Latent Space — die geheime Landkarte der KI

Kapitel online

Der verborgene Raum, in dem Bedeutung zu Koordinaten wird: Nachbarschaft, Vektor-Rechnung und Interpolation zum Anfassen — die Grundlage von LLMs und Bild-KIs.

Aktuell · Kapitel 29 von 50 · ca. 25 Min. · 250 XP

Einfach gesagt: Jedes Wort bekommt eine Adresse auf einer unsichtbaren Landkarte — du siehst, wie nah beieinander liegt, was ähnlich gemeint ist.

Kurze Frage vorweg: Woher weiß eine KI, dass „Hund“ und „Katze“ irgendwie zusammengehören — „Hund“ und „Kartoffelsalat“ aber eher nicht? Sie hat schließlich nie ein Tier gesehen und nie etwas gegessen. Sie kennt nur Zahlen.

Die Antwort ist eine der schönsten Ideen der ganzen KI: Modelle bauen sich eine **Landkarte der Bedeutungen**. Jedes Wort, jedes Bild, jeder Satz bekommt darauf eine Adresse — Koordinaten. Und was sich ähnlich ist, landet nah beieinander. Diese Landkarte heißt **Latent Space** (lateinisch *latere* = verborgen sein): der *verborgene Raum*. Verborgen deshalb, weil ihn niemand hineinprogrammiert hat. Das Modell hat ihn sich beim Training selbst gebaut.

Bis hierher hast du Modelle *gebaut*. Ab jetzt schauen wir in fertige *hinein* — und fangen an der Stelle an, an der alles zusammenläuft. Begegnet ist dir diese Landkarte übrigens schon zweimal: in „Text & LLMs“ als Word Embedding und in „Tensor-Operationen“ als Vektor-Rechnung mit König und Königin. Heute wird sie begehbar.

HerrKI-Tipp: Merk dir für dieses Kapitel einen einzigen Satz — **„Bedeutung wird zu Ort.“** Alles andere folgt daraus.

Was du hier lernst

- Was ein **Latent Space** ist — und warum „Karte“ die beste Metapher dafür ist
- Warum **Nähe = Ähnlichkeit** bedeutet (mit einer Karte zum Antippen)
- Wie man mit Bedeutung **rechnet**: König − Mann + Frau ≈ Königin
- Was passiert, wenn du **zwischen** zwei Punkten spazieren gehst
- Warum **ChatGPT und Bild-KIs** ohne diesen Raum gar nicht funktionieren würden

Für den Unterricht

Einstieg (5 Min): Kapitel gemeinsam lesen, Alltagsbeispiel aufgreifen.

Erkunden am Gerät (15 Min): die interaktiven Module auf herrki.de/310_latent_space

ausprobieren.

Sichern mit Quiz (5 Min): „Kurzer Check" besprechen, „Das nimmst du mit" festhalten.

Runter zum Tensor: Von Wörtern zu Koordinaten

In den Kapiteln über Text und LLMs hast du schon gesehen: Bevor ein Modell mit Sprache arbeiten kann, wird jedes Token in eine lange Zahlenreihe übersetzt — ein **Embedding**. Und diese Zahlenreihe ist nichts Neues: Es ist exakt der **Vektor aus Kapitel 010** — ein 1D-Tensor mit der Form [768]. Zum Beispiel so:

„Hund“

↓

`[0.24, −0.81, 0.05, 0.63, …]`
[768]

→

Ein Punkt

↓

in einem Raum mit
768 Richtungen
1 Adresse

Und jetzt der Gedankensprung, um den es heute geht: **768 Zahlen sind nichts anderes als 768 Koordinaten.** Bei zwei Zahlen wäre es ein Punkt auf einem Blatt Papier (rechts/links, oben/unten). Bei drei Zahlen ein Punkt im Zimmer. Bei 768 Zahlen … ein Punkt in einem Raum, den sich kein Mensch vorstellen kann — den man aber genauso behandeln darf: Man kann Abstände messen, Richtungen laufen, Nachbarn suchen.

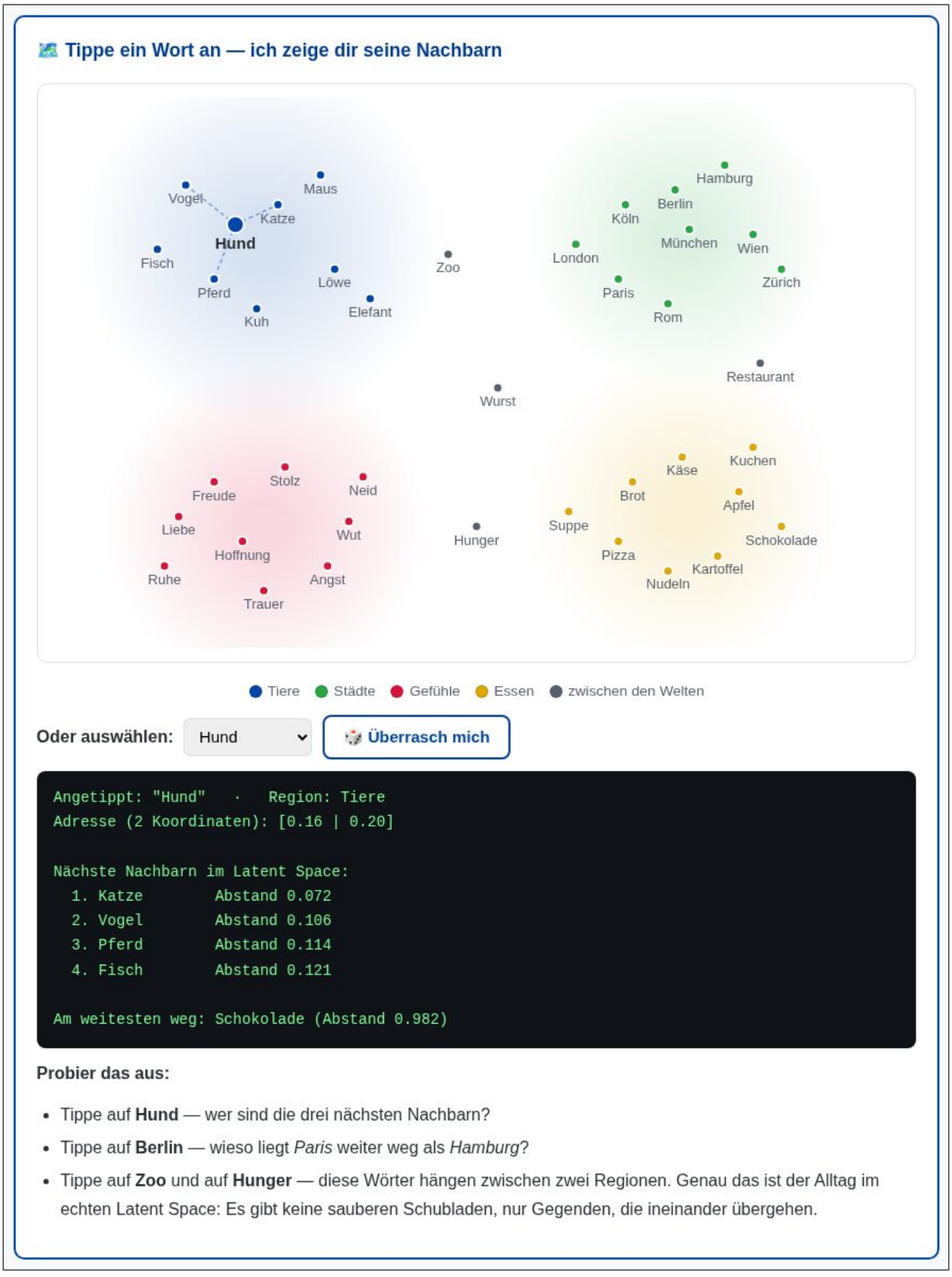
⚠ **Ehrlich bleiben: Die Karte hier ist geschummelt.** Ich zeige dir gleich eine Landkarte mit **2** Dimensionen, echte Modelle nutzen 384, 768, 1536 oder 12 288. Das ist derselbe Kompromiss wie bei einer Weltkarte: Die Erde ist eine Kugel, das Papier ist flach — irgendwo muss gestaucht werden (Grönland ist auf den meisten Karten viel zu groß). Genauso platten Verfahren wie *t-SNE* oder *UMAP* hunderte Dimensionen auf zwei herunter, damit wir überhaupt etwas sehen. Die Nachbarschaften stimmen dabei ziemlich gut — die Abstände nur ungefähr.

Die Landkarte der Bedeutungen

Hier ist deine Karte. 40 deutsche Wörter, jedes an seiner Adresse. Kein Mensch hat die Wörter sortiert — stell dir vor, ein Modell hätte sie beim Lesen von Millionen Sätzen genau dorthin gelegt, weil sie in ähnlichen Zusammenhängen vorkommen.

Am Gerät ausprobieren: herrki.de/310_latent_space#abschnitt-3

Direkt zum Modul



Interaktives Modul – am Gerät ausprobieren

Ähnlichkeit ist einfach nur Abstand

Das Schöne daran: Sobald Bedeutung ein Ort ist, wird „ähnlich“ zu einer stinknormalen Rechenaufgabe. Der Abstand zwischen zwei Punkten — Satz des Pythagoras, den kennst du aus der Schule:

```
// Abstand zwischen zwei Punkten – mehr steckt hinter "Ähnlichkeit" nicht
function abstand(a, b) {
  let summe = 0;
  for (let i = 0; i < a.length; i++) {
    summe += (a[i] - b[i]) ** 2; // pro Dimension: Differenz zum Quadrat
  }
  return Math.sqrt(summe); // klein = ähnlich, groß = verschieden
}
```

Und falls dir das bekannt vorkommt: Genau diese Rechnung ist eine **Tensor-Operation aus Kapitel 060** — elementweise subtrahieren, quadrieren, aufsummieren, Wurzel ziehen. In echten Systemen läuft sie nicht über 2, sondern über 768 Zahlen und über Millionen Punkte gleichzeitig, als eine einzige Matrix-Multiplikation auf der Grafikkarte. Deshalb ist „Verstehen“ für einen Computer am Ende genau das: eine Tensor-Rechnung.

Falls du das nachschlagen willst: Genau diese vier Handgriffe hast du in „Tensor-Operationen“ schon selbst durchgerechnet — hier bekommen sie nur einen neuen Namen: **Ähnlichkeit**.

Diese fünf Zeilen laufen millionenfach pro Sekunde in jeder semantischen Suche, in jedem Empfehlungssystem, in jedem RAG-System. (In der Praxis nimmt man meist die *Kosinus-Ähnlichkeit*, die nur den **Winkel** zwischen zwei Vektoren misst und die Länge ignoriert — das Prinzip bleibt dasselbe.)

Rechnen mit Bedeutung

Jetzt wird es richtig verrückt. Wenn Bedeutung ein Ort ist, dann sind **Richtungen** auch etwas wert. Und tatsächlich: In echten Wort-Embeddings zeigt eine ganz bestimmte Richtung immer „männlich → weiblich“. Eine andere zeigt „Einzahl → Mehrzahl“. Eine dritte „Land → Hauptstadt“.

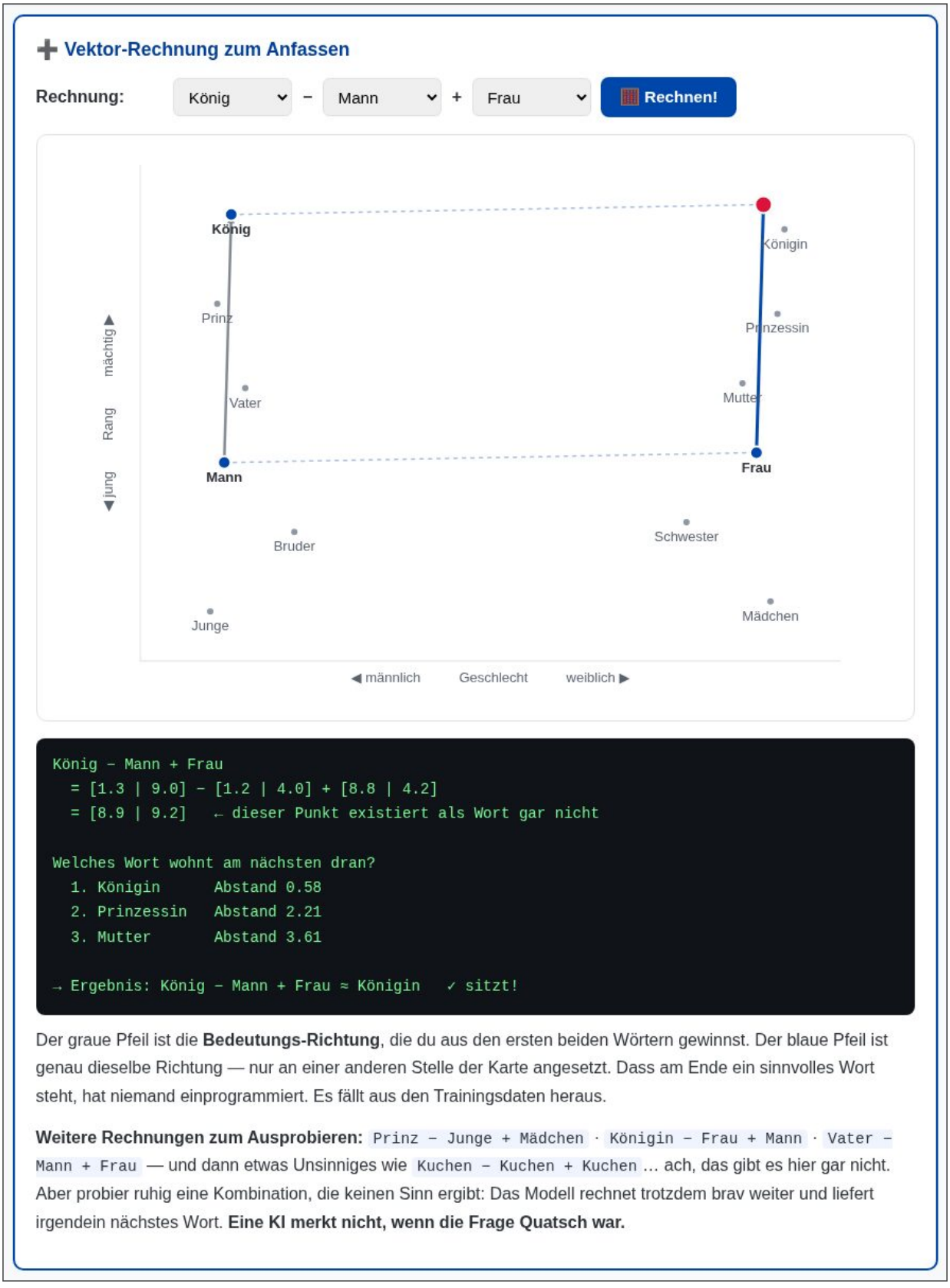
Wenn das stimmt, muss man mit Wörtern rechnen können. Und das kann man — es ist das berühmteste Kunststück der KI-Geschichte:

König − Mann + Frau ≈ Königin

Übersetzt: „Nimm König. Zieh alles ab, was ihn zum Mann macht. Addiere alles, was eine Frau ausmacht. Wo landest du?“ Auf der kleinen Karte unten haben die Achsen sogar eine erkennbare Bedeutung — links/rechts ist das Geschlecht, oben/unten der Rang. Probier es aus:

Am Gerät ausprobieren: herrki.de/310_latent_space#abschnitt-4

Direkt zum Modul



Interaktives Modul – am Gerät ausprobieren

⚠ **Und jetzt die unangenehme Seite.** Dieselbe Rechnung ergibt in echten, auf Internet-Texten trainierten Modellen auch Ergebnisse wie *„Arzt − Mann + Frau ≈ Krankenschwester“* oder *„Programmierer − Mann + Frau ≈ Hausfrau“*. Das ist kein Bug — der Latent Space ist ein ehrlicher Spiegel der Texte, aus denen er gebaut wurde. Wenn in Millionen Sätzen Ärzte männlich und Pflegekräfte weiblich sind, wird genau das zur Geometrie. Deshalb ist „Was steckt in den Trainingsdaten?“ keine Nebenfrage, sondern die Hauptfrage.

Spazieren gehen im Latent Space

Eine Karte kann man nicht nur ablesen — man kann auf ihr **laufen**. Und hier kommt der Punkt, an dem viele zum ersten Mal wirklich staunen: Zwischen zwei Wörtern liegt kein Loch. Da liegt Bedeutung, die es gar nicht als eigenes Wort gibt.

Am Gerät ausprobieren: herrki.de/310_latent_space#abschnitt-5

Direkt zum Modul

🚶 Lauf über die Karte

Start: Hund **Ziel:** Pizza

Weg: 0 %

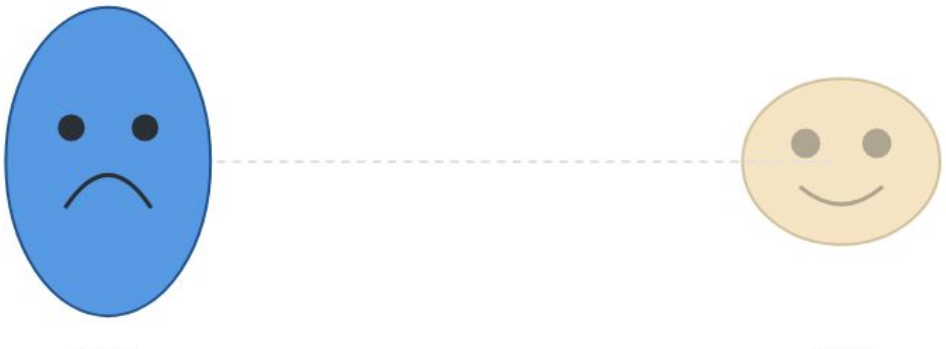
```
Wähle Start und Ziel und schieb den Regler – der rote Punkt wandert oben über die Karte.
```

Schieb den Regler langsam von 0 % auf 100 % und schau nach oben auf die Karte — der rote Wanderpunkt läuft mit. Unterwegs kommst du an Regionen vorbei, für die es kein passendes Wort gibt. **Genau da wohnt das, was KI kreativ wirken lässt.**

Dasselbe mit Bildern: der Morph

Bei Bildern wird der Spaziergang sichtbar. Das Gesicht unten ist kein Foto und kein Video — es wird aus **vier Zahlen** gezeichnet: Kopfform, Mundkurve, Augengröße, Farbton. Diese vier Zahlen sind sein Latent-Vektor. Links steht ein Punkt im Raum, rechts ein anderer. Der Regler läuft dazwischen:

🎚️ Von A nach B — durch den Raum, nicht durch Überblenden



Punkt A Punkt B

Position: 0 %

`Kopfform: 0.18` `Mundkurve: -0.85` `Augen: 0.30` `Farbton: 212`

> **Der Unterschied zum Überblenden:** Würdest du zwei Fotos ineinander überblenden, sähest du in der Mitte *zwei durchsichtige Gesichter übereinander* — Matsch. Hier siehst du in der Mitte ein **echtes drittes Gesicht**, das es vorher nicht gab. Weil nicht die Bilder gemischt werden, sondern ihre **Bedeutungen**. Das ist der ganze Trick hinter Bild-KI.

Interaktives Modul – am Gerät ausprobieren

Dasselbe mit Bildern: der Morph

Bei Bildern wird der Spaziergang sichtbar. Das Gesicht unten ist kein Foto und kein Video — es wird aus **vier Zahlen** gezeichnet: Kopfform, Mundkurve, Augengröße, Farbton. Diese vier Zahlen sind sein Latent-Vektor. Links steht ein Punkt im Raum, rechts ein anderer. Der Regler läuft dazwischen:

Warum ohne diesen Raum nichts mehr läuft

Bild-KIs (Diffusion) rechnen fast nur im Latent Space

Stable Diffusion heißt mit vollem Namen *Latent Diffusion* — und das ist kein Zufall. Rechne mal mit: Ein Bild mit 512 × 512 Pixeln in Farbe hat 512 · 512 · 3 = **786 432 Zahlen**. Für jeden einzelnen Entrauschungs-Schritt (und davon gibt es 20 bis 50) wäre das brutal teuer.

Pixel-Bild
[512, 512, 3]
786 432 Zahlen

→ Encoder →

Latent
[64, 64, 4]
16 384 Zahlen

→ Decoder →

Fertiges Bild
[512, 512, 3]
786 432 Zahlen

Ein **Encoder** presst das Bild auf 16 384 Zahlen zusammen — rund **48-mal weniger**. Die ganze Arbeit („aus Rauschen wird Struktur“) passiert in dieser kompakten Form, und erst ganz am Schluss malt ein **Decoder** daraus wieder echte Pixel. Deshalb läuft Bild-KI überhaupt auf normalen Grafikkarten. Und deshalb funktionieren die berühmten Morph-Videos: Man läuft im Latent Space von einem Punkt zum nächsten und lässt jeden Zwischenschritt ausmalen.

LLMs sind eine Reise durch den Latent Space

Was in Kapitel 050 als Pipeline beschrieben war, kannst du jetzt geometrisch sehen: Jedes Token startet als Punkt (sein Embedding). Jede Transformer-Schicht **verschiebt** diesen Punkt ein Stück — abhängig davon, was drumherum steht. Das Wort „Bank“ startet an einer neutralen Adresse und wandert dann Richtung „Geld“ oder Richtung „Park“, je nach Satz. Am Ende schaut das Modell nach: Welches Wort wohnt dem Punkt am nächsten, an dem ich gelandet bin?

Semantische Suche und RAG

Der praktischste Nutzen im Alltag: Klassische Suche vergleicht Buchstaben — wer „Auto“

tippt, findet kein Dokument, in dem nur „Fahrzeug“ steht. Semantische Suche verwandelt die Frage *und* jedes Dokument in Punkte und sucht die nächsten Nachbarn. Deshalb findet ein gut gebauter Firmen-Chatbot die richtige Handbuchseite, obwohl kein einziges Wort wörtlich übereinstimmt. Genau das ist **RAG** (Retrieval Augmented Generation) — und es ist im Kern das, was du oben angetippt hast, nur mit ein paar Millionen Punkten mehr.

Weil das im Alltag so nützlich ist, hat es ein eigenes Kapitel dazu bekommen: „Embeddings in der Praxis: Semantische Suche & RAG“ — dort baust du die Suche Schritt für Schritt selbst.

Zum Weiterlesen

- **Wo das Rechnen mit Bedeutung herkommt:** Tomas Mikolov u. a., *„Efficient Estimation of Word Representations in Vector Space“* (word2vec), 2013 — arxiv.org/abs/1301.3781 — hier wurde „König − Mann + Frau“ zum ersten Mal messbar
- **Zum Anschauen:** Christopher Olah, *„Visualizing MNIST“* — colah.github.io (colah.github.io/posts/2014-10-Visualizing-MNIST) — zeigt Bild für Bild, wie aus vielen Dimensionen eine begehbare Karte wird
- **Zum Selbst-Herumlaufen:** TensorFlow Embedding Projector — projector.tensorflow.org — dieselbe Nachbarschafts-Suche wie oben, nur mit echten Modellen und Zehntausenden Wörtern
- **Der älteste Faden:** Geoffrey Hinton & Ruslan Salakhutdinov, *„Reducing the Dimensionality of Data with Neural Networks“* (Science, 2006) — PDF (cs.toronto.edu) (cs.toronto.edu/~hinton/absps/science.pdf) — warum ein Netz überhaupt lernen kann, Daten auf wenige Richtungen einzudampfen

Kurzer Check – mit Lösung

Lösungen für Lehrkräfte – für Lernende die Online-Version nutzen.

Frage 1: Zwei Wörter liegen im Latent Space dicht beieinander. Was heißt das?

- Sie haben gleich viele Buchstaben.
- **✔ Sie werden in ähnlichen Zusammenhängen benutzt — ihre Bedeutung ist ähnlich.**
- Sie stehen im Wörterbuch direkt nebeneinander.

Genau. Der Ort entsteht allein daraus, in welchen Umgebungen ein Wort im Training auftaucht. Schreibweise und Alphabet spielen keine Rolle — deshalb liegen „Arzt“ und „Ärztin“ nah beieinander, „Arzt“ und „Harz“ aber nicht.

Frage 2: Warum arbeitet Stable Diffusion im Latent Space statt direkt auf den Pixeln?

- Weil sich Pixel nicht entrauschen lassen.
- Weil im Latent Space die fertigen Bilder gespeichert sind und nur herausgesucht werden.
- **✔ Weil dort rund 48-mal weniger Zahlen zu verarbeiten sind — das spart enorm Rechenzeit.**

Richtig: [64, 64, 4] statt [512, 512, 3]. Nichts wird „nachgeschlagen“ — der Decoder malt das Bild jedes Mal neu aus den komprimierten Zahlen.

Frage 3: Die 2D-Karte in diesem Kapitel ist …

- **✔ … eine stark heruntergerechnete Projektion; echte Räume haben hunderte bis tausende Dimensionen.**
- … exakt die Form, in der auch ChatGPT seine Wörter ablegt.
- … ein Zufallsbild ohne Bezug zur echten KI.

Wie eine Weltkarte: nützlich, aber verzerrt. Nachbarschaften bleiben meist erhalten, Abstände und Flächen nicht — genau deshalb heißt es „Projektion“ und nicht „Abbild“.

Das nimmst du mit

- **Bedeutung wird zu Ort.** Ein Embedding ist eine Adresse in einem Raum mit sehr vielen Richtungen.
- **Nähe = Ähnlichkeit.** „Verstehen“ sieht von innen aus wie Abstandsmessen.
- **Richtungen tragen Bedeutung.** Deshalb kann man mit Wörtern rechnen — inklusive der geerbten Vorurteile.
- **Zwischenräume sind nicht leer.** Interpolation erzeugt Neues, statt Altes zu überblenden.
- **Niemand hat die Karte gezeichnet.** Sie ist beim Training von selbst entstanden — das ist der eigentliche Clou.

Wie es weitergeht

Du weißt jetzt, *wo* die KI ihre Bedeutungen aufbewahrt. Als Nächstes wird spannend, *wie* die Punkte in Bewegung geraten: Der **Attention-Mechanismus** entscheidet, welches Wort welches andere verschiebt. Danach schauen wir uns an, wie aus purem Rauschen ein Bild wird — und warum ein Modell, das nur Nachbarn sucht, manchmal überzeugend Unsinn erzählt.

Übrigens: Du hast heute nichts trainiert und trotzdem verstanden, wie moderne KI innen aussieht. Das ist keine schlechte Ausbeute für 20 Minuten.

So zitierst du dieses Kapitel

HerrKI: Latent Space — die geheime Landkarte der KI. herrki.de/310_latent_space, ein privates Hobbyprojekt von Terragon.de, abgerufen am 09.09.2026.