DAV
Deutsche
Aktuarvereinigung e.V.

Schriftliche Prüfung im Grundwissen

Data Science & Künstliche Intelligenz

gemäß Prüfungsordnung 6
der Deutschen Aktuarvereinigung e. V.

am 9. Mai 2026

Hinweise:

- Als Hilfsmittel ist ein Taschenrechner zugelassen.
- Die Gesamtpunktzahl beträgt 180 Punkte. Die Klausur ist bestanden, wenn mindestens 90 Punkte erreicht werden.
- Bitte prüfen Sie die Ihnen vorliegende Prüfungsklausur auf Vollständigkeit. Die Klausur besteht aus 7 Seiten.
- Alle Antworten sind zu begründen und bei Rechenaufgaben muss der Lösungsweg ersichtlich sein.
- Alle Antworten sind in deutscher Sprache zu erstellen (fremdsprachige Fachbegriffe ausgenommen)
- Bitte vermeiden Sie bei der Lösungserstellung die nicht zusammenhängende Streuung der Lösungen zu den einzelnen Aufgabenteilen.
- Aus Gründen der besseren Lesbarkeit wird auf die gleichzeitige Verwendung der Sprachformen männlich, weiblich und divers (m/w/d) verzichtet.

Mitglieder der Prüfungskommission:

Dr. Zoran Nikolić, Dr. Stefan Nörtemann, Prof. Dr. Anja Bettina Schmiedt, Prof. Dr. Fabian Transchel, Dr. Wiltrud Weidner

Aufgabe 1 [Gesellschaftliches Umfeld und Regulatorik] [45 Punkte]

Die neue KI-Abteilung der Pfefferminzia AG (bei der Sie seit kurzem als Aktuarin / Aktuar tätig sind) hat ein paar Ideen für neue KI-Anwendungen zusammengetragen. Ihre Vorgesetzte erinnert sich daran, Ihnen kürzlich die Teilnahme an dem Grundwissen-Seminar „Data Science & Künstliche Intelligenz“ genehmigt zu haben und zieht Sie deshalb zu den Beratungen hinzu.

a) [14 Punkte] Geben Sie an und begründen Sie, in welche Kategorie der Regulierung gemäß der KI-Verordnung (AI Act) die jeweilige KI-Anwendung fällt.
   (1) Clusterverdichtung zur Verbesserung der Performanz bei Projektionsrechnungen in der Lebensversicherung.
   (2) Kundenkommunikation mit einem Chatbot.
   (3) Tarifierung in der KfZ-Haftpflichtversicherung.
   (4) Automatisierte Abrechnung in der privaten Krankenversicherung.
   (5) Tarifierung in der Lebensversicherung.
   (6) Automatisierte Erstellung von Berichten im Financial Reporting (z.B. Solvency II).
   (7) Stornoprognose auf Basis eine Deep-Learning-Verfahrens.

b) [10 Punkte] Nennen Sie für Hochrisikoanwendungen fünf Anforderungen an den Betreiber der KI-Anwendung.

c) [5 Punkte] Ein Kollege schlägt vor, die obige Stornoprognose (Punkt (7)) auch für die Stornoprävention im Kundenmanagement einzusetzen. Wie schätzen Sie hier die Rechtmäßigkeit mit Blick auf den Datenschutz ein? Begründen Sie Ihre Antwort.

d) [5 Punkte] Da für einige der obigen Anwendungen keine Regulatorik vorgeschrieben ist, denkt Ihre Vorgesetzte darüber nach, eine Selbstregulierung in Form eines Verhaltenskodex zu etablieren. Was halten Sie von der Idee? Begründen Sie Ihre Antwort.

e) [11 Punkte] Ihre Vorgesetzte beauftragt Sie, einen Vorschlag für eine Selbstregulierung der Pfefferminzia AG für den Einsatz von KI-Anwendungen zu machen.
   (1) Nennen Sie zwei Quellen / Texte, in denen Sie Anregungen finden, welche Punkte eine solche Regulierung enthalten könnte.
   (2) Formulieren Sie eine Skizze für einen Vorschlag, in dem Sie drei Kernanforderungen, jeweils aufstellen.

Lösungsvorschlag:

Aufgabe 1

a) Anwendung (2) fällt in die Kategorie „Transparenzpflichten“, hier besteht die Pflicht, den Kunden darüber zu informieren, dass er mit einer KI kommuniziert. Anwendung (5) fällt gemäß Anhang III der KI-Verordnung in die Kategorie der Hochrisikosysteme.

Alle anderen Anwendungen fallen in die vierte Kategorie, für die keine regulatorischen Anforderungen vorgesehen sind. Hier könnte es ggf. eine Selbstverpflichtung geben. Jeweils 2 Punkte für die richtige Einordnung jeder Kategorie.

b) Für Hochrisikoanwendungen, wie z.B. Anwendung (5) aus Aufgabenteil a), bestehen es unter anderem folgende Anforderungen:

- Es ist ein Risikomanagement für die KI-Anwendung zu installieren.
- Benötigt wird eine Daten-Governance für KI-Anwendungen.
- Es gibt besondere Anforderrungen an die technische Dokumentation für die KI-Anwendung.
- Es wird eine Menschliche Aufsicht gefordert.
- Zudem bestehen besondere Anforderungen an Genauigkeit, Robustheit und Cybersicherheit der KI-Anwendung.

Je 2 Punkte für jede korrekt formulierte Anforderung.

c) Für eine Anwendung der KI-basierten Stornomodellierung im Kundenmanagement ist die Verarbeitung personenbezogener Daten notwendig, da sich eventuelle Maßnahmen zur Stornovermeidung an konkrete Versicherungsnehmer richten (1 Punkt). Diese Verarbeitung ist nur dann erlaubt, wenn einer der Verarbeitungsgründe aus Artikel 6 der Datenschutzgrundverordnung begründet werden kann (2 Punkte). In diesem Fall kann mit der Wahrung der berechtigten Interessen des Verantwortlichen (für die Verarbeitung), konkret den Interessen des Versicherungsunternehmens argumentiert werden (2 Punkte).

d) Eine unternehmensweite Selbstverpflichtung könnte sinnvoll sein, um nach außen zu dokumentieren, dass bei der Pfefferminzia AG ein ethischer Umgang mit KI-Anwendungen gepflegt wird, der über die regulatorischen Anforderungen hinausgeht. Zudem kann ein verbindlicher Verhaltenskodex Unsicherheiten im Umgang mit KI-Anwendungen für alle Beteiligten mindern (5 Punkte)

Alternative Lösung: Die Versicherungsbranche ist bereits umfänglich reguliert, so dass viele der ethischen Anforderungen, die in einer Selbstverpflichtung formuliert werden können, bereits anderweitig gefordert sind (VAG, VVG, …). Falls darüber hinaus ein Verhaltenskodex als wünschenswert angesehen wird, sollte man besser eine Branchenlösung

anstreben, wie sie sich im Code-of-Conduct Datenschutz des GDV hinlänglich bewährt hat.

e) Eine zentrale und fundierte Quelle für Ideen und Anregungen für einen Verhaltenskodex sind die „Ethics Guidelines for Trustworthy AI“ der High-Level Expert Group on Artificial Intelligence. Daneben ist auch die KI-Verordnung selbst eine geeignete Quelle. Weitere Anregungen finden sich im Standard IEEE 7000 oder in der Normungsroadmap Künstliche Intelligenz des Deutschen Instituts für Normung e. V. (DIN) (je 2 Punkte pro Quelle, maximal 4 Punkte).

Ein Verhaltenskodex könnte u.a. folgende Anforderungen enthalten. Die Pfefferminzia AG verpflichtet sich und alle ihre Mitarbeitenden (1 Punkt) zur

- Prüfung und Bewertung der Auswirkungen des Betriebs von KI-Systemen auf die ökologische Nachhaltigkeit und Entwicklung einer Strategie zur Minimierung negativer Auswirkungen;
- Bewertung und Verhinderung der negativen Auswirkungen von KI-Systemen auf schutzbedürftige Personen, einschließlich im Hinblick auf die Barrierefreiheit für Personen mit Behinderungen, sowie auf die Gleichstellung der Geschlechter;
- Erleichterung einer inklusiven und vielfältigen Gestaltung von KI-Systemen, unter anderem durch die Einsetzung inklusiver und vielfältiger Entwicklungsteams.

(Je 2 Punkte pro Anforderung).

Aufgabe 2 [Data Science – Von den Daten zum Modell zur Anwendung & KI machen] [45 Punkte]

In der Pfefferminzia AG soll aus der Fachabteilung, dem Aktuariat und der IT-Abteilung ein interdisziplinäres „Kompetenzzentrum KI“ gegründet werden, deren Teil Sie sind. In einem ersten Arbeitsschritt soll evaluiert werden, wie ein für das ganze Unternehmen kohärentes KI-Entwicklungsmodell aussehen kann und welche detaillierten Überlegungen dafür anzustellen sind.

a) [5 Punkte] Nennen Sie fünf verschiedene Anwendungen von Data Science und/oder Künstlicher Intelligenz in der Versicherungsbranche. Nennen Sie zu jeder Anwendung eine relevante, neuartige Datenquelle.

b) [9 Punkte] Beschreiben Sie Unterschiede und Gemeinsamkeiten von strukturierten, semi-strukturierten und unstrukturierten Daten und nennen Sie für jeden Typ zwei Beispiele.

c) [6 Punkte] Nennen Sie die Phasen eines Ihnen bekannten Prozessmodells für Data Mining (Anfertigung von Ablaufskizzen ist möglich, eine Beschreibung der jeweiligen Phasen ist nicht erforderlich) und erläutern Sie an einem der Beispiele aus der Teilaufgabe b), welche Tätigkeiten hier spezifisch durchgeführt werden würden.

d) [6 Punkte] Nennen und beschreiben Sie mit wenigen Worten drei unterschiedliche Plattformen und/oder Stacks, um Data Science- & KI-Anwendungen durchzuführen. Beschreiben Sie jeweilige Vor- und Nachteile stichpunktartig.

e) [10 Punkte] Auf Basis der in Teilaufgabe a) genannten Anwendungen evaluieren Sie für die in Teilaufgabe d) genannten Frameworks/Stacks, ob diese für die erforderlichen Aufgaben geeignet sind.

f) [9 Punkte] Fassen Sie abschließend zusammen, welcher Tech-Stack für Ihr „Kompetenzzentrum KI“ eingeführt und verwendet werden soll. Begründen Sie Ihre Antwort auf Basis der vorhergehenden Aufgaben.

Lösungsvorschlag

Aufgabe 2

a) Anwendungen mit Datenquellen:
   1. Pay as you Live in der Lebensversicherung auf Basis von Verhaltensdaten wie einem Schrittzähler oder anderen Vitality Trackern.
   2. Analyse der erwarteten Kundenprofitabilität auf Basis von Contact Points der Kundenbeziehung oder eines Clusterings des Kundenverhaltens nach Vertragskonstellation.
   3. Analyse von Fahrverhalten in der Kfz-Versicherung auf Basis von GPS-Daten und/oder Beschleunigungsdaten aus dem Fahrzeug.
   4. Echtzeitanalyse von Hagelschäden auf Basis von Radar- und Niederschlagsdaten.
   5. Erstellung von Risikokarten zu Überschwemmungen auf Basis von Klimamodellen und Geländemodellierungen.

b) *Strukturierte Daten*: Klassische tabellarisch vorliegende Daten, für die jede Zeile die gleichen Spalten aufweist. Beispiele sind Kundendaten oder Sterbetafeln.
Semi-Strukturierte Daten: Daten, deren charakteristische Eigenschaft darin liegt, dass ein Schema vorliegt (also alle möglichen Eigenschaften bekannt sind), aber nicht immer vorhanden/gefüllt sind, bzw. wechselnde Beziehungen haben können. Beispiele: Der konkrete zeitliche Verlauf einer individuellen Kundenbeziehung oder die resultierende Deckung bei zusammengesetzten Vertragsbausteinen.
Unstrukturierte Daten: Daten, die für einen Vorgang oder Prozess relevante Informationen enthalten, deren Schema aber unbekannt, mutabel oder zeitlich nicht stabil ist. Beispiele: Dokumente von Schadenaufnahmen oder Voice Messages im Kundenservice.

c) Das CRISP-DM steht für: Cross-Industry Standard Process for Data Mining und ist in sechs Phasen gegliedert. Beispielhafte Anwendung: Für die Analyse von unstrukturierten Daten, also bspw. Schadenakten könnten Aspekte in dem Zyklus wie folgt aussehen.

   1. Business Understanding: Klärung der Relevanz von automatisierter Erfassung und Auswertung von Schadenberichten als Effizienzgewinn.

   2. Data Understanding: Schadenakten sind semi-strukturiert, wobei Schadenmeldungen sogar unstrukturiert sein können. Das Data Understanding muss sicherstellen, dass Qualitäts- und Referenzkriterien vorhanden sind, anhand derer die Verarbeitung später verfolgt werden kann. Außerdem sind zulässige von unzulässigen Bedingungen für die Verarbeitung zu trennen, bspw. hinsichtlich der Sensitivität der Schadenmeldungen.

   3. Data Preparation: In der eigentlichen Verarbeitung gilt es, eine Fall-Taxonomie zu schaffen, anhand derer bspw. eine Entscheidungsmatrix für die weitergehende Verarbeitung abgeleitet werden kann. Außerdem müssen Sparten, Themen (sog. Sentiments) und Dringlichkeit extrahiert werden.

   4. Modelling: Ableitung bspw. eines Entscheidungsbaums dazu, welche Aspekte des Berichts gesondert (also z.B. dunkel) verarbeitet werden können auf Basis der zuvor festgelegten Fall-Taxonomie.

5. Evaluation: Prüfung, ob die vorhandenen Beispielfälle korrekt eingeordnet bzw. verarbeitet werden. Prüfung gegen die im Data Understanding definierten Qualitätskriterien. Prüfung von synthetischen oder individualisiert eingefügten Grenzfällen.
6. Implementierung: Verarbeitung von Fällen im Live-System mit einer die Modelle nutzenden Datenpipeline und fortlaufendem Monitoring.

d) Beispiele für Data Science-Stacks:
   1. Kaggle: Online-Framework für Modellierung auf virtuellen Servern. Gute Zugänglichkeit und Skalierbarkeit, aber große Datenschutzfragen.
   2. Google Colab: Online-Framework zur Modellierung auf Virtuellen Servern, sinnvollerweise unter API-Verbindung zu einem Data Warehouse in der Google Cloud. Vendor-Lockin muss geprüft werden.
   3. Anaconda Suite: Lokale oder netzwerkbasierte Lösung von selbstgepflegten Environments mittels z.B. Jupyter Notebooks. Größerer Administrationsaufwand als bei Cloud-Stacks.

e) Analyse:
   1. Pay as you Live in der Lebensversicherung: Cloud-Lösungen stellen aufgrund der hohen Sensitivität der Daten hohe Anforderungen, sodass eine on premise-Lösung mittels des genannten Anaconda-Stacks (oder vglb.) zweckmäßig erscheint.
   2. Analyse der erwarteten Kundenprofitabilität: Da Kundenverhaltensdaten in der Auswertung komplett anonym verarbeitet werden können, ist jede aufgeführte Infrastruktur geeignet. Performance-Flaschenhälse können in der Cloud besser umgangen werden.
   3. Analyse von Fahrverhalten in der Kfz-Versicherung: prinzipiell wie 2), allerdings könnte durch der Kundeneindruck einer „umfassenden Überwachung“ implizierte Reputationsrisiken durch on-premise-Auswertung eher vermieden werden.
   4. Echtzeitanalyse von Hagelschäden: Während eine lokale Verarbeitung auch möglich ist, scheint die Cloud für diesen Anwendungsfall deutlich besser geeignet, da alle erforderlichen Daten ohnehin online sind und Echtzeitfähigkeit on-premise schwieriger sicherzustellen ist.
   5. Erstellung von Risikokarten zu Überschwemmungen: Eine Anwendung, die für die Auslastung der lokalen Rechencluster durch ihre asynchrone und nicht-prioritäte Lastverteilung besonders gut geeignet ist.

f) Die Abwägung zwischen „Cloud Native“-Lösungen wie Google Colab und dem Einrichten von Entwicklungsumgebungen on Premise wie der Anaconda-Suite ist nicht immer einfach zu treffen. Während Skalierbarkeit und reduzierter Wartungsaufwand für Cloud-Lösungen sprechen, sind weitere gewichtige Faktoren der Datenschutz und komplette Kontrolle über das eigene Deployment nicht zu vernachlässigen. Während also für erste Proof-of-Concept-Anwendungen die Cloud ein guter Inkubator sein kann, ist für langfristige Sicherheit und die Vermeidung von Datenabflüssen und unabsehbaren Hintergrundkosten eine lokale Lösung als sinnvoller zu erachten. Eine strategische Ausrichtung muss auch die strategischen Zielsetzungen des Unternehmens ganz gezielt zur Abwägung heranziehen.

Aufgabe 3 [Überwachtes und unüberwachtes maschinelles Lernen] [45 Punkte]

a) [25 Punkte] Überwachtes Lernen

1. [3 Punkte] Beschreiben Sie kurz den Unterschied zwischen einem Regressions- und Klassifikationsproblem des überwachten Lernens. Veranschaulichen Sie ferner den Unterschied mit jeweils einem Beispiel aus der Versicherungsbranche.

2. [6 Punkte] Wenn ein überwachtes Lernverfahren auf Trainings- und Testdaten überprüft wird, können drei Fälle auftreten: Underfitting, Overfitting, Generalisierung. Erläutern Sie die drei Begriffe.

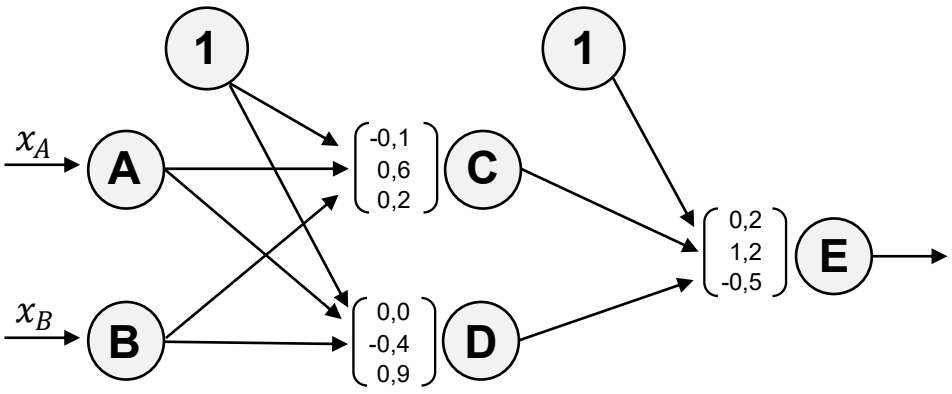
3. [3 Punkte] Für ein binäres Klassifikationsproblem wurde ein überwachtes Lernverfahren trainiert. Die Modellperformance wurde mittels einer ROC-Kurve auf den Trainings- und Testdaten bewertet. Beide ROC-Kurven stimmen nahezu überein und weisen jeweils einen AUC-Wert von ca. 0,88 auf.

   , Inwiefern eignet sich eine ROC-Kurve zur Beurteilung der Modellperformance? Würden Sie anhand der gegebenen Informationen für Underfitting, Overfitting oder eine Generalisierungsfähigkeit des Modells argumentieren? Begründen Sie Ihre Antwort.

4. [3 Punkte] Ein künstliches neuronales Netz ist formal als ein Tripel (N, V, w) definiert. Wie sind die einzelnen Bestandteile des Tripels definiert?

5. [4 Punkte] Für die Implementierung eines künstlichen neuronalen Netzes ist eine Aktivierungsfunktion f zu wählen. Welche Rolle spielt die Aktivierungsfunktion bei der Bestimmung des Outputs einer Unit? Geben Sie außerdem zwei Beispiele für Aktivierungsfunktionen mit Funktionsvorschrift und Funktionsnamen an.

6. [6 Punkte] Berechnen Sie für das abgebildete neuronale Netz einen Forwardpass für den Inputvektor $(x_A; x_B) = (0{,}4; 1{,}0)$, wenn als Aktivierungsfunktion *ReLu* (mit Schwellenwert Null) gewählt wird. Berechnen Sie dazu die Outputs der Units C, D und E.



b) [20 Punkte] Unüberwachtes Lernen

1. [2 Punkte] Beschreiben Sie die Grundidee (d.h. Ausgangspunkt und Ziel) der Clusteranalyse.

2. [2 Punkte] Skizzieren Sie zwei Anwendungsbeispiele der Clusteranalyse aus der Versicherungsbranche.

3. [5 Punkte] Erläutern Sie agglomerative Verfahren der Clusteranalyse anhand ihres algorithmischen Aufbaus.

4. [2 Punkte] Wie lautet die Formel zu Berechnung von Klassendistanzen $D(C_1, C_2)$ zwischen je zwei Clustern C_1, C_2 nach dem Complete-Linkage-Verfahren? Welcher alternative Name für das Verfahren leitet sich aus der Berechnungsvorschrift ab?

5. [9 Punkte] Gegeben sei eine Menge von Objekten $O = \{1,2,3,4\}$ mit folgender Objektdistanzmatrix:

$$\begin{pmatrix} 0 & 2 & 6 & 5 \\ 2 & 0 & 4 & 7 \\ 6 & 4 & 0 & 3 \\ 5 & 7 & 3 & 0 \end{pmatrix}$$

   Führen Sie ein agglomeratives Clusterverfahren mit Complete-Linkage durch. Geben Sie dabei in jedem Iterationsschritt das Clustering an.

Lösungsvorschlag Aufgabe 3:

a) [25 Punkte] Überwachtes Lernen

1. [3 Punkte]

   - Ist die Zielgröße in einem Problem des überwachten Lernens numerisch, sprechen wir von einem Regressionsproblem, ist die Zielgröße kategoriell, von einem Klassifikationsproblem. (1 Punkt)

   - Beispiel für eine numerische Zielgröße: Höhe eines Schadensbetrags (z.B. Kfz-, Hausrat-, Gebäudeversicherung); Lebensdauer oder Restlaufzeit einer Police; … (1 Punkt für Nennung eines Beispiels)

   - Beispiel für eine kategorielle Zielgröße: Kündigt der Kunde die Police? (Ja / Nein); Ist ein gemeldeter Schaden betrugsverdächtig? (Ja / Nein); …(1 Punkt für Nennung eines Beispiels)

2. [6 Punkte]

   - Underfitting: Die Vorhersagen auf den Trainingsdaten liegen – im Vergleich zu der bekannten Zielgröße – häufig daneben. Das Modell ist nicht in der Lage, die Komplexität der Daten zu erfassen und kann so weder auf Trainings- noch auf Testdaten gute Prognosen erstellen. (2 Punkte)

   - Overfitting: Die Vorhersagen auf den Trainingsdaten sind häufig (genug) – im Vergleich zu der bekannten Zielgröße – gut. Die Vorhersagen auf den Testdaten liegen jedoch häufig daneben, da das Modell zu stark an die Trainingsdaten anpasst ist und somit nicht in der Lage ist, die gesehenen Daten zu verallgemeinern. (2 Punkte)

   - Generalisierung: Die Vorhersagen sind sowohl auf den Trainingsdaten als auch auf den Testdaten häufig (genug) – im Vergleich zu der bekannten Zielgröße – gut. In diesem Fall sagt man, das Verfahren generalisiert, d.h. es ist in der Lage, das Gelernte gut auf neue Daten anzuwenden (Transferleistung). (2 Punkte)

3. [3 Punkte]

   - Eine ROC-Kurve zeigt die Trennfähigkeit eines binären Klassifikators, d.h. wie gut das Modell positive von negativen Fällen unterscheiden kann. Die AUC fasst diese Trennfähigkeit in einer Kennzahl als Flächeninhalt unter der Kurve zusammen. (1 Punkt)

   - Da Trainings- und Testdaten nahezu identische AUC-Werte von etwa 0,88 aufweisen, spricht die insgesamt hohe Trennleistung gegen Underfitting. Die

Übereinstimmung der AUC-Werte deutet zudem darauf hin, dass kein Overfitting vorliegt. Insgesamt spricht dies für eine gute Generalisierungsfähigkeit des Modells auf ungesehenen Daten. (2 Punkte)

4. [3 Punkte] Ein künstliches neuronales Netz ist ein Tripel (N, V, w) mit zwei Mengen N und V sowie einer reell wertigen Funktion w: $V \rightarrow \mathbb{R}$.

   - N ist die Menge der Neuronen (Units). (1 Punkt)
   - $V = \{(i,j) \mid i,j \in N\} \subseteq N \times N$ ist die Menge von Verbindungen zwischen je zwei Neuronen aus N. (1 Punkt)
   - w: $V \rightarrow \mathbb{R}$ definiert Gewichte, wobei $w((i,j))$ das Gewicht der Verbindung von Neuron i zu Neuron j ist und kurz mit $w_{i,j}$ bezeichnet wird. (1 Punkt)

5. [4 Punkte] Der Output einer Unit wird aus der Summe der gewichteten Inputwerte, (ggf.) einem Bias b und der Aktivierungsfunktion f berechnet:

   - Output = $f(x)$, $x =$ Summe der gewichteten Inputwerte + (ggf.) Bias b (2 Punkte)

   Beispiele für Aktivierungsfunktionen: (je 1 Punkt für ein Beispiel)

   - Rectified Linear Unit (*ReLU*): $f(x,\theta) = \max(0, x-\theta)$ mit Schwellenwert θ
   - Sigmoid Funktion: $f(x) = \frac{1}{1+e^{-x}}$
   - …

6. [6 Punkte] Forwardpass für den Inputvektor $(x_A;\ x_B) = (0{,}4;\ 1{,}0)$:

   - Output von Unit C: $ReLU(-0{,}1 + 0{,}4 \cdot 0{,}6 + 1{,}0 \cdot 0{,}2) = ReLU\ (0{,}34) = 0{,}34$ (2 Punkte)
   - Output von Unit D: $ReLU(0{,}4 \cdot (-0{,}4) + 1{,}0 \cdot 0{,}9) = ReLU\ (0{,}74) = 0{,}74$ (2 Punkte)
   - Output von Unit E: $ReLU(0{,}2 + 0{,}34 \cdot 1{,}2 - 0{,}5 \cdot 0{,}74) = ReLU\ (0{,}238) = 0{,}238$ (2 Punkte)

b) [20 Punkte] Unüberwachtes Lernen

1. [2 Punkte] Grundidee der Clusteranalyse:

- Ausgangspunkt: Gegeben sei eine Menge von Objekten $O = \{1, \dots, n\}$ mit jeweils p Merkmalen beschrieben durch die Merkmalsvektoren $x_1, \dots, x_n \in \mathbb{R}^p$. (1 Punkt)

- Ziel: Gesucht ist eine möglichst "gute" Partition (Clustering) $C = \{C_1, \dots, C_k\}$ bestehend aus den Klassen (Cluster) $C_1, \dots, C_k$ der n Objekte, sodass eine "hohe" Homogenität innerhalb der Cluster und eine "hohe" Heterogenität zwischen den Clustern besteht. (1 Punkt)

1. [2 Punkte] Anwendungsbeispiele der Clusteranalyse aus der Versicherungsbranche:

   - Portfoliokonsolidierung, z.B. für die Migration von Versicherungsbeständen (1 Punkt) oder für Projektionsrechnungen

   - Risikofaktorenanalyse, z.B. für die Berufsklassifizierung in der Berufsunfähigkeitsversicherung (1 Punkt) oder für die Bildung von Typen- und Regionsklassen in der Kfz-Versicherung

   - …

2. [5 Punkte] Agglomerativer Algorithmus:

   - Starte mit der feinsten Partition bestehend aus n Clustern $C^{(n)} = \{\{1\}, \{2\}, \dots, \{n\}\}$. (D.h. Ausgangspunkt ist die ***feinste*** Partition, bei der jedes der n Objekte ein eigenes Cluster bildet.) (1,5 Punkte)

   - Für $k = n, n-1, \dots, 3$:

     a) Berechne die Ähnlichkeiten (d.h. die Klassendistanzen) zwischen allen Clustern der Partition $C^{(k)}$ aus k Clustern. (1 Punkt)

     b) Fusioniere die beiden ähnlichsten Cluster (d.h. die mit der geringsten Klassendistanz) zu einem Cluster und erhalte die Partition $C^{(k-1)}$ aus $k-1$ Clustern. (1 Punkt)

     (D.h. in jedem Iterationsschritt werden zwei Cluster zu einem neuen, gröberen Cluster zusammengefasst.)

   - Fusioniere die verbleibenden $k = 2$ Cluster zu der Partition $C^{(1)} = \{\{1, 2, \dots, n\}\}$. (D.h. das Verfahren endet mit der ***gröbsten*** Partition, in der alle n Objekte in einem Cluster enthalten sind.) (1,5 Punkte)

3. [2 Punkte] Gegeben der Objektdistanzen $d_{ij}, i, j \in O$, ist die Berechnung von Klassendistanzen $D(C_1, C_2)$ zwischen je zwei Clustern C_1, C_2 im Complete-Linkage-Verfahren wie folgt definiert:

$D(C_1, C_2) = \max_{i \in C_1,\, j \in C_2} d_{ij}$ (1 Punkt)

Daraus leitet sich die alternative Bezeichnung „Furthest-Neighbor"-Verfahren ab. (1 Punkt)

4. [9 Punkte]

- Initiales Clustering: $C^{(4)} = \{\{1\},\{2\},\{3\},\{4\}\}$ (1 Punkt)

  Zugehörige Klassendistanzmatrix (= Objektdistanzmatrix):

$$\begin{pmatrix} 0 & 2 & 6 & 5 \\ 2 & 0 & 4 & 7 \\ 6 & 4 & 0 & 3 \\ 5 & 7 & 3 & 0 \end{pmatrix}$$

  Kleinstes Element ist $D(\{1\},\{2\}) = 2$. (1 Punkt)

- Fasse daher die beiden Cluster $\{1\}$ und $\{2\}$ zu einem Cluster zusammen und erhalte das Clustering $C^{(3)} = \{\{1,2\},\{3\},\{4\}\}$ (1 Punkt) mit den (neuen) Klassendistanzen nach Complete-Linkage: (2 Punkte)

  - $D(\{1,\ 2\},\{3\}) = \max\{6,\ 4\} = 6$
  - $D(\{1,\ 2\},\{4\}) = \max\{5,\ 7\} = 7$
  - $D(\{3\},\{4\}) = 3$

  Zugehörige Klassendistanzmatrix:

$$\begin{pmatrix} 0 & 6 & 7 \\ 6 & 0 & 3 \\ 7 & 3 & 0 \end{pmatrix}$$

  Kleinstes Element ist $D(\{3\},\{4\}) = 3$. (1 Punkt)

- Fasse daher die beiden Cluster $\{3\}$ und $\{4\}$ zu einem Cluster zusammen und erhalte das Clustering $C^{(2)} = \{\{1,\ 2\},\ \{3,\ 4\}\}$ (1 Punkt) mit den (neuen) Klassendistanzen nach Complete-Linkage:

  - $D(\{1,\ 2\},\{3,4\}) = \max\{6,\ 5,\ 4,\ 7\} = 7$ (1 Punkt)

  Zugehörige Klassendistanzmatrix:

$$\begin{pmatrix} 7 & 0 \\ 0 & 7 \end{pmatrix}$$

- Fasse die beiden verbleibenden Cluster zusammen und erhalte das Clustering $C^{(1)} = \{\{1,\ 2,\ 3,\ 4\}\}$. (1 Punkt)

Aufgabe 4 [Generative Künstliche Intelligenz] [45 Punkte]

(a) [6 Punkte] Beantworten Sie die folgenden Fragen – eine Begründung ist nicht erforderlich. Pro Teilaufgabe treffen zwei Aussagen zu. Bitte schreiben Sie jeweils höchstens zwei Antworten auf Ihren Lösungsbogen. Für jede richtige Nennung wird ein Punkt vergeben. Bei Nennung von drei oder vier Antworten werden keine Punkte vergeben.

(1) [2 Punkte] Welche der folgenden Punkte sind Limitationen von GenAI?

- Geringe Anwenderfreundlichkeit
- Thematisch begrenzte Anwendbarkeit
- Geringe Nachvollziehbarkeit
- Entstehung eines Bias

(2) [2 Punkte] Welche der folgenden Aussagen treffen auf RAGs (Retrieval Augmented Generation) zu?

- Gut geeignet für Code-Generierung und Übersetzung
- Zugriff auf aktuelle Echtzeitdaten
- Verwendung eines statischen Wissensstandes
- Keine Verwendung von LLMs

(3) [2 Punkte] Was sind Anwendungsbeispiele von Decoder-only Transformern?

- Autovervollständigung
- Maschinelle Übersetzung
- Codegenerierung
- Textzusammenfassung

(b) [4,5 Punkte] Wofür stehen die drei Buchstaben in der Abkürzung GPT? Erläutern Sie die drei Bestandteile in jeweils einem oder zwei Sätzen.

(c) [7,5 Punkte] Nennen die fünf Bestandteile in der Struktur des Decoder-only Transformers und erläutern Sie jeden der Bestandteile kurz.

(d) [10 Punkte] Es sei das folgende Beispiel-Korpus gegeben:

> „Eine **Formel** hilft bei der Analyse komplexer Daten. Eine gute **Formel** hilft dabei, präzise Vorhersagen zu treffen. Eine gute **Formel** wird schnell zum Standard. Eine schlechte **Formel** scheitert jedoch. Eine schlechte **Formel** scheitert,

wenn sie falsch ist. Eine schlechte **Formel** scheitert aber auch, wenn sie unnötig komplex ist. Eine gute **Formel** hilft hingegen ein möglichst einfaches und passendes Abbild der Realität zu schaffen. Am Ende entscheidet die Qualität, ob eine **Formel** hilft oder scheitert."

(1) [7 Punkte] Ermitteln Sie die Wahrscheinlichkeiten, die sich unter Verwendung des Bi-Gram-Modells für das nächste Wort des vom User vorgegebenen Satzes „Verwendet man eine schlechte Formel" auf Basis des Beispiel-Korpus ergeben. Nennen Sie alle Wörter, welche eine von Null verschiedene Wahrscheinlichkeit haben. Stellen Sie tabellarisch die Wahrscheinlichkeiten für diese Wörter dar.

(2) [1 Punkt] Welches Wort würde folgen, wenn Sie eine Greedy-Auswahl anwenden?

(3) [2 Punkte] Welches nächste Wort würden Sie bei einer Greedy-Auswahl für das Tri-Gram-Modell erhalten? Begründen Sie Ihre Antwort kurz.

Hinweis: Unterscheiden Sie zur Beantwortung der Frage nicht zwischen Groß- und Kleinschreibung.

(e) [17 Punkte] Beantworten Sie die folgenden Fragen:

(1) [2 Punkte] Was sind Embeddings und wozu werden sie genutzt?

(2) [7 Punkte] Erstellen Sie eine vollständige Co-Occurence Matrix für den Satz „**Diese Formel ist korrekt, aber jene Formel ist falsch.**" für die Fensterbreite $k = 2$.

(3) [2 Punkte] Nennen Sie einen Nachteil von Co-Occurence Matrizen und erklären Sie diesen anhand des von Ihnen ausgefüllten Ausschnitts der Matrix.

(4) [1 Punkt] Geben Sie die Count-based Embeddings der Wörter „Formel", „korrekt" und „falsch" an.

(5) [1 Punkt] Definieren Sie die Ähnlichkeit zwischen zwei Embeddings.

(6) [4 Punkte] Berechnen Sie auf Basis von Teilaufgabe (4) die Ähnlichkeit der Embeddings für das Paar („Formel", „korrekt") sowie für das Paar („Formel", „falsch").

Lösungsvorschlag Aufgabe 4:

(a) Lösungen:

(1) iii., iv.

(2) ii., iii.

(3) i., iii.

(b) **Generative:** Das Modell erzeugt Text durch autoregressiven Output, d. h. es sagt das nächste Token basierend auf den vorherigen voraus.

Pretrained: Das Modell wird zunächst auf großen Mengen öffentlicher Text-Korpora trainiert, eine spätere Verfeinerung erfolgt durch Supervised Fine-Tuning und Reinforcement Learning.

Transformer: Die Architektur basiert auf einem Decoder-only Transformer, der Masked Attention-Mechanismen nutzt, um Kontextinformationen effizient zu verarbeiten.

(c) Lösungen:

[1,5 Punkte] (I) Embedding: Token-Embedding mit Embedding-Matrix

[1,5 Punkte] (II) Positional Encoding: Position des Tokens im Text wird eingebunden

[1,5 Punkte] (III) Transformer-Blöcke: Iterative Kontextualisierung und nicht-lineare Transformation der Embeddings

[1,5 Punkte] (IV) Language Modeling Head: Wahrscheinlichkeitsverteilung des nächsten Tokens

[1,5 Punkte] (V) Sampling: Auswahl des nächsten Tokens

(d) Lösung:

(1) Das letzte Wort des Satzanfangs ist „Formel", sodass die folgenden Bi-Gram-Wahrscheinlichkeiten ermittelt werden müssen:

$$\hat{P}(hilft|Formel) = \frac{Count(Formel,\ hilft)}{Count(Formel)} = \frac{4}{8} = 0{,}5$$

$$\hat{P}(scheitert|Formel) = \frac{Count(Formel,\ scheitert)}{Count(Formel)} = \frac{3}{8} = 0{,}375$$

$$\hat{P}(wird|Formel) = 0{,}125$$

Für die Tabelle ergibt sich somit:

Wort	**Wahrscheinlichkeit**
hilft	0,5
scheitert	0,375
wird	0,125

Hinweis: Die Tabelle ist als Antwort ausreichend, der obige Rechenweg ist zur Beantwortung der Frage nicht zwingend nötig.

(2) Bei einer Greedy-Auswahl wird das Wort mit der höchsten Wahrscheinlichkeit gewählt, hier also „hilft".

(3) Beim Tri-Gram-Modell müssen die beiden letzten Wörter des Satzanfangs berücksichtigt werden, hier also die Sequenz (schlechte, Formel). Diese tritt im Beispielkorpus dreimal auf, jedoch immer gefolgt vom Wort „scheitert". Als Wahrscheinlichkeit ergibt sich also 1 für das darauffolgende Wort „scheitert". Bei einer Greedy-Auswahl würde das nächste Wort „scheitert" lauten.

(e) Lösung:

(1) Embeddings sind Vektorrepräsentationen von Wörtern oder Tokens, die deren Bedeutung in einem mathematischen Raum abbilden. Durch Berechnung der Ähnlichkeit zwischen zwei Vektoren/Embeddings (mittels des Kosinus bzw. des normierten Skalarprodukts) kann semantische Ähnlichkeit mathematisch erfasst werden. Embeddings werden beispielsweise für Transformer-Modelle genutzt.

(2) Die Co-Occurence Matrix von „Diese Formel ist korrekt, aber jene Formel ist falsch." sieht wie folgt aus:

	Diese	Formel	ist	korrekt	aber	jene	falsch
Diese	0	1	1	0	0	0	0
Formel	1	0	2	1	1	1	1
ist	1	2	0	1	1	1	1
korrekt	0	1	1	0	1	1	0
aber	0	1	1	1	0	1	0
jene	0	1	1	1	1	0	0
falsch	0	1	1	0	0	0	0

(3) Mehrere Antworten sind möglich; am naheliegendsten aufgrund des Beispiels ist die Größe/Dimension, die die Matrix bzw. Embeddings annehmen – bei einem kurzen (und in beiden Satzteilen sehr ähnlichen) Satz entsteht hier bereits eine 7x7 Matrix.

Eine andere plausible Antwort ist, dass die Matrix dünnbesetzt ist.

(4) Die Embeddings lauten:

$$v_{Formel} = (1,0,2,1,1,1,1)$$
$$v_{korrekt} = (0,1,1,0,1,1,0)$$
$$v_{falsch} = (0,1,1,0,0,0,0)$$

(5) Die Ähnlichkeit zwischen zwei Embeddings v und w wird quantifiziert als

$$\text{Ähnlichkeit}(v,w) := \frac{v \cdot w}{|v||w|} = \cos(\theta)$$

mit θ = Winkel zwischen v und w.

(6) Durch Einsetzen in obige Formel ergibt sich:

$$\text{Ähnlichkeit}(v_{Formel}, v_{korrekt}) := \frac{v_{Formel} \cdot v_{korrekt}}{|v_{Formel}||v_{korrekt}|} = \frac{2+1+1}{\sqrt{1+4+1+1+1+1}\cdot\sqrt{1+1+1+1}} = \frac{4}{\sqrt{36}} = \frac{2}{3} \approx 0{,}67$$

$$\text{Ähnlichkeit}(v_{Formel}, v_{falsch}) := \frac{v_{Formel} \cdot v_{falsch}}{|v_{Formel}||v_{falsch}|} = \frac{2}{\sqrt{1+4+1+1+1+1}\cdot\sqrt{1+1}} = \frac{2}{\sqrt{18}} \approx 0{,}47$$