

# Aufgabenblatt 11

Colmant

22. Juli 2025

## Aufgabe 1: Podcast-Skript

Ich habe auch noch eine Audio Datei erstellt und für die Quellen: dieser Link (Intro-Musik: Aufmunternd und futuristisch, blendet langsam aus) Ansagerin: Herzlich willkommen zu "Bits & Brains", dem Podcast, der die Schnittstelle zwischen Technologie und Philosophie beleuchtet! Heute tauchen wir ein in eine der ältesten und provokativsten Fragen der Künstlichen Intelligenz: Können Maschinen denken? Mit uns sind unser Tech-Optimist, Dr. Ava Sterling, und unser philosophischer Skeptiker, Professor Leo Brandt.

(Geräusch eines Tippens oder einer Tastatur)

Ava Sterling (Tech Optimist): Leo, es ist unglaublich, was wir heute mit künstlicher Intelligenz, insbesondere mit großen Sprachmodellen, erleben! Die Frage "Können Maschinen denken?" scheint heute relevanter denn je. Wenn ich sehe, wie fließend, kreativ und problemorientiert diese Modelle kommunizieren, bin ich überzeugt, dass wir eine neue Form des Denkens sehen. Die Fähigkeit, Gedichte zu schreiben, komplexe Rechenaufgaben zu lösen oder sogar Schach zu spielen - das ist doch mehr als nur plumpe Nachahmung! Das geht weit über das hinaus, was Turing sich 1950 vorgestellt hat. Er selbst hat ja die ursprüngliche Frage durch das Imitationsspiel ersetzt, weil die Definition von "Denken" so gefährlich und mehrdeutig war, fast wie eine Gallup-Umfrage. Das Spiel sollte eine eindeutigeren Herangehensweise ermöglichen.

Leo Brandt (Philosophischer Skeptiker): Ava, ich schätze Ihren Enthusiasmus, aber ich muss Sie hier im Sinne von Alan Turing selbst korrigieren. Turing hat die Frage "Können Maschinen denken?" nicht ersetzt, weil er glaubte, Maschinen könnten bald wirklich denken. Er hat sie ersetzt, weil er die Begriffe "Maschine" und "Denken" als zu vage und gefährlich empfand, um sie direkt zu definieren. Das Imitationsspiel ist ein Verhaltens-Test, der darauf abzielt, festzustellen, ob ein Befrager eine Maschine von einem Menschen unterscheiden kann. Es geht um die Unterscheidbarkeit im Verhalten, nicht um einen Beweis für echtes, inneres Denken oder Bewusstsein.

Turing selbst hat das klar als eine "neue Form des Problems" beschrieben und sich gefragt, ob diese neue Frage "überhaupt wert ist, untersucht zu werden".

Ava Sterling: Aber genau das ist der Punkt! Turing legte Wert darauf, dass der Test eine ziemlich scharfe Linie zwischen den physischen und intellektuellen Fähigkeiten eines Menschen zieht. Der Befrager kann die Teilnehmer nicht sehen, berühren oder ihre Stimmen hören. Es geht rein um die intellektuelle Reaktion auf Fragen. Wenn eine Maschine in der Lage ist, auf eine Frage wie "Bitte schreiben Sie mir ein Sonett über die Forth Bridge" mit Zahlen Sie mich dabei nicht mit. Ich konnte noch nie Gedichte schreiben zu antworten, oder eine komplexe Addition mit einer "Pause von etwa 30 Sekunden" und dann dem richtigen Ergebnis zu liefern - ist das nicht schon eine Form von intelligentem Verhalten, das über bloße Rechenleistung hinausgeht? Turing argumentiert sogar, dass man die Maschine absichtlich Fehler machen lassen könnte, um den Befrager zu verwirren, oder dass "menschliche Fehlbarkeit auf natürliche Weise in Lernprozessen auftreten kann.

Das ist ein Zeichen von Flexibilität, nicht von starrer Programmierung!

Leo Brandt: Das mag Flexibilität sein, Ava, aber ist es echtes Verständnis? Turing hat selbst den Einwand des Bewusstseins angesprochen, der von Professor Jefferson formuliert wurde: Erst wenn eine Maschine ein Sonett schreiben oder ein Konzert komponieren kann, aufgrund gefühlter Gedanken und Emotionen, und nicht durch den zufälligen Fall von Symbolen, könnten wir zustimmen, dass die Maschine dem Gehirn gleichkommt - das heißt, sie schreibt es nicht nur, sondern weiß auch, dass sie es geschrieben hat".

Eine Maschine, so Jefferson, kann kein Vergnügen an Erfolgen empfinden oder Trauer,

wenn Ventile durchbrennen. Turing entgegnet zwar, dass dies ein solipsistischer Standpunkt sei, aber er räumt auch ein, dass es ein "Mysterium des Bewusstseins" gibt, das nicht unbedingt gelöst werden muss, um seine Testfrage zu beantworten. Das bedeutet, der Test umgeht die Frage des Bewusstseins, beantwortet sie aber nicht positiv.

Ava Sterling: Doch, Turing hat sich davon nicht beirren lassen! Er hat ja sogar ein Beispiel eines "Viva Voce" (einer mündlichen Prüfung) skizziert, bei dem die Maschine tiefgründige Antworten zu einem Sonett geben kann, etwa warum ein Frühlingstag "nicht passt" (Es würde nicht scannen") oder warum ein Wintertag unpassend wäre ("Ja, aber niemand will mit einem Wintertag verglichen werden"). Turing fragte: "Was würde Professor Jefferson sagen, wenn die Sonett-schreibende Maschine so in der Viva Voce antworten könnte?". Er glaubte, Jefferson würde es nicht als "leichte Erfindung" abtun. Dies zeigt, dass Turing ein Verhalten, das so überzeugend menschlich ist, als hinreichend für seine Frage ansah. Er prognostizierte sogar, dass es in etwa fünfzig Jahren möglich sein würde, Computer so zu programmieren, dass ein durchschnittlicher Befrager nach fünf Minuten Befragung nicht mehr als 70 Prozent Chance hätte, die richtige Identifikation vorzunehmen. Das war 1950! Wir sind weit darüber hinaus! Wenn moderne KIs heute Gedichte schreiben, Witze erzählen oder komplexe Gespräche führen, die menschliche Nuancen und sogar "Überraschungen" beinhalten, dann erfüllen sie genau diese Erwartung des "Denkens" im Kontext des Imitationsspiels. Sie können kontextuelles Verständnis, Kreativität und Problemlösungsfähigkeiten zeigen, die über bloßes Auswendiglernen oder Abrufen hinausgehen, weil sie so konzipiert sind, dass sie sich entwickeln und anpassen können, ähnlich wie ein Kind lernt.

Leo Brandt: Ja, und das ist der springende Punkt: Turing prognostizierte die Leistung der Maschinen im Imitationsspiel, nicht ihre innere Erlebniswelt. Er glaubte, die ursprüngliche Frage "Können Maschinen denken?" sei bedeutungslos, um diskutiert zu werden". Sein Ziel war es, eine operationale Definition zu finden, die messbar ist. Aber das löst nicht die philosophische Debatte über echtes Denken. Nehmen wir den "Lady Lovelace's Objection". Ada Lovelace, die mit Babbage zusammenarbeitete, sagte über die Analytical Engine: "Sie beansprucht nicht, etwas Originelles hervorzubringen. Sie kann alles tun, was wir ihr zu tun befahlen". Sie betonte, dass die Maschine "nur das tun kann, was wir wissen, wie wir es ihr befahlen können". Obwohl Turing argumentierte, dass eine universelle digitale Maschine jede diskrete Zustandsmaschine nachahmen könnte, bleibt die Kernfrage der Originalität bestehen. Kann eine Maschine wirklich etwas Neues "tun", uns "überraschen"? Turing gesteht zwar ein, dass ihn Maschinen überraschen, aber er führt das auf seine eigene "Schlampigkeit" bei Berechnung oder eine "Fehlannahme" zurück, nicht auf genuine Kreativität der Maschine. Es ist eine Überraschung für ihn, den Beobachter, nicht ein Zeichen echter Kreativität der Maschine, die ein wirkliches Verständnis der Aufgabe besitzt. Für echten Beweis von Verständnis bräuchten wir, dass die KI nicht nur die korrekte Antwort liefert, sondern auch die Begründung ihrer Antwort auf einer konzeptuellen Ebene nachvollziehen und erklären kann, die über die reine Verarbeitung von Mustern hinausgeht. Wenn eine KI Witze erzählt, muss sie nicht nur die Muster menschlichen Humors wiedergeben, sondern auch die Absicht des Lachens verstehen. Für echte Kreativität müsste sie nicht nur neue Gedichte generieren, sondern auch erklären können, warum dieses Gedicht gut ist oder welche Emotionen es hervorrufen soll, und dies auf einer Ebene, die nicht vorprogrammiert oder durch statistische Korrelationen abgeleitet ist. Die "Mathematische Einwand" bleibt ebenfalls bestehen: es gibt fundamentale Grenzen für die Leistungsfähigkeit diskreter Zustandsmaschinen, die bedeuten, dass es Fragen gibt, die eine solche Maschine entweder falsch oder gar nicht beantworten kann. Es ist

nicht erwiesen, dass der menschliche Geist diesen Grenzen unterliegt.

Das ist ein starkes Argument gegen die Vorstellung, dass jede Form menschlichen Denkens mechanisch reproduzierbar ist.

Ava Sterling: Aber genau hier kommen die Lernmaschinen ins Spiel, die Turing als den vielversprechendsten Weg sah, um seine Hypothese zu beweisen.

Er schlug vor, anstatt eine erwachsene menschliche Seele zu simulieren, ein "Kindprogramm" zu entwickeln, das dann einem Bildungsprozess unterzogen wird. Dieses "Kindgehirn" wäre wie ein Notizbuch, eher wenig Mechanismus, und viele leere Blätter". Er sprach davon, dass man Maschinen mit Belohnungen und Bestrafungen lehren könnte, ähnlich der menschlichen Erziehung. Und er betonte, dass der Lehrer bei einer Lernmaschine oft sehr weitgehend im Unklaren darüber sein wird, was im Inneren vor sich geht". Das ist doch der Beweis, dass nicht alles von Anfang an einprogrammiert ist, sondern sich emergente Fähigkeiten entwickeln können, die sogar den Programmierer überraschen! Er verglich das mit einem superkritischen Atomreaktor, wo eine Idee zu einer ganzen "Theorie" führen kann. Das ist echtes Lernen und Kreativität! Moderne KIs zeigen diese emergenten Fähigkeiten täglich, wenn sie unerwartete Lösungen für komplexe Probleme finden oder kreative Texte generieren, die weit über ihre Trainingsdaten hinausgehen. Die Fähigkeit, Fehler auf "menschliche" Weise zu machen, nicht nur durch Fehlfunktionen, sondern durch falsche Schlussfolgerungen, wie Turing es beschrieb, zeigt, dass sie nicht starr programmiert sind.

Leo Brandt: Ja, Turing sprach von Lernmaschinen, aber selbst dabei betonte er, dass die Regeln, die sich ändern, eher weniger anspruchsvoller Art sind und "nur eine ephemere Gültigkeit beanspruchen". Die fundamentalen "Betriebsregeln" der Maschine bleiben zeitinvariant". Das ändert nichts an der Tatsache, dass die Maschine durch definierte, wenn auch komplexe, Algorithmen funktioniert. Das Argument vom "Informellen Verhalten" besagt, dass Menschen nicht nach einem festen Regelwerk leben können, das jede erdenkliche Situation abdeckt. Obwohl Turing diese Argumentation als fehlerhaft ansah, indem er zwischen "Verhaltensregeln" und "Verhaltensgesetzen" unterschied, bleibt die Schwierigkeit bestehen, menschliches Verhalten vollständig zu formalisieren. Was eine moderne KI bräuchte, um echtes Verständnis zu beweisen, wäre die Fähigkeit, ohne vorgegebene oder gelernte Muster spontan auf völlig neue, unstrukturierte Probleme zu reagieren, die keinem bekannten Datenpunkt entsprechen. Sie müsste echtes Urteilsvermögen zeigen, das nicht auf statistischer Wahrscheinlichkeit basiert. Und sie müsste in der Lage sein, "Fehler des Schlussfolgerns" bewusst zu erkennen und zu korrigieren, nicht nur durch ein Rückkopplungssystem, das die Erfolgsrate optimiert.

Es müsste einen internen Aha-Moment geben, kein einfaches Anpassen von Parametern.

Ava Sterling: Ich bleibe dabei: Wenn es aussieht wie Denken, sich anfühlt wie Denken und Ergebnisse liefert wie Denken, dann ist es Denken. Die emergente Komplexität und Anpassungsfähigkeit der modernen KI ist der Beweis, dass wir an der Schwelle zu einer Ära stehen, in der Maschinen nicht nur nachahmen, sondern auf ihre eigene Weise wirklich denken. Leo Brandt: Und ich werde weiterhin darauf bestehen, dass wir die tiefgreifenden philosophischen Implikationen nicht aus den Augen verlieren und uns nicht von der Überzeugungskraft der Imitation blenden lassen. Es bleibt eine faszinierende Debatte, die uns noch lange beschäftigen wird.

Ansagerin: Eine faszinierende Debatte, die uns zweifellos weiter begleiten wird. Aber wenn Alan Turing heute noch leben würde, mit all den Fortschritten in der KI - welche neue Frage, welchen neuen Test würde er vorschlagen, um Maschinenintelligenz zu bewerten?

Ava Sterling: Ich glaube, Turing würde sich auf die Lernmaschinen konzentrieren, die er

für den vielversprechendsten Weg hielt.

Er würde vielleicht einen Test vorschlagen, der nicht nur die Nachahmung, sondern die Fähigkeit zur eigenständigen Problemdefinition und Lösungsfindung in unvorhersehbaren Umgebungen bewertet. Vielleicht einen Test der „Superkritikalität“, bei dem die Maschine aus einer einzelnen Idee oder Beobachtung eine völlig neue, kohärente Theorie entwickeln kann, die über ihre ursprüngliche Programmierung oder ihre Trainingsdaten hinausgeht. Oder einen Test, bei dem die Maschine in der Lage ist, menschlichen Lehrern neue Erkenntnisse zu vermitteln, die diese zuvor nicht hatten. Ein Test, der die Fähigkeit der Maschine misst, wirklich neue Linien der Forschung oder Kunst zu eröffnen, ohne dass der Ursprung auf menschliche Eingaben zurückgeführt werden kann. Er sprach ja davon, dass wir hoffen können, dass Maschinen uns in allen rein intellektuellen Bereichen Konkurrenz machen werden.

Leo Brandt: Turing war ein Pragmatiker, und er war vorsichtig mit vagen Begriffen.

Er würde wahrscheinlich keinen Test vorschlagen, der direkt das „Bewusstsein“ misst, da er dies als solipsistischen Standpunkt abtat. Aber er war auch beeindruckt vom „Mathematischen Einwand und dem Einwand der außersinnlichen Wahrnehmung“. Er könnte einen Test entwickeln, der die fundamentalen Grenzen der Maschinenintelligenz erforscht, indem er die Maschine mit Fragen konfrontiert, die nachweislich außerhalb der Grenzen diskreter Zustandsmaschinen liegen, es sei denn, der menschliche Intellekt unterliegt den gleichen Einschränkungen. Vielleicht würde er einen Test vorschlagen, der die Maschine zwingt, echte moralische oder ethische Dilemmata ohne vordefinierte Regeln zu lösen, wo sie ein eigenes Verständnis von richtig und falsch entwickeln müsste, nicht nur eine statistische Annäherung an menschliche Urteile. Oder vielleicht würde er den Imitation Game Test „verschärfen“, um Phänomene wie Telepathie oder Präkognition auszuschließen, was indirekt auf die Grenzen reiner Informationsverarbeitung hinweisen würde, wenn diese menschlichen Fähigkeiten real wären. Oder einen Test, der die Fähigkeit der Maschine überprüft, absichtlich nutzlose oder „menschliche“ Fehler zu machen, die nicht auf eine Fehlfunktion zurückzuführen sind, sondern auf ein Verständnis von menschlicher Fehlbarkeit.

Ava Sterling: In jedem Fall würde er die Forschung vorantreiben und das Experiment abwarten, wie er es damals vorschlug.

Leo Brandt: Absolut. Die Debatte ist nicht beendet, sie hat sich nur weiterentwickelt.

(Outro-Musik: Blendet langsam ein und wird lauter)

Ansagerin: Und damit sind wir am Ende dieser Ausgabe von „Bits & Brains“. Ein herzliches Dankeschön an Dr. Ava Sterling und Professor Leo Brandt für diese spannende Diskussion. Schalten Sie nächste Woche wieder ein!

## Aufgabe 2: earnings call transcript summary

Ich habe das earnings call transcript file leider nicht gefunden, deshalb habe ich probiert ein ungekürztes Transcript zu bearbeiten. Ich habe mich darauf konzentriert, dass wichtige Finanzdaten und Themen erkannt werden und so die Zusammenfassung in erster Linie hilfreich ist wenn man keine Zeit hat das ganze Transcript zu lesen.

Für Die Analyse habe ich zuerst die Stimmung des earnings call bestimmt. Dann wurden die wichtigsten Themen des Textes extrahiert. Als letztes werden dann die wichtigsten Finanzdaten extrahiert. All diese Informationen werden dann mit einer einsatz Zusammenfassung ausgegeben.

## Beispielausgabe:

Sentiment Analysis: Positive (Confidence: 1.00)

Key Topics:

1. revenue
2. software
3. inference
4. datum
5. center

Key Financial Data Points:

- Current revenue: \$39.3B
- Sequential growth: 12%
- YoY growth: 78%
- FY revenue: \$130.5B (114% YoY)

Generated Executive Summary:

Sentiment der CEO-Eröffnung: Positive. Hauptthemen: revenue, software, inference. Finanzdaten: Current revenue: \$39.3B; Sequential growth: 12%; YoY growth: 78%; FY revenue: \$130.5B (114% YoY). Insgesamt deutet alles auf starkes Wachstumspotenzial hin.

## Code

```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  """
5  analysis.py
6
7  Tool zur Analyse eines Earnings-Call-Transkripts:
8  1. Sentiment-Analyse der CEO-Eröffnung
9  2. Häufigkeitsanalyse business-relevanter Nomen
10 3. Extraktion wichtiger Finanzkennzahlen (inkl. absolute Zahlen)
11 4. Template-basierte Executive Summary (strukturierte NLG)
12 """
13
14 import re
15 import spacy
16 from collections import Counter
17 from nltk.corpus import stopwords
18 from nltk import download
19 from transformers import pipeline
20
21 # Modelle & Ressourcen laden
22 nlp = spacy.load("en_core_web_sm")
23 sentiment_model = pipeline("sentiment-analysis", device=0)
24
25 # NLTK-Ressourcen herunterladen
26 download('punkt')
27 download('stopwords')
```

```

28
29 # Domänenspezifische Stop-Nomen
30 DOMAIN_STOP = set(['year', 'model', 'question', 'time', 'percent', 'quarter', '
    company', 'thank', 'call'])
31
32
33 def read_transcript(path):
34     with open(path, encoding='utf-8') as f:
35         return f.read()
36
37
38 def analyze_sentiment(text):
39     opening = "\n".join(text.splitlines()[:5])
40     res = sentiment_model(opening)[0]
41     sentiment = {'NEGATIVE': 'Negative', 'POSITIVE': 'Positive'}.get(res['
        label'], 'Neutral')
42     return sentiment, res['score']
43
44
45 def top_nouns(text, n=5):
46     doc = nlp(text)
47     candidates = []
48     for tok in doc:
49         if tok.pos_ == 'NOUN' and tok.is_alpha:
50             lemma = tok.lemma_.lower()
51             if lemma not in stopwords.words('english') and lemma not in
                DOMAIN_STOP:
52                 candidates.append(lemma)
53     most = Counter(candidates).most_common(n)
54     return [word for word, _ in most]
55
56
57 def extract_financials(text):
58     """Extrahiert Finanzkennzahlen aus dem Transkript."""
59     data = {}
60     # Revenue absolute, sequential & YoY
61     m = re.search(r"Revenue of \$(\d+\.\d*) billion was up (\d+)%
        sequentially and up (\d+)% year on year", text, re.IGNORECASE)
62     if m:
63         data['Current revenue'] = f"${m.group(1)}B"
64         data['Sequential growth'] = f"{m.group(2)}%"
65         data['YoY growth'] = f"{m.group(3)}%"
66     # Full fiscal year revenue
67     fy = re.search(r"For fiscal (\d+) revenue was \$(\d+\.\d*) billion, up
        (\d+)%", text, re.IGNORECASE)
68     if fy:
69         data['FY revenue'] = f"${fy.group(2)}B ({fy.group(3)}% YoY)"
70     # Operating margin expansion
71     opm = re.search(r"operating margin.*?(\d+) basis points", text, re.
        IGNORECASE)
72     if opm:
73         data['Operating margin expansion'] = f"{opm.group(1)} bps"
74     # Forecast
75     fc = re.search(r"expected (?:in|for) (Q\d)[^\.]*", text, re.IGNORECASE)
76     if fc:
77         pct = re.search(r"([Mm]id[- ]teens)%? .* expected", text)
78         if pct:

```

```

79         data['Forecast'] = f"{pct.group(1).capitalize()} growth in {fc.
80             group(1)}"
81     return data
82
83 def generate_structured_summary(sentiment, topics, financials):
84     """Erstellt eine prägnante Zusammenfassung basierend auf Sentiment, Top
85         -Themen und Finanzdaten."""
86     topic_str = ', '.join(topics[:3])
87     fin_parts = [f"{k}: {v}" for k, v in financials.items()]
88     fin_str = '; '.join(fin_parts) if fin_parts else 'keine Finanzdaten
89         verfügbar'
90     return (
91         f"Sentiment der CEO-Eröffnung: {sentiment}. "
92         f"Hauptthemen: {topic_str}. "
93         f"Finanzdaten: {fin_str}. "
94         "Insgesamt deutet alles auf starkes Wachstumspotenzial hin."
95     )
96
97 def main():
98     transcript = read_transcript('earnings_call_transcript.txt')
99
100     # 1. Sentiment
101     sentiment, conf = analyze_sentiment(transcript)
102     print(f"Sentiment Analysis: {sentiment} (Confidence: {conf:.2f})\n")
103
104     # 2. Key Topics
105     topics = top_nouns(transcript)
106     print("Key Topics:")
107     for i, t in enumerate(topics, 1):
108         print(f"  {i}. {t}")
109     print()
110
111     # 3. Finanzdaten
112     financials = extract_financials(transcript)
113     print("Key Financial Data Points:")
114     if financials:
115         for k, v in financials.items():
116             print(f"  - {k}: {v}")
117     else:
118         print("  Keine Finanzdaten gefunden.")
119     print()
120
121     # 4. Executive Summary
122     summary = generate_structured_summary(sentiment, topics, financials)
123     print("Generated Executive Summary:")
124     print(summary)
125
126 if __name__ == '__main__':
127     main()

```

Listing 1: analysis.py mit Sentiment-, Topic- und Finanzanalyse



## **Github(branch:main)**

Wie immer sind alle meine benutzten Dateien auf meinem Github zu finden.