Einleitung und Motivation

In einer zunehmend digitalisierten Welt, in der die Kommunikation primär über E-Mail erfolgt, sind Spam und Phishing zu allgegenwärtigen Bedrohungen geworden. Millionen unerwünschter E-Mails durchfluten täglich die Posteingänge und stellen eine ernsthafte Gefahr für die Sicherheit und Effizienz der Kommunikation dar. Diese Herausforderung erfordert zuverlässige und präzise Lösungen zur Spam-Erkennung, um Nutzer vor betrügerischen Aktivitäten zu schützen und ihre Posteingänge sauber zu halten. Unsere Motivation für diese Arbeit entspringt der Notwendigkeit, fortschrittliche prädiktive Analysen für die Erkennung von Spam-E-Mails bereitzustellen. Mit der zunehmenden Verfügbarkeit von Daten und den Fortschritten im Bereich des maschinellen Lernens haben sich neue Möglichkeiten eröffnet, um diese Bedrohungen effektiv zu bekämpfen. Ziel dieser Dokumentation ist es, eine transparente und zugängliche Plattform zu schaffen, die es Nutzern ermöglicht, die Ergebnisse unserer Modelle zu verstehen und deren Leistung zu visualisieren (K. Debnath and N. Kar, 2021).

Related Work

In der Forschung zur Spam-Erkennung haben maschinelle Lernmethoden bereits signifikante Fortschritte erzielt. Verschiedene Studien haben gezeigt, dass Algorithmen wie Naive Bayes, Support Vector Machines (SVM), und Entscheidungsbäume effektiv zur Identifizierung von Spam-E-Mails eingesetzt werden können (Olatunji, Sunday Olusanya, 2019)

Eine der frühesten und weit verbreiteten Techniken zur Spam-Erkennung ist der Naive Bayes-Algorithmus, der aufgrund seiner Einfachheit und Effizienz in vielen E-Mail-Filtern implementiert wurde. Weiterentwicklungen in der maschinellen Lernforschung haben jedoch komplexere Modelle hervorgebracht, die eine höhere Genauigkeit und Robustheit bieten.

Support Vector Machines (SVM) sind eine dieser fortschrittlicheren Techniken, die besonders für ihre Fähigkeit bekannt sind, klare Trennungen zwischen Spam und legitimen E-Mails zu finden. Studien haben gezeigt, dass SVMs, insbesondere in Kombination mit Textverarbeitungsmethoden wie TF-IDF (Term Frequency-Inverse Document Frequency), bemerkenswerte Ergebnisse in der Spam-Erkennung liefern können (Olatunji, Sunday Olusanya, 2019).

Entscheidungsbäume und deren erweiterte Formen, wie Random Forests und Gradient Boosting Machines (GBMs), haben ebenfalls ihre Wirksamkeit bewiesen. Diese Modelle profitieren von ihrer Fähigkeit, komplexe Muster und Interaktionen zwischen Merkmalen zu lernen, was zu einer höheren Vorhersagegenauigkeit führt (Taylor, O. E., and P. S. Ezekiel,2020).

Neuere Forschungsarbeiten haben sich auch auf tiefe neuronale Netze konzentriert, insbesondere Convolutional Neural Networks (CNNs) und Recurrent Neural Networks (RNNs) (Sharmin, Tazmina,2020). Diese Netzwerke, die ursprünglich für Bild- und

Sprachverarbeitung entwickelt wurden, haben auch in der Textklassifizierung beeindruckende Erfolge erzielt. Sie sind in der Lage, kontextuelle Informationen und sequenzielle Abhängigkeiten besser zu erfassen, was die Genauigkeit der Spam-Erkennung weiter verbessert.

Zusammengefasst zeigen diese Studien, dass maschinelles Lernen ein mächtiges Werkzeug zur Bekämpfung von Spam ist. Die kontinuierliche Weiterentwicklung dieser Techniken trägt dazu bei, die Effizienz und Genauigkeit von Spam-Filtern zu erhöhen, was wiederum die Sicherheit und Zuverlässigkeit der E-Mail-Kommunikation stärkt. Unsere Arbeit baut auf diesen bestehenden Ansätzen auf und zielt darauf ab, die Leistung verschiedener Spam-Erkennungsmodelle interaktiv zu test- und vergleichbar zu machen. So können Anwender die Modelle nicht nur theoretisch bewerten, sondern auch praktisch erleben und deren Vorhersagen in Echtzeit nachvollziehen.

Datenvorbereitung und Explorative Datenanalyse (EDA)

In der Datenvorbereitung und explorativen Datenanalyse (EDA) haben wir mehrere Datensätze ausprobiert, um unsere Modelle optimal anzulernen. Zu Beginn der Analyse haben wir verschiedene öffentlich zugängliche Spam-Datensätze gesammelt, die eine breite Palette von E-Mail-Nachrichten und deren Klassifikationen als Spam oder Ham (nicht-Spam) enthielten.

Der erste Schritt bestand darin, diese Datensätze zu säubern und zu vereinheitlichen. Dies umfasste die Entfernung von irrelevanten Spalten und die Bereinigung von unvollständigen oder fehlerhaften Daten. Unser Ziel war es, die Datensätze auf die wesentlichen Merkmale zu reduzieren, die für die Spam-Erkennung entscheidend sind.

Im Verlauf der Datenaufbereitung haben wir festgestellt, dass viele Datensätze zusätzliche Informationen wie Absenderadresse, Betreffzeile und Empfangszeit, oder NaN-Werte enthielten. Für unsere Zwecke konzentrierten wir uns jedoch ausschließlich auf die Nachricht selbst und deren Klassifikation. Daher wurden alle anderen Spalten gelöscht, sodass nur noch die Kategorie (Spam oder Ham) und die Nachricht verbleiben.

Diese Reduktion auf die wesentlichen Merkmale ermöglichte es uns, uns auf den Inhalt der Nachrichten zu konzentrieren und die Modelle effizienter zu trainieren. Durch die Anwendung von Textverarbeitungsmethoden wie Tokenisierung, Stemming und Stopword-Entfernung bereiteten wir die Textdaten weiter vor, um sie für maschinelle Lernalgorithmen nutzbar zu machen.

Während der EDA haben wir verschiedene Visualisierungen erstellt, um die Verteilung der Spam- und Ham-Nachrichten zu untersuchen und potenzielle Muster oder Unterschiede zu erkennen. Wir analysierten die Wortfrequenzen und -längen, um Einblicke in die typischen Merkmale von Spam-Nachrichten zu gewinnen.

Durch diese sorgfältige Datenvorbereitung und -analyse konnten wir sicherstellen, dass unsere Modelle auf einem soliden und gut strukturierten Datensatz basieren, was letztendlich zu einer verbesserten Vorhersagegenauigkeit beitrug.

Modell Evaluation

Die Modellbewertung ist ein zentraler Bestandteil unseres Projekts zur Spam-Erkennung. Hier stellen wir die Ergebnisse unserer Modelle vor und erläutern die verwendeten Metriken zur Leistungsbewertung.

Wir haben mehrere maschinelle Lernmodelle trainiert und evaluiert, darunter:

1. Naive Bayes
2. Support Vector Machines (SVM)
3. Decision Tree
4. k-Nearest Neighbors (KNN)
5. Random Forest

Zur Bewertung der Modelle haben wir die folgenden Metriken verwendet:

- **Genauigkeit**: Der Anteil der korrekt klassifizierten E-Mails (Spam und Ham) an der Gesamtzahl der E-Mails.
- **Präzision**: Der Anteil der korrekt als Spam klassifizierten E-Mails an allen als Spam klassifizierten E-Mails.
- **Recall (Sensitivität)**: Der Anteil der korrekt als Spam klassifizierten E-Mails an allen tatsächlichen Spam-E-Mails.
- **F1-Score**: Das harmonische Mittel von Präzision und Recall, um ein ausgewogenes Maß der Modellleistung zu erhalten.

Nach dem Training und der Evaluierung unserer Modelle haben wir folgende Ergebnisse erzielt:

	Model	Accuracy	Precision	Recall	F1-Score
0	Naive Bayes	0.9659	0.9837	0.7707	0.8643
1	SVM	0.9821	0.9597	0.9108	0.9346
2	KNN	0.9327	1	0.5223	0.6862
3	Decision Tree	0.9641	0.8824	0.8599	0.871
4	Random Forest	0.9794	1	0.8535	0.921

Abbildung 1: Modell-Metrik-Tabelle

Die Ergebnisse zeigen, dass alle Modelle eine hohe Genauigkeit bei der Erkennung von Spam-E-Mails erreichen. Insbesondere Support Vector Machines (SVM) hat mit einer Genauigkeit von 98,2% und einem F1-Score von 93,5% die beste Leistung unter den getesteten Modellen gezeigt.

Der Naive Bayes-Algorithmus lieferte solide Ergebnisse, insbesondere wegen seiner Einfachheit und Effizienz bei der Textklassifizierung, obwohl er leicht hinter den leistungsfähigeren Modellen wie Random Forest und SVM zurückblieb.

Decision Trees und k-Nearest Neighbors (KNN) lieferten respektable Ergebnisse, waren jedoch insgesamt weniger präzise und robust im Vergleich zu Random Forest und SVM. Dies ist verständlich, da Decision Trees anfällig für Überanpassung sind und KNN bei großen Datensätzen und hoher Dimensionalität weniger effizient ist.

Abbildung 2 visualisiert die Leistungen der verschiedenen Modelle in Bezug auf die oben genannten Metriken.

Durch die umfassende Evaluierung unserer Modelle konnten wir die Stärken und Schwächen jedes Ansatzes identifizieren und das am besten geeignete Modell für unsere Spam-Erkennungslösung auswählen. Dies gewährleistet eine zuverlässige und präzise Filterung von Spam-E-Mails, was die Sicherheit und Effizienz der E-Mail-Kommunikation erheblich verbessert.

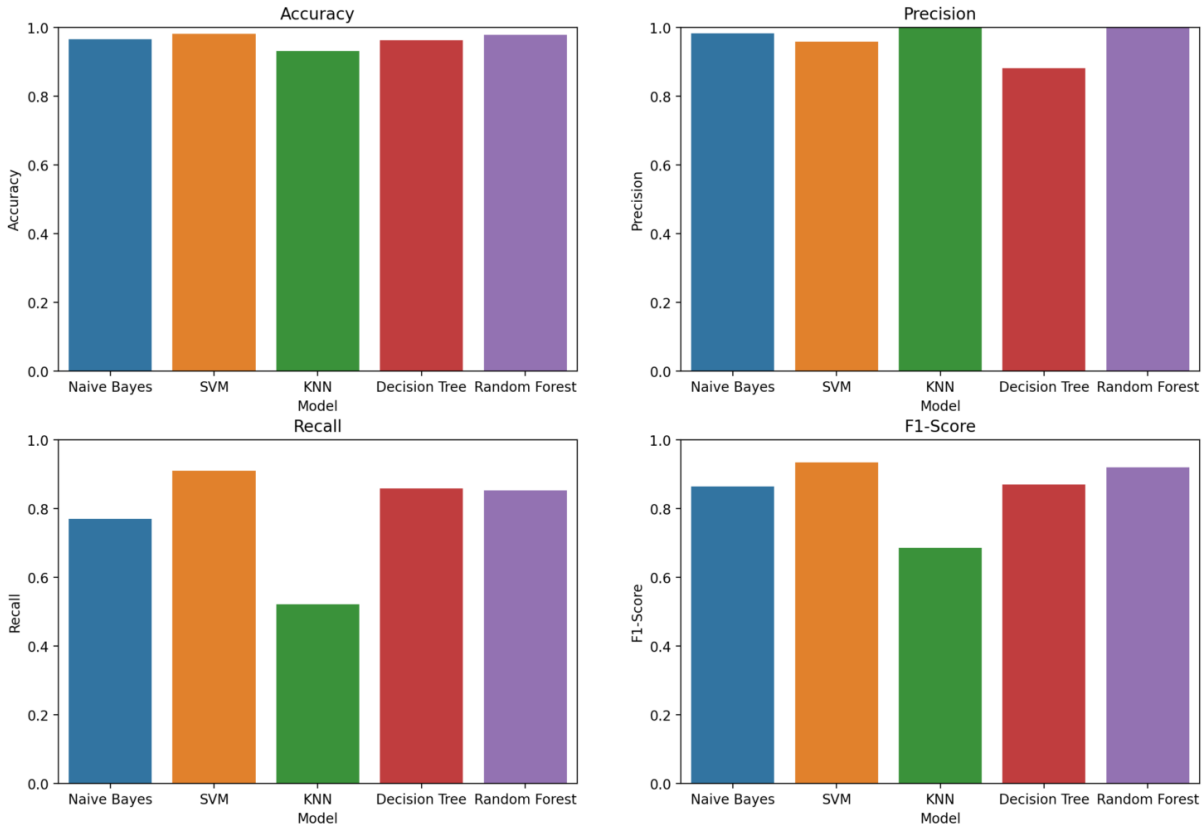

Abbildung 2: Modell-Metrik-Barchart

Streamlit App

Unsere Streamlit App bietet eine interaktive Plattform zur visuellen Darstellung des Modells und seiner Vorhersagen. Benutzer können verschiedene Eingaben vornehmen und die Auswirkungen auf die Vorhersageergebnisse in Echtzeit beobachten. Diese App dient sowohl zur Demonstration als auch zur praktischen Anwendung des Modells.

Die App ist unter folgender URL erreichbar:

https://ml4b-emailclassifier.streamlit.app

Zusammenfassung und Ausblick

In diesem Projekt haben wir erfolgreich mehrere maschinelle Lernmodelle zur Spam-Erkennung trainiert und evaluiert. Unsere Untersuchung umfasste die Modelle Naive Bayes, Support Vector Machines (SVM), k-Nearest Neighbors (KNN), Decision Tree und Random Forest. Nach der Implementierung und Bewertung der Modelle auf einem Datensatz von E-Mails konnten wir feststellen, dass Support Vector Machines (SVM) die höchste Genauigkeit und die besten Werte für Präzision, Recall und F1-Score aufwies.

Die Streamlit-App bietet eine benutzerfreundliche Oberfläche, um die Leistung der verschiedenen Modelle zu vergleichen und Vorhersagen für neue E-Mails in Echtzeit zu erhalten. Dies ermöglicht Anwendern, die Stärken und Schwächen jedes Modells zu verstehen und zu sehen, wie gut sie in der Praxis funktionieren.

Obwohl unsere Modelle bereits gute Ergebnisse erzielen, gibt es mehrere Möglichkeiten zur weiteren Verbesserung und Erweiterung dieses Projekts:

1. **Erweiterung der Datensätze**:
    - Die Verwendung größerer und vielfältigerer Datensätze könnte die Generalisierungsfähigkeit der Modelle verbessern.
    - Einbeziehung von E-Mails in verschiedenen Sprachen und aus unterschiedlichen Quellen, um die Robustheit der Modelle zu erhöhen.
2. **Fortschrittliche Feature-Engineering-Techniken**:
    - Die Implementierung von Techniken wie TF-IDF (Term Frequency-Inverse Document Frequency) oder Word Embeddings (z.B. Word2Vec, GloVe) könnte die Textrepräsentation verbessern und zu besseren Modellleistungen führen.
    - Experimentieren mit syntaktischen und semantischen Analysen der E-Mails.
3. **Ensemble-Methoden**:
    - Die Kombination der Vorhersagen mehrerer Modelle (Ensemble-Methoden) könnte zu robusteren und genaueren Vorhersagen führen.

4. **Echtzeit-Implementierung**:
    - Die Integration des Modells in E-Mail-Systeme zur Echtzeit-Erkennung von Spam könnte die praktische Anwendung und den Nutzen des Projekts erhöhen.
    - Implementierung von automatisierten Feedback-Schleifen, um das Modell kontinuierlich mit neuen Daten zu aktualisieren und zu verbessern.

Durch die Umsetzung dieser Verbesserungen könnte die Spam-Erkennung noch effektiver und vielseitiger werden, was sowohl die Sicherheit als auch die Effizienz in der E-Mail-Kommunikation weiter steigern würde.

Quellen

Machine Learning Techniques for Spam Detection in Email and IoT Platforms: Analysis and Research Challenges (wiley.com)

Email Spam Detection using Deep Learning Approach | IEEE Conference Publication | IEEE Xplore

A Study of Machine Learning Algorithms on Email Spam Classification - ProQuest

Review spam detection | Proceedings of the 16th international conference on World Wide Web (acm.org)

Neuer Tab (splunk.com)

Was ist Prädiktive Analytik? | Jaspersoft

Email Spam Detection Using Machine Learning Algorithms | IEEE Conference Publication | IEEE Xplore

ShiraniMehr-SMSSpamDetectionUsingMachineLearningApproach.pdf (stanford.edu)

Radware Bot Manager Captcha (perfdrive.com)

Maschinelles Lernen: Grundlagen & Anwendungen | StudySmarter

Convolutional neural networks for image spam detection: Information Security Journal: A Global Perspective: Vol 29, No 3 (tandfonline.com)

Chapter 3 Email Spam Filtering – ScienceDirect