LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

CENTRUM FÜR INFORMATIONS- UND SPRACHVERARBEITUNG
STUDIENGANG COMPUTERLINGUISTIK

KLAUSUR ZUM BACHELORMODUL „PROBEKLAUSUR VORLESUNG SYMBOLISCHE PROGRAMMIERSPRACHE“

PROBEKLAUSUR,
DR. BENJAMIN ROTH
KLAUSUR AM

VORNAME:	
NACHNAME:	
MATRIKELNUMMER:	
STUDIENGANG:	☐ B.Sc. Computerlinguistik, ☐ B.Sc. Informatik, ☐ Magister
	☐ anderer:

Die Klausur besteht aus **7 Aufgaben**. Die Punktzahl ist bei jeder Aufgabe angegeben. Die Bearbeitungsdauer beträgt **45 Minuten**. Bitte überprüfen Sie, ob Sie ein vollständiges Exemplar erhalten haben.
Tragen Sie die Lösungen in den dafür vorgesehenen Raum im Anschluss an jede Aufgabe ein. Falls der Platz für Ihre Lösung nicht ausreicht, benutzen Sie bitte **nur** die ausgeteilten Zusatzblätter!
Verwenden Sie einen dokumentenechten Kugelschreiber oder Füller, **keine** Bleistifte. Es sind **keine Hilfsmittel** zugelassen. Geben Sie Programmcode immer in **Python** an. **Sie können Fragen auf Englisch bearbeiten**. Bitte tragen Sie **zuerst**, d.h., bevor Sie die Aufgaben lösen, auf **allen** Seiten Ihren Namen ein und füllen Sie die Titelseite aus.

Aufgabe	mögliche Punkte	erreichte Punkte
1. Evaluierung von Klassifikatoren	4	
2. Naive Bayes Klassifikator	6	
3. Objektorientierung	5	
4. Klassifikation und Clustering	3	
5. NLTK and Lexical Information	6	
6. WordNet	3	
7. POS Tagging	3	
Summe	30	
Note		

Einwilligungserklärung (optional)
Hiermit stimme ich einer Veröffentlichung meines Klausurergebnisses in der Veranstaltung „PROBEKLAUSUR Vorlesung Symbolische Programmiersprache“ vom unter Verwendung meiner Matrikelnummer im Internet zu.

Datum: _____ Unterschrift: _____

NAME:

Aufgabe 1 Evaluierung von Klassifikatoren

Gegeben ein binärer Klassifikator für die Klassen `True` und `False`.

(a) Geben Sie Formel zur Berechnung von Precision, Recall und F1-Measure an (für Klasse `True`). Erklären Sie alle verwendeten Variablen.

$$Prec = \frac{TP}{TP+FP}$$

$$Rec = \frac{TP}{TP+FN}$$

$$F = \frac{2 \cdot Prec \cdot Rec}{Prec + Rec}$$

TP: Anzahl richtig als positiv klassifiziert
FP: Anzahl falsch als positiv klassifiziert
FN: Anzahl falsch als negativ klassifiziert
TN: Anzahl richtig als negativ klassifiziert

(b) Geben Sie Formel zur Berechnung der Accuracy an. Erklären Sie alle verwendeten Variablen.

4 PUNKTE

$$Acc = \frac{TP+TN}{TP+TN+FP+FN}$$

NAME:

Aufgabe 2 Naive Bayes Klassifikator

(a) Wie lautet bei einem Binären Textklassifikator (Naive Bayes) das Entscheidungskiterium?

$P(\text{True}|\text{text}) > P(\text{False}|\text{text})$ $\Leftrightarrow$ prediction = True

$\downarrow$

$\frac{P(\text{True}|\text{text})}{P(\text{False}|\text{text})} > 1$

(b) Was sind die sogenannten Log-Odds, und wie ergeben sie sich aus dem Entscheidungskiterium?

$\Leftrightarrow \frac{P(\text{text}|\text{True}) \cdot P(\text{True})}{P(\text{text}|\text{False}) \cdot P(\text{False})} > 1$ // Satz v. Bayes

$\Leftrightarrow$ // Anwendung d. Logarithmus

$\log P(\text{text}|\text{True}) + \log P(\text{True}) - \log P(\text{text}|\text{False}) - \log P(\text{False}) > 0$

(c) Erklären Sie das Konzept der **bedingten Unabhängigkeitannahme** am Beispiel der Berechnung der Wahrscheinlichkeit für $P(\text{Text}|\text{Label})$.

Wenn das Label bekannt ist, sind die Wörter statistisch unabhängig, die Wahrscheinlichkeit des Textes ist das Produkt der Wortwahrscheinlichkeiten.

$P(\text{text}|\text{label}) = P(w_1, w_2 \dots w_n|\text{label})$

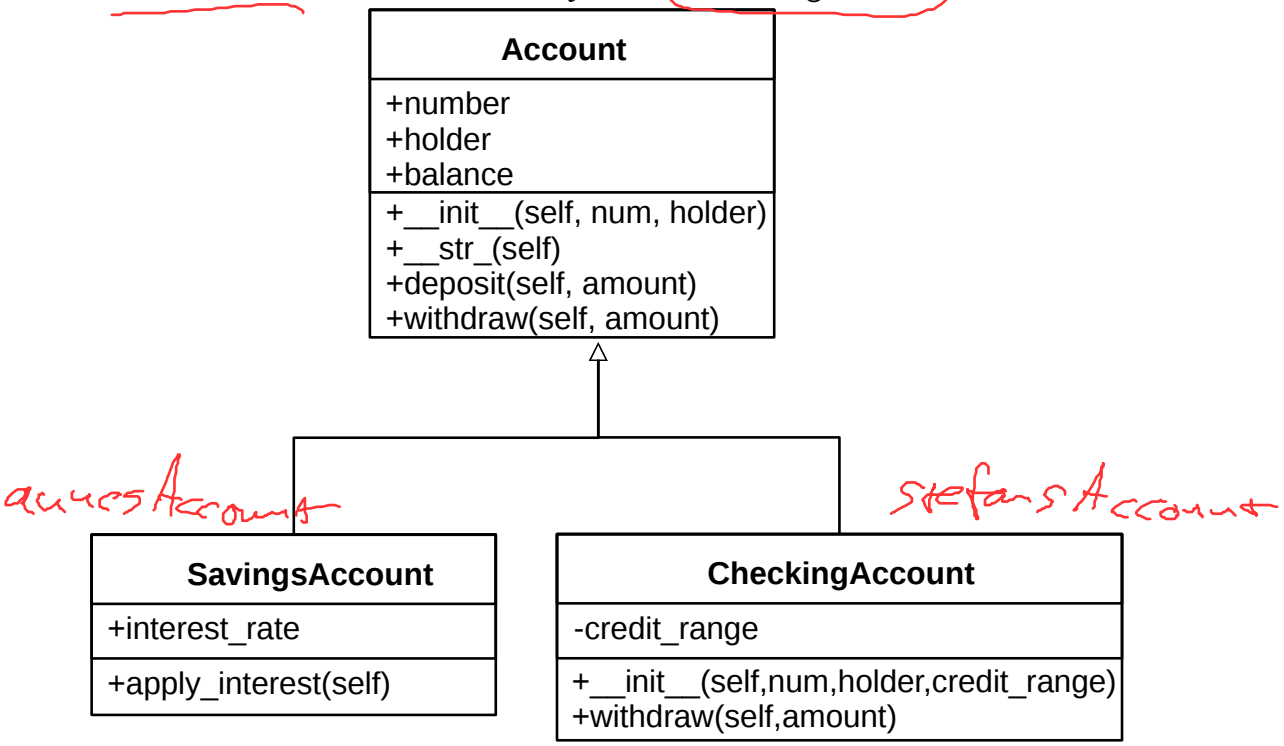
6 PUNKTE

$\overset{\text{bed. Unabh.}}{=} P(w_1|\text{label}) \cdot P(w_2|\text{label}) \cdot \dots \cdot P(w_n|\text{label})$

NAME:

Aufgabe 3 Objektorientierung

Gegeben die UML-Klassenhierarchie unten. `annesAccount` ist ein Instanzobjekt von `SavingsAccount`, und `stefansAccount` ist ein Instanzobjekt von `CheckingAccount`.



Geben Sie für die folgenden Aufrufe jeweils an, ob sie definiert sind, und wenn ja, in welcher Klasse die aufgerufene Methode definiert wurde:

- `SavingsAccount(2, "Anne")`
- `CheckingAccount(5, "Stefan", 300)`
- `str(annesAccount)`
- `str(stefansAccount)`
- `annesAccount.deposit(200)`
- `stefansAccount.deposit(500)`
- `annesAccount.withdraw(300)`
- `stefansAccount.withdraw(300)`
- `annesAccount.apply_interest()`
- `stefansAccount.apply_interest()`

5 PUNKTE

NAME:

Aufgabe 4 Klassifikation und Clustering

(a) Was ist der Unterschied zwischen `überwachtem Lernen (supervised)` und `unüberwachtem Lernen (unsupervised)` (Antwort in einem Satz)? (1 Punkt)

überwachtes Lernen: Es werden annotierte Daten verwendet.
Ziel: Vorhersage der Annotation, z.B. Klassifikation.

unüberwachtes Lernen: Die verwendeten Daten sind nicht annotiert.
Ziel: Finden von Struktur in den Daten, z.B.
durch Clustering.

(2 Punkte)

(b) Welche der folgenden Aussagen sind wahr?

(a) K-means ist ein unüberwachter Algorithmus. wahr. (Clustering)
(b) K-nearest neighbors ist ein überwachter Algorithmus wahr. (Klassifikation)
(c) Naive Bayes ist ein unüberwachter Algorithmus. falsch (Klassifikation)
(d) Lesk ist ein überwachter Algorithmus falsch (Regelbasiert - weder überwacht noch unüberwacht)

3 PUNKTE

NAME:

Aufgabe 5 NLTK and Lexical Information

(a) Definieren Sie folgende Begriffe: (2 Punkte)

(a) Token Vorkommen eines Wortes (bzw einer Zeichensequenz)

(b) Type Wortform, Zeichensequenz als Eintrag im Lexikon/Vokabular.

(c) Collocation

Zwei oder mehr Wörter, die besonders häufig zusammen vorkommen.

(d) Bigram

Zwei aufeinander folgende Tokens.

(b) Nennen Sie zwei Beispiele der Lexika, die es im NLTK gibt. Für welche NLP Aufgaben können Sie diese Lexika verwenden? (2 Punkte)

nltk.corpus.stopwords → Suchmaschinen, herausfiltern nicht-informativer Wörter

nltk.corpus.words → Rechtschreibprüfung

nltk.corpus.names → Anaphern-Resolution

(c) Gegeben folgender Programmcode: (2 Punkte)

```
1  import nltk
2
3  text = nltk.corpus.genesis.words("english-kjv.txt")
4  bigrams = nltk.bigrams(text)
5  cfd = nltk.ConditionalFreqDist(bigrams)
6
7  print(list(cfd["living"]))
8  >>>['creature', 'thing', 'soul', '.', 'substance', ',']
9
10 print(list(cfd["living"].values()))
11 >>> [7, 4, 1, 1, 2, 1]
12
13 result = cfd["living"].max()
```

Was wird in der Zeile 13 berechnet?

Wort, welches am häufigsten nach 'living' vorkommt.

Was ist der Inhalt der Variable `result`?

6 PUNKTE

'creature'

NAME:

Aufgabe 6 WordNet

(a) Erklären Sie kurz die Idee des Lesk-Algorithmus. (1 Punkt)

Ziel: Word-sense Disambiguation (WSD)
Idee: Es wird verglichen, welche Wörter im Kontext eines Wortvorkommens auch in den Definitionen der Wortbedeutungen vorkommen.

(b) Die unten angegebene Tabelle zeigt 2 Bedeutungen von dem Wort `"bank"`. In welcher Bedeuting wird dieses Wort laut Lesk-Algorithmus im Satz `"Where do you bank in this town?"` benutzt? Begründen Sie Ihre Antwort. (2 Punkte)

Sense	Definition
Synset('bank.v.03')	do business with a bank or keep an account at a bank
Synset('deposit.v.02')	put into a bank account

3 PUNKTE

⇒ bank.v.03 (Overlap 3 vs. 1)

NAME:

Aufgabe 7 POS Tagging

Gegeben die Hypothese: `Ein Satz endet niemals mit einer Präposition (preposition).` Beschreiben Sie, wie Sie diese Hypothese mit Hilfe von NLTK verifizieren können. 3 PUNKTE

1. Alle Sätze eines Korpus POS-taggen (oder manuelle Annotation laden falls vorhanden)
2. Conditional Frequency Distribution der POS-Tags erstellen.
3. Abfragen ob 'PUNCT' als POS-tag nach 'PRP' vorkommt. (dann ist Hypothese widerlegt, ansonsten können wir annehmen, dass sie stimmt)