zur KI:

Am 30. November 2022 veröffentlichte OpenAI (openai.com) das große Sprachmodell ChatGPT 3.5 (GPT: Generative Pre-trained Transformer). Die Fähigkeiten dieses Sprachmodells waren verbunden mit Faszination, Ängsten und lösten einen KI-Hype aus. Diese Künstliche Intelligenz setzt eine neue technische Revolution auf die Schienen und ist bereits dabei, insbesondere die Berufs- und Bildungswelt in atemberaubender Geschwindigkeit zu verändern.

Der Teil-2 der KI-Einsichten bietet eine kleine Sammlung von Details und einen Blick auf Mitteilungen, eigene Erkenntnisse und Fragen wie:

- Was steckt im Kern hinter der KI?
- Wie trainiere ich die KI?

Es gibt in den Medien und auf dem Buchmarkt zahlreiche Beiträge und Veröffentlichungen zu dem Thema KI. In ihnen stecken nicht selten Wiederholungen und Erläuterungen, die nicht wirklich eine bessere Sicht im geheimnisvollen KI-Nebel verschaffen. Manchmal wird der Nebel durch darin steckende Unverständlichkeiten und Langatmigkeiten sogar noch verstärkt.

Sicherlich entwickelt sich erst durch die eigene Programmierung kleiner neuronaler Netze und durch deren Training ein Gefühl der Gewissheit in der Art: „Ja, das funktioniert ja tatsächlich."

Jedoch bereits bei dem Versuch, diese Gewissheit gedanklich auf ein Netzwerk mit 100 Billionen Neuronen zu übertragen, kann einem ganz schwindelig werden. Und bei der Frage: „Seid ihr auch alle da?" ergäbe sich beim Abzählen von 100 Billionen Neuronen folgendes Problem: Dafür reicht ein Menschenleben zeitlich gewiss nicht aus.

Es ist wohl eine Tatsache, dass der Erfolg von neuronalen Netzen im unbeirrten Verfolgen von scheinbar irrwitzigen Ideen einiger Weniger liegt. Vieles von dem, was die KI heute im Stande ist zu leisten, basiert kaum auf wissenschaftliche Erkenntnisse und Methoden als vielmehr auf dem Prinzip von Try und Error. So hat sich über die Jahre ein großer Erfahrungs- und Erkenntnisschatz in den Händen und Köpfen von wenigen Insidern der üblichen großen Tech-Konzerne angesammelt. Dieses Knowhow und die immense Rechenleistung, der es bedarf, um die großen Sprachmodelle zu berechnen und funktionieren zu lassen, sind zurzeit ausschließlich dort vorhanden.

Die nun folgenden Seiten geben einen kleinen zusammenfassenden Einblick und Überblick in das, was die Künstliche Intelligenz im Kern und drum herum bildet und bewegt - nicht mehr und nicht weniger.

Neuronale Netze

Neuronale Netze bestehen aus künstlichen Neuronen gleicher Struktur, die in der Regel oder aus funktionalen Aspekten, untereinander mittels gewichteter Vektoren verbunden sind. **GPT 3.5** besteht aus 175 Millionen solcher Neuronen. Bei GPT-4 sind es bereits 100 Billionen Neuronen. In diesen Netzen steckt das antrainierte Antwort-Verhalten. Die neuronalen Netze sind sogenannte **Feedforward**-Netze. Diese lassen keine Programmschleifen (algorithmische Iterationen) zu. Neuronale Netze liefern immer Antworten. Jedoch müssen diese nicht unbedingt immer wahr sein (bei KI als **Halluzinationen** bezeichnet). Letzteres ist geradezu menschlich.

Neuronale Netze Details

Unter diesem Punkt sind einige interessante Details bezüglich der Netz-Architektur und Funktionalität von GPT gelistet.

1) Neuronale Netze haben eine begrenzte Aktualität: Die Daten von GPT-3.5 haben einen Aktualitätsstand vom September 2021. Bei GPT-4 liegt die Aktualitätsgrenze im April 2023.

2) An der Token-Erzeugung von GPT-3.5 können bis zu 400 Neuronen-Schichten beteiligt sein.
3) Millionen der Neuronen sind untereinander vernetzt. Das beinhaltet, dass derart vernetzte Neutronen mit 175 Milliarden anderen Neuronen in einer gewichteten Beziehung stehen.
4) Um einen neuen Antwort-Token zu erzeugen, müssen unter Umständen bis zu 400 Schichten durchlaufen und zig 175-Milliarden-schwere Matrix-Operationen ausgeführt werden. Diese Matrix-Operationen werden durch hochspezialisierte, super schnelle Prozessoren ausgeführt, den GPUs (Grafik-Prozessoren). Bei Betrachtung dieser Zahlen überrascht es trotz der GPUs, in welch unglaublich kurzer Zeit die KI-Systeme Antworten auf gesendete Prompts generieren.

Die Funktionalität von künstlichen neuronalen Netzen ist generell eng verbunden mit folgenden Begriffen:

Gradientenverfahren:
Zur Minimierung der neuronalen Netz-Fehler. Das Verfahren kommt aus dem Operation Research und beruht wesentlich auf Anwendung der partiellen Differenzialrechnung. Im Rahmen der KI wird ein auf die KI spezialisiertes Gradientenverfahren eingesetzt.

Tensoren:
Bei der Künstlichen Intelligenz werden Matrizen als Tensoren bezeichnet. Diese Tensoren haben die programmiertechnische Eigenschaft, dass sie explizit für die Ausführung von Matrix-Operationen speziellen Hardware-Einheiten zugewiesen werden können, beispielsweise den schnellen GPU-Prozessoren.

Sigmoid-Funktion:
Diese Funktion wird in neuronalen Netzen als Aktivierungsfunktion verwendet. Ihr Ergebniswert liegt im Wertebereich von 0 bis 1. Steigt beispielsweise der Aktivierungswert eines Neurons auf 0,5 oder darüber, so sendet das Neuron ein Signal an die nachfolgende Neuronenschicht.

Sprachmodelle

Durch die KI bekannt geworden sind die großen Sprachmodelle oder engl. Large Language Modells (**LLM**). Diese befähigen KI-Systeme beinahe in Ist-Zeit natürliche Sprachen zu verarbeiten, Bilder zu malen und vieles mehr.

Beispiele für große Sprachmodelle:

CPT:	OpenAI / Microsoft - Generative Pre-trained Transformer
BERT:	Google - Bidirectional Encoder Representations from Transformers
Llama2:	Sprachmodell von Meta (Facebook)
Luminous:	Aleph Alpha, DE, Ziel: Erklärt Ergebnisse und Quellen der Modelle

Hugging-Face-Ökosystem:
Open Source Modelle, die kostenfrei heruntergeladen werden können.

Sprachmodelle und Natural Language Processing (NLP) sind eng miteinander verbunden. Die großen Sprachmodelle sind sehr leistungsfähig und können viele Dinge, zu denen Menschen fähig sind, beispielsweise: Fragen beantworten, Geschichten erzählen, Texte zusammenfassen, Texte variieren, rechnen, höhere Mathematik, programmieren, sich mit Menschen unterhalten (Chats), Kunst schaffen und kreativ sein, Videos und Filme erstellen und interpretieren.

Token

Alle Sätze (Wort-Sequenzen) vom Input-**Prompt** werden in Token verwandelt. Token können ganze Wörter, Wort-Teile, Satzzeichen oder Symbole sein.

Wie viele Token ergibt dieser wundervolle Satz?
Wie|viele|Token|erg|ibt|dieser|wund|erv|olle|Satz|?|

Im Beispiel oben sehen wir, dass aus 7 Wörtern und dem Satzzeichen insgesamt 11 Token werden. Ein Erfahrungswert ist es hierbei, dass sich beim GPT-**Tokenizer** bei 75 Wörter durchschnittlich 100 Token ergeben.

Die Token spannen den semantischen Bedeutungsraum bei GPT und anderen Sprachmodellen auf.

GPT-Tokenizer: https://platform.openai.com/tokenizer

Zahlenwerk

Computer können nur mit Zahlen rechnen. So ist bei GPT jedem Wort ein Zahlenwert zugeordnet. Ebenfalls aus Zahlen bestehen die Beziehungsgewichte und Token-Werte (Zahlen mit bis zu 18 Nachkommastellen).

Einbettung (Embedding)

Bei einer Einbettung wird die Bedeutung, die Essenz von Begriffen, in einem Array von Zahlen dargestellt und so ein hochdimensionaler Bedeutungsraum aufgespannt. Wörter mit einer ähnlichen Bedeutung liegen im Bedeutungsraum nahe beieinander, in einer Art der Nachbarschaft. Synonyme liegen also bei dieser Methode sehr dicht beieinander.

Die Einbettung von Bildern erfolgt ähnlich. Beispielsweise, was ist sichtbar auf einem Bild – Hund, Katze, Maus, Haus oder Flugzeug? Auch hier kann in ähnlicher Weise ein entsprechender Bedeutungsraum für sichtbare Dinge aufgespannt werden.

Das alles riecht schon nach viel notwendiger Erfahrung auf dem Weg zum Erfolg - eben nach Erfahrung im Umgang mit den KI-Dingen und nach viel Try and Error.

Hyper-Parameter

Bei GPT oder auch bei anderen KI-Systemen gibt es Hyper-Parameter, die einen Einfluss auf das Antwort-Verhalten ausüben. Einer dieser Parameter ist der sogenannte Temperatur-Parameter. Hat dieser einen Wert von Null, so wird die Antwort auf einen inhaltlich gleichen Prompt immer identisch sein. Bei einem Wert von beispielsweise 0,8 werden die Antworten bereits spürbar variantenreicher. Bei einem noch höheren Wert nimmt die Variation immer weiter zu. Das hat seine Ursache darin, dass zufallsgesteuert bei steigendem Wert zunehmend nicht ausschließlich der best-passende Token der Antwort beigefügt wird, sondern durchaus mal ein anderer, naheliegender Token. Daraus ergibt sich beiläufig eine entsprechende Wechselwirkung, dadurch, dass sich daraus folgend auch der Antwort-Kontext ein wenig verschiebt.

Bei einigen KI-Systemen kann der Temperatur-Wert von außen eingestellt werden und zusammen mit dem Prompt an das System gesendet werden.

Transformer

Diese Art neuronaler Netze hat die Sprachverarbeitung erheblich verbessert. Durch einen Aufmerksamkeitsmechanismus (Attention) bringen die Strukturen alle eingegebenen Inhalte miteinander in Verbindung - unabhängig von ihrer Distanz zueinander. Sie geben Teilen der Text-Sequenz mehr oder weniger Aufmerksamkeit. Am Ende liefert der Transformer eine Wahrscheinlichkeits-verteilung für jedes Wort aus seinem Sprachschatz.

Die Algorithmen der Transformer für Natural Language Processing (NLP) beachten also unter anderem die Reihenfolge der Wörter sowie deren Kontext. Ein Transformer besteht aus mehreren Schichten neuronaler Netze:

- Input Embedding
  - Positional Encodings: Wortposition im Satz per numerischer Werte
  - Input Embedding: Wort als numerisches Symbol
- Self-Attention
  - Aufmerksamkeitsmechanismus
  - alle Wörter in einer Sequenz sehen und den Kontext dieser Wörter erkennen
  - Erkennen: Welche Teile der Input-Daten sind am relevantesten für den Output
  - Beziehungen zwischen verschiedenen Elementen der Inputdaten berechnen
  - Encoder und Decoder

Um sich vorzustellen, um welche Größenordnungen es sich bei solchen Transformern programmtechnisch handelt, sei hier das Transformer-Modul von GPT-3 erwähnt. Die Kerndatei ist pytorch_model.bin hat eine Größe von ca. 3,7 Gigabyte oder eben 3.700 Megabyte.

Das Trainieren von komplexen Transformern ist sehr Ressourcen-intensiv. So dauerte das Trainieren von GPT-3.5 auf spezialisierten Hochleistungsrechnern ganze 34 Tage Rechenzeit.

Transformer Details

Unter diesem Punkt sind einige interessante Details bezüglich der Transformer-Architektur und -Funktionalität gelistet.

1) Für den Einbettungsvektor werden Token-Werte und Token-Positionen addiert. Das erfolgt nicht auf Grundlage wissenschaftlicher Erkenntnisse, sondern auf Grundlage von Erkenntnissen und Erfahrungen. Dies sei ein Beispiel für die Erkenntnis- und Erfolgsmethode des Try and Error bei der KI-Entwicklung.
2) Der Transformer von GPT-3 hat 96 sogenannter Aufmerksamkeitsblöcke. Innerhalb dieser Blöcke gibt es Attention-Heads, die unabhängig voneinander, auf Werten der Eingabe-Vektoren operieren. Diese können Bezüge oder Assoziationen zu anderen Token der Sequenz herstellen und damit deren Kontext in der Sequenz bewerten. Der Input durchläuft so Block um Block, Schicht um Schicht – immer weiter vorwärts, nie zurück. Ein Token funkt nur einmal oder nicht. Es funktioniert und wurde im Try and Error Verfahren Step by Step funktional optimiert.
3) Nur auf der äußersten Ebene von GPT gibt es eine Art Feedback-Schleife, die auf die Eingabe und selbst erzeugten Token wiederholt zugreift.
4) Es ist zu vermuten, dass Transformer eine neuronale Codierung von Eigenschaften der menschlichen Sprache schaffen, die für uns erst einmal wenig verständlich und sehr komplex ist – am Ende jedoch funktioniert.

Deep Learning

Beim Deep Learning wird einem KI-System in der Regel eine riesige Menge an Daten zugeführt. Während des Trainings werden die Verbindungen (Gewichte) zwischen den Parametern (Neuronen/Knoten) neu errechnet (optimiert). Durch dieses Training wird Wissen in dem Netz hinterlegt. Anschließend ist das trainierte neuronale Netz zum Beispiel befähigt, Sprachen zu übersetzen oder Fragen zu beantworten. Die Qualität der Übersetzungen und Antworten hängt wesentlich auch von der Qualität der Trainingsdaten ab.

Beispiele Trainingsmethoden:

Finetuning:	Vortrainiertes Sprachmodell mit eigenen Daten für spezielle Aufgaben nachtrainieren.
Few-Shot Learning:	Anpassung eines Modells an neue Problemlösungen anhand Nutzung von nur wenigen Beispielen.
Prompt Engineering:	Durch Anfragen die Ergebnisse optimieren.
Anhand Beispiele:	Anpassung mit manuell erzeugten Beispielen und Bewertung durch Menschen.

Transfer Learning

Das Transfer Learning beinhaltet einen Einsatz von vortrainierten Sprach-Modellen. Die vortrainierten Modelle können mit eigenen Datensätzen weitertrainiert werden. Mittels dieser Methode kann der notwendige Trainingsaufwand erheblich reduzieren.

Training mit Texten

Eine Methode, ein neuronales Netz mit Texten zu trainieren ist es, in Sätzen eine Wort-Lücke zu lassen und das System das best-passende Wort für die Lücke vorhersagen zu lassen. Die englische Sprache beispielsweise verfügt über zirka 50.000 Wörter. Bei GPT ist jedem dieser Worte ein Zahlenwert zugeordnet (z.B. das Wort *the* hat den Wert 914). Jedoch sind nur zirka 3.000 Token ganze Wörter, die Restlichen sind Wort-Fragmente.

Eine Grundlage für die Trainingsdaten bietet beispielsweise das Internet mit seinen Milliarden frei verfügbaren Texten (**Big Data**). Mit Hilfe dieser vielen Beispieltexte ergeben sich Ranking-Listen mit entsprechend hohen Wahrscheinlichkeitswerten für die best-passenden Lückenfüller. Da auch der Text-Kontext eine wichtige Rolle spielt, wird dieser Kontext ebenso in den Einbettungsvektoren berücksichtigt. Hinweis: GPT arbeitet mit Token statt mit Wörter. Zwecks Kontext-Verständnis hier nachfolgend einige Beispiele.

Beispiel für die Bedeutung der **Wort-Reihenfolge**

a) Du fährst immer besser Fahrrad.
b) Besser du fährst immer Fahrrad.

Beispiel für die **Wortbedeutung** im Kontext
Hinweis: Im bereinigten Text sieht der Rechner die Wörter in Kleinschreibung.

a) Wir fliegen heute nach Berlin.
b) Im Winter gibt es wenige Fliegen in Berlin.
oder
a) Du kannst jetzt den Hahn aufdrehen!
b) Der Hahn kräht heute aber spät.

Mögliche Learning-Methoden bei LLMs:
- Text-Inhalte aus dem Web zuführen und anderen digitalen Quellen (z.B. eBooks)
- Training mit Hilfe von Text-Lücken
- Qualität-Feedback durch Menschen (Antwort-Bewertung)

Training mit Bilddaten

Ein Training eines KI-Systems im Rahmen der Computer-Vision (Bildverarbeitung) erfolgt in der Regel durch eine große Anzahl von markierten und etikettierten Bildern in möglichst vielen Variationen. Die Anzahl Bilder kann hierbei in die 100-Tausende gehen. Auch ein ständiges Wiederholen von denselben Bild-Sequenzen gehört zum Training. Diese Wiederholungen werden als **Epochen** bezeichnet. In jeder nachfolgenden Epoche befindet sich das KI-System in einem höher trainierten, leistungsfähigeren Gesamtzustand. Das sind zumindest Ziel und Erfolgshoffnung beim Training.

Ein gewichtiges Beispiel für die Verarbeitung von Bildern steckt im Thema **autonomes Fahren** bei Fahrzeugen.

Beispiel: **Der Amazon-Clouddienst AWS** für das autonome Fahren nimmt Fahrt auf (Februar 2024). Ein Training, das bisher Jahre dauerte und Millionen von gefahrenen Meilen erforderte, dauert jetzt per KI und Simulationen nur noch Stunden oder Minuten.

Einbeziehung externer Datenquellen

Bei den großen Sprachmodellen (LLM) können externe Datenquellen für das Erstellen eines eigenen spezialisierten KI-Systems mit angebunden werden. Dieses Prinzip wird als **Retrieval Augmented Generation** (**RAG**) bezeichnet. Dadurch werden die Fähigkeiten von LLMs gezielt erweitert.

Durch das Einbeziehen externer Datenquellen ergeben sich folgende Vorteile:
- Eigene spezialisierte Texte sind mit Grundlage für die Prompt-Beantwortung
- Es können somit auch ganz aktuelle Daten integriert werden
- Die Kosten für die System-Entwicklung reduzieren sich deutlich
- Die Kosten für das Training reduzieren sich erheblich

Einbindung externer Datenquellen

Für eine Einbindung externer Datenquellen per **Retriaval Augmented Generation** ist eine Aufbereitung der externen Daten notwendig, um diese zusammen mit dem eigentlichen User-Prompt als Input an GPT oder ein anderes KI-System zu übergeben.

Aufbereitung der Daten:
- Splitten und Extraktion der Daten gemäß maximaler Kontext-Länge des LLM
- Generieren von Zusammenfassungen und Schlagwörtern
- Generierung der Embedding-Vektoren

Für das Erstellen der Embedding-Vektoren sind entsprechende Tools der LLMs zu nutzen. Bei GPT-3.5 besteht ein Embedding-Vektor aus 1.536 Elementen. Die maximale Kontext-Länge eines Input-Kontextes ist auf 16.385 Token beschränkt.

Auch bei den externen Daten beschreibt ein Embedding-Vektor die semantische Richtung eines Textes in einem hochdimensionalen semantischen Bedeutungsraum. Diese externen Vektoren sind in jeder Hinsicht konform zu den Vektoren im eigentlichen KI-System.

Bei der Programmierung dominiert die Sprache Python und die entsprechenden Modul-Bibliotheken. Alle eigenen Daten und generierten Vektoren können in einem eigenen Daten-Pool aus Dateien oder Datenbanken abgelegt werden.

Wird ein Prompt an das eigene KI-System dieser Bau-Art gesendet, so werden dieser Prompt und die best-passsenden Inhalte aus dem eigenen Daten-Pool aufbereitet und zusammen an das KI-System gesendet.

Durch das Anbinden der ergänzenden Daten an den Prompt sind die Kosten für diesen Input entsprechend höher. Das ist bei allem zu beachten.

Kosten für das Training

Die Kosten für 1K-Token (1.024 Token) betragen bei GPT-3.5 und GPT-4.0 im Februar 2024:

	GPT-3.5	*GPT-4.0*
Input:	0,0005€	0,01€
Output:	0,0015€	0,03€

Für das Training und Finetuning seien hier nur die Input-Kosten betrachtet, da das ein wesentlicher Kostenfaktor für das Training ist. Die weiteren oder abweichenden Kostenfaktoren für einzusetzende API-Tools oder das Tuning werden hier nicht berücksichtigt. Die nachfolgende, vereinfachte Preiskalkulation dient nur der beispielhaften Orientierung. Wie ihr oben seht, sind die Kosten bei einem Einsatz von GPT-4 im Vergleich zu GPT-3.5 bereits erheblich höher.

Durchschnittlich entspricht 1K-Token etwa 750 Wörtern. Dieses Dokument enthält etwas mehr 3.000 Wörter, also umgerechnet 4.000 Token (zirka 4K Token). Für die nachfolgenden Berechnungen werden, ganz vereinfacht, die oben genannten Kostenbeträge verwendet.

Input-Kosten für Anzahl Dokumente dieser Größenordnung in Euro:

1:	0,0005 x 4 = 0,0015
1.000:	1,50
10.000:	15,00

100.000: 150,00

1.000.000: 1.500,00

Das sieht nach einer überschaubaren Größenordnung aus. Handelt es sich jedoch um umfangreichere Dokumente, die beispielsweise die maximale 16K-Token-Länge voll beanspruchen, dann vervierfacht sich der oben genannte Preis bereits.

Müsste immer das gesamte System - also GPT-3.5 gleich mittrainiert werden, so wäre das ein ganz anderes Kosten-Volumen. Die verhältnismäßig wenigen eigenen Daten würden in diesem Fall kaum noch ins Gewicht fallen.

Kosten auf Basis der Kalkulation oben bei 175 Milliarden Token:

Das entspricht 175K Millionen Token und damit einem Kosten-Volumen von:

1.500,- x 175 = 262.500,- Euro

Bei diesem Preis käme wahrscheinlich niemand mehr auf die Idee, seine 1.000 Dokumente für einen Zugriff per KI verfügbar zu machen.

Bei alledem ist hier nicht berücksichtigt, dass ein Training in der Regel beinhaltet, dass Texte oder Bilder mehrfach einzugeben sind. Dies ist beispielsweise insbesondere bei einem nicht vortrainierten Sprachmodell der Fall. Für ein Training großer Sprachmodelle von Null an benötigen die schnellsten Computer der Welt Tage, Wochen oder gar Monate. Die Kosten hierfür gehen in die Millionen.

Kosten je Prompt

Bei einem eigenen KI-System, das auf Grundlage der oben beschriebenen RAG-Architektur mit eigenen externen Daten arbeitet, ist davon auszugehen, dass dem eigentlichen Prompt in der Regel möglichst viele eigene Daten hinzugefügt werden. Anschließend wird alles zusammen als Input an GPT übergeben.

Als Kosten für den Input ergeben sich so: 0,0005 x 16 = 0,008 Euro

Bei durchschnittlich 1K Output ergibt sich: 0,0015 x 1 = 0,0015 Euro

Zusammengerechnet betragen die Kosten je Prompt hierbei zirka: 1 Cent

Weitere API-Kosten für den Einsatz von Interpreter- und Retrieval-Tools sind hierbei nicht berücksichtigt.

Kosten für die Entwicklung von KI-Systemen

Auch bei kleinen, eigenen Anwendungen auf Basis der oben vorgestellten RAG-Architektur muss das notwendige und vielfältige Wissen entsprechender KI-Spezialisten für die Erstellung solcher Systeme vorhanden sein. Und das ist hierbei eine wesentliche Kostenposition. In Berichten von KI-Projekten ist durchaus auffallend häufig zu lesen, dass Projekte dieses Genres durch Fehleinschätzungen oder an Ressourcenmangel scheitern.

Für die Entwicklung von eigenen KI-basierten Systemen gibt es drei Entwicklungsumgebungen, die hier ganz kurz vorgestellt sind.

1) GitHub: Online-Entwicklungsumgebung. Gehört seit 2018 zu Microsoft. Hier sind viele GPT-Projekte beheimatet.
   Web: https://github.com

2) Google Colab: KI-Entwicklungsumgebung für das webbasierte Jupyter-Notebook-System. Hier wird das eigene Notebook Teil des Cloudsystems.
   Web: https://colab.research.google.com

3) Hugging Face: US-amerikanisches Unternehmen mit OpenSource Sprachmodellen, Transformer-Bibliotheken und vielem mehr für die unabhängige Entwicklung eigener KI-Systeme.
   Web: https://huggingface.co

CustomGPTs von OpenAI

Für alle, die eigene Versionen von ChatGPT ohne Programmierung und Finetuning für ihre Kunden oder generell zur Verfügung stellen möchten, gib es bei OpenAI ein Interface, um solche Anwendungen zu konfigurieren. Da OpenAI im Prinzip dafür sicherlich intern ein für diese Zwecke vorprogrammiertes und

automatisiertes Finetuning nach dem oben dargestellten Verfahren durchführt, kann kein Ergebnisniveau wie bei einer selbst programmierten und spezialisierten Anwendung erwartet werden. Zudem steht OpenAI bei diesen Anwendungen per se stärker im Konflikt mit dem Urheberrecht. Das ist ein Problem, aus dem möglicherweise zusätzlich ein Restriktionen-Dschungel entstehen wird.

Zu guter Letzt fallen auch bei den CustomGPTs die oben genannten Kosten je Prompt bei dem Einsatz von Anwendungen auf CustomGPT-Basis an.

Einsatz von OpenSource Sprachmodellen

Ein Einsatz von OpenSource Sprachmodellen als Alternative zu den üblich bekannten Modellen ist gewiss immer insbesondere dann mit in Erwägung zu ziehen, wenn es sich um spezielle KI-Aufgabenstellungen handelt, deren Ergebnis nicht auf eine exzellente Sprachausgabe zielt oder ein großes vortrainiertes Wissen voraussetzt. Diese alternativen Sprachmodelle können eben für spezielle Fälle sehr leistungsfähig sein und bieten vor allem neben einer größeren Unabhängigkeit einen weiteren Vorteil: Sie sind sehr viel kostengünstiger!

Und in dem Zusammenhang gilt sicherlich: „ Wozu mit Kanonen auf Spatzen schießen?“

Die schnellsten Rechner der Welt

Die schnellsten Rechner der Welt stehen in den USA und China. Sowohl in der Anzahl als auch in der Leistungsstärke wird sich die Kluft der Leistungsfähigkeit im Vergleich zum Rest der Welt bereits im laufenden Jahr 2024 sehr stark vergrößern.

In der Schweiz, im Kanton Tessin, soll im Frühjahr das Computersystem Alps mit 10.000 GPUs an den Start gehen. In Deutschland soll Ende 2024 ein Computer mit einer Leistung von einer Trillionen Flops pro Sekunde den Betrieb aufnehmen (mit Welt-Spitzen-Leistung). Das Jülicher System wird dann gewiss der schnellste Computer in Europa sein.

Was jedoch geschieht derweil in den USA und in China?
Um die KI-Basis für die Meta-AG (Facebook) zu schaffen, investiert der US-Konzern massiv in Rechenleistung, die zum Trainieren großer Modelle erforderlich ist. Für Meta sollen bis Ende 2024 mehr als 340.000 GPUs vom Typ Nvidias H100 für die KI arbeiten. Dieser Prozessortyp ist der bevorzugte Chip für die Entwicklung generativer KI. Elon Musk plant für 2024 ebenfalls eine 60 Milliarden US-Dollar schwere Investition in die Künstliche Intelligenz. Microsoft und Amazon werden mit ähnlichen Investitionen ihre KI weiter vorantreiben.

Die US-Regierung will mit dem „Chips Act“ ein Milliardenprogramm aufgelegt, um die für Künstliche Intelligenz so wichtige Chipindustrie in den USA zu stärken.
Mit ähnlich großen Zielen geht Peking vor: China soll 2025 die führende KI-Nation der Welt werden. Dafür investiert die Volksrepublik Milliarden in Bildung, Forschung und junge Firmen.

Die genannten Zahlen und Verlautbarungen zeigen, dass die Dynamik und Wucht, mit der die Künstliche Intelligenz die Arbeitswelt verändern wird, in den nächsten Jahren deutlich zunehmen werden.

Wissen ist Macht

Eine alte Weisheit, die insbesondere bei Geheimdiensten einen hohen Stellenwert hat. In Zukunft wird sicherlich in diesem Zusammenhang eine Technologie-Frage gleichrangig immer häufiger gestellt werden:

Wer hat die Macht über die Technologie des Wissens?

Abschließende Hinweise

Der Inhalt dieser KI-Info-PDF zeigt stichpunktartig die Quintessenz aus den vielen, teils nicht so einfach auffindbaren Informationen, die es zu den technischen Themen der Künstlichen Intelligenz gibt. Falls ihr Kritik, Hinweise oder Fragen an mich senden wollt, dann bitte schlicht und einfach an folgende Email-Adresse senden:

Günter Neumann, Email-Adresse: neumann@finde-unterkunft.de

Hinweis: Für diese KI-Info-PDF habe ich ChatGPT, Bing, Perplexity und You.com genutzt, um die hier enthaltenen Beispiel-Prompts zu testen oder um Primär-Informationen zu erhalten. In manchen Fällen sind Teile aus diesen oder weiterführenden Prompt-Antworten und Informationen übernommen worden oder haben partiell mit zu Inhalten dieser PDF inspiriert.

Und nochmals der Hinweis von ganz oben:
Das Unterkunft-Portal www.finde-unterkunft.de bietet **Direkt-Kontakt** zu mehr als **40.000 Unterkünften deutschlandweit**.

Die beiden **KI-Einsichten-PDFs - Teil-1 und Teil-2**

findet ihr auch auf den Seiten von: www.sql-schule.de/ki_pdf.php

sowie weitere Hinweise zum Thema KI und Datenbanken.

Impressum siehe: https://finde-unterkunft.de?pg=0010