**Dokumentation:   
Projekt 2 – Phase 2 --- Yelp-Dataset Analyse**

**3.1 Verteilung der Sternebewertungen nach Filterung analysieren**

**Warum:**

Es ist wichtig, die Verteilung der Sternebewertungen zu verstehen, um zu erkennen, ob bestimmte Bewertungsstufen dominieren und wie die Filterung die Verteilung beeinflusst hat.

**Wie:**

Ein Kreisdiagramm (Tortendiagramm) wird erstellt, um die prozentuale Verteilung der Sternebewertungen zu visualisieren. Dabei werden 3-Sterne-Bewertungen ausgeschlossen, da sie meist neutral sind und die Analyse fokussieren soll.

**Code:**

**Ergebnis:**

Das Kreisdiagramm zeigt die prozentuale Verteilung der Sternebewertungen, nachdem die 3-Sterne-Bewertungen ausgeschlossen wurden:

* 5 Sterne: 45.9 %
* 4 Sterne: 22.2 %
* 2 Sterne: 7.8 %
* 1 Stern: 13.9 %

Ein Bild, das Text, Diagramm, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

**3.2 Verteilung der Labels: Positive vs. Negative Bewertungen**

**Warum?**

Die Klassifizierung der Bewertungen in positive und negative Kategorien ist ein zentraler Schritt, um das Vorhersagemodell für Stimmungen oder Sternebewertungen zu erstellen. Wir möchten die Verteilung der Labels untersuchen und sicherstellen, dass die Daten ausgewogen oder zumindest gut interpretierbar sind

**Wie?**

1. Wir haben nur Bewertungen mit 1, 2, 4 und 5 Sternen beibehalten   
   (3 Sterne wurden ausgeschlossen, da sie als neutral gelten).
2. Labels wurden wie folgt erstellt:
   * 1.0 (positiv): Bewertungen mit 4 oder 5 Sternen.
   * 0.0 (negativ): Bewertungen mit 1 oder 2 Sternen.
3. Die Verteilung der Labels wurde analysiert und anschließend visuell dargestellt, um das Verhältnis der positiven und negativen Bewertungen besser zu verstehen

Ein Bild, das Text, Schrift, Software, Webseite enthält.

Automatisch generierte Beschreibung**Code**

**Ergebnis**

1. **Label-Verteilung:**
   * **Positive Bewertungen (Label 1):** 680.509 Einträge
   * **Negative Bewertungen (Label 0):** 216.537 Einträge
   * Positive Bewertungen sind in der Mehrzahl, was typisch für viele Bewertungsdatensätze ist
2. **Visuelle Darstellung:**
   * Die Verteilung wurde in einem Balkendiagramm dargestellt
   * Es zeigt deutlich, dass es viel mehr positive als negative Bewertungen gibt

**Diagramm**

Ein Bild, das Text, Screenshot, Schrift, Zahl enthält.

Automatisch generierte Beschreibung

**3.3 Feature-Auswahl**

**Warum?**

Um ein Vorhersagemodell zu erstellen, benötigen wir sinnvolle und gut definierte Eingabefeatures. Diese Features sollten relevante Informationen enthalten, die es dem Modell ermöglichen, die Zielvariable (z. B. positive oder negative Bewertung) präzise vorherzusagen

**Wie?**

1. **Textinhalte:** Die Hauptinformation in unserem Datensatz liegt im Bewertungstext (text). Daher wird der Text als primäres Feature verwendet
2. **Textmerkmale:** Wir berechnen zusätzliche textbasierte Eigenschaften wie die Länge der Bewertung, die Anzahl der Wörter oder spezifische Wortmerkmale
3. **Sternebewertung (stars\_x):** Wir verwenden die Sternebewertung als zusätzliches Feature zur Unterstützung der Analyse
4. **Kategorisierung:** Branchen- oder Geschäftskategorien (categories) können ebenfalls als Feature verwendet werden

**Code**



**Ergebnis**

1. **Hauptfeatures:**
   * text: Der vollständige Text der Bewertung.
   * text\_length: Länge der Bewertung (Anzahl der Zeichen).
   * stars\_x: Die ursprüngliche Sternebewertung.
   * categories: Kategorien, zu denen das bewertete Geschäft gehört.
   * label: Die Zielvariable (0 für negativ, 1 für positiv).
2. **Zusätzliche Merkmale:**
   * word\_count: Die Anzahl der Wörter in der Bewertung.

Ein Bild, das Text, Screenshot, Schrift, Zahl enthält.

Automatisch generierte Beschreibung

**3.4 Klassifikationsbericht, Accuracy und Confusion Matrix**

**Warum?**

Der Klassifikationsbericht und die Confusion Matrix sind essenzielle Werkzeuge, um die Leistung des Modells zu bewerten. Sie helfen uns, die Genauigkeit, Präzision, und die Fähigkeit des Modells, verschiedene Klassen (positive und negative Bewertungen) korrekt zu erkennen, zu verstehen.

**Wie?**

1. **Klassifikationsbericht:**
   * Erstellt mit der Funktion classification\_report aus sklearn
   * Analysiert die Metriken Precision, Recall und F1-Score für jede Klasse   
     (0 = negativ, 1 = positiv).
   * Betrachtet den Gesamtdurchschnitt (macro avg und weighted avg),   
     um die Gesamtleistung des Modells zu bewerten
2. **Confusion Matrix:**
   * Erstellt mit der Funktion confusion\_matrix aus sklearn
   * Visualisiert mit einer Heatmap
   * Zeigt die Anzahl der richtig und falsch klassifizierten Bewertungen pro Klasse

**Code:**

**Ergebnis:**

1. **Confusion Matrix:**
   * Zeigt, dass:
     + **40.026** Bewertungen fälschlicherweise als positiv vorhergesagt wurden (falsch positiv)
     + **3.168** Bewertungen korrekt als negativ erkannt wurden   
       (richtig negativ)
     + **133.951** Bewertungen korrekt als positiv erkannt wurden   
       (richtig positiv)
     + **2.265** Bewertungen fälschlicherweise als negativ vorhergesagt wurden (falsch negativ)

Ein Bild, das Text, Screenshot, Schrift, Zahl enthält.

Automatisch generierte Beschreibung

1. **Klassifikationsbericht:**
   * Precision, Recall und F1-Score sind für Klasse 1 (positive Bewertungen) hoch, jedoch niedrig für Klasse 0 (negative Bewertungen)
   * Die Gesamtgenauigkeit liegt bei **76 %**, was für ein Textklassifizierungsmodell in Ordnung, aber verbesserungswürdig ist

Ein Bild, das Text, Screenshot, Schrift, Zahl enthält.

Automatisch generierte Beschreibung

1. **Genauigkeit:**  
   Genauigkeit des Modells: 0.76

**Fazit:**

Der Klassifikationsbericht und die Confusion Matrix zeigen, dass das Modell positive Bewertungen sehr gut erkennt, jedoch Schwierigkeiten hat, negative Bewertungen zu klassifizieren. Verbesserungen könnten durch bessere Features, Balancierung der Klassen oder spezifische Optimierungen des Modells erzielt werden.

**3.5 Darstellung der Verteilung der Sternebewertungen mit KNIME**

**Warum?**

Die Verteilung der Sternebewertungen (nach der Filterung auf 4 und 5 Sterne) soll visuell dargestellt werden, um die Häufigkeiten dieser Bewertungen besser zu verstehen. Dies dient der Überprüfung und Interpretation der Datenqualität sowie der Vorbereitung für weitere Analysen.

**Wie?**

1. **Datenquelle**:
   * Die Daten wurden aus einer großen CSV-Datei (vorher erstellt und gefiltert) in KNIME importiert.
   * Relevante Spalten: stars\_x (Sternebewertungen) und label (Zielvariable für positive Bewertungen).
2. **Schritte in KNIME**:
   * **CSV Reader**: Die Datei wurde mit dem CSV Reader importiert.
   * **Column Filter**: Es wurden nur die Spalten stars\_x, text und label behalten.
   * **Row Filter**: Nur Bewertungen mit 4 und 5 Sternen wurden eingeschlossen.
   * **Domain Calculator**: Die Metadaten für die Spalten wurden aktualisiert, um eine korrekte Verarbeitung zu ermöglichen.
   * **String Manipulation**: Die Spalte stars\_x wurde in einen String umgewandelt, um für die Kategoriedimension im Bar Chart Node verfügbar zu sein.
   * Ein Bild, das Text, Diagramm, Reihe, Schrift enthält.

     Automatisch generierte Beschreibung**Bar Chart Node**: Ein Balkendiagramm wurde erstellt, um die Verteilung der Bewertungen darzustellen.
3. **Visualisierung**:
   * Die X-Achse repräsentiert die Sternebewertungen (4 und 5).
   * Die Y-Achse zeigt die Anzahl der Bewertungen.
   * Ein Bild, das Text, Screenshot, Diagramm, Zahl enthält.

     Automatisch generierte BeschreibungJede Bewertungskategorie wird farblich unterschieden.

**Ergebnis**

* **Interpretation des Balkendiagramms**:
  + Bewertungen mit **5 Sternen** sind deutlich häufiger als solche mit 4 Sternen.
  + Dieses Muster entspricht den Erwartungen, da 5-Sterne-Bewertungen oft die Mehrheit positiver Bewertungen ausmachen
* **Diagramm**: Das Diagramm zeigt deutlich, dass die meisten Bewertungen 5 Sterne erhalten haben, gefolgt von 4 Sternen

**Technische Herausforderungen und Lösungen:**

1. **Problem**: Die Spalte stars\_x war zunächst nicht auswählbar.
   * **Lösung**: Mithilfe des Domain Calculator Nodes wurden die Metadaten aktualisiert. Anschließend wurde die Spalte mit dem String Manipulation Node in einen String umgewandelt
2. **Problem**: Das Diagramm war anfangs unleserlich.
   * **Lösung**: Optimierung der Filter- und Visualisierungsoptionen sowie Bereinigung des Workflows