Fragenextraktion und Absichtsklassifizierung anhand von COVID-19 Twitter-Daten

Tuesday 25th March, 2025 - 09:44

Ivaylo Krumov
University of Luxembourg
Email: ivaylo.krumov.001@student.uni.lu

Dieser Bericht wurde unter der Aufsicht von::
Sviatlana Höhn
University of Luxembourg
Email: sviatlana.hoehn@uni.lu

1. Einleitung

Dieses Projekt zielt darauf ab, die potenzielle Machbarkeit eines autonomen algorithmischen Ansatzes zur Verarbeitung und Klassifizierung von Daten zu analysieren. Das Projekt nutzt den Bereich der Absichtsklassifikation zur Durchführung dieser Studie.

In diesem Papier wird eine eingehende Antwort auf die Frage gegeben, wie ungeordnete Rohdaten mit Hilfe von AI richtig in Gruppen sortiert werden können. Um dies zu zeigen, wird ein Programm beschrieben, das die Absichtsklassifikation nutzt, um einen großen Datensatz zu klassifizieren. Darüber hinaus liefert ein wissenschaftlicher Text den Kontext zum Thema des Programms und zu diesem Projekt im Allgemeinen. Es wird erwartet, dass der Bericht mit den gegebenen Informationen einen nützlichen Einblick in die Machbarkeit der algorithmusbasierten Klassifizierung von Daten als Ganzes geben kann.

2. Projektbeschreibung und Voraussetzungen

Das Projekt besteht aus einem wissenschaftlichen und einem technischen Arbeitsergebnis. Der wissenschaftliche Teil ist ein Text, der eine gründliche Antwort auf die Frage "Kann ein Programm von Menschen geschriebene Fragen verarbeiten, um deren Absicht zu erkennen?" gibt. Um dies zu erreichen, untersucht der Text explizit einen Hauptbereich - die Absichtsklassifikation - während implizit auch die Bereiche der natürlichen Sprachverarbeitung und des natürlichen Sprachverständnisses behandelt werden, und beantwortet methodisch die wissenschaftliche Frage.

Das technische Arbeitsergebnis ist ein Programm zur Identifizierung von Fragen aus einer großen Menge von Textdaten und zur Klassifizierung dieser Fragen nach ihrer Absicht. Es wurde in der Programmiersprache Python geschrieben und nutzt die Python-Pakete Spacy, Tweepy, Gensim und Rasa, während Google Colab als Programmierumgebung verwendet wurde.

Die wichtigste Voraussetzung für die wissenschaftliche Arbeit ist die Fähigkeit, wissenschaftliche Texte kompetent zu schreiben. Die wichtigsten technischen Voraussetzungen sind allgemeine Erfahrung mit der Programmierung in mehreren Programmiersprachen und angemessene Kenntnisse der Programmierung in Python.

3. Wissenschaftliches Arbeitsergebnis - Kann ein Programm von Menschen geschriebene Fragen verarbeiten, um deren Absicht zu erkennen?

3.1. Anforderungen und Gestaltung

Die wichtigsten funktionalen Anforderungen an das Arbeitsergebnis sind die methodische Beantwortung der wissenschaftlichen Frage durch die Analyse der Schlüsselbegriffe und deren korrekte Definition zum besseren Verständnis des Themas, die Verwendung anderer wissenschaftlicher Quellen zur Unterstützung bestimmter Ideen des Arbeitsergebnisses und schließlich die Kombination aller gewonnenen Informationen zur Formulierung einer schlüssigen Antwort auf die wissenschaftliche Frage. Was die nicht-funktionalen Anforderungen anbelangt, so wird erwartet, dass das Arbeitsergebnis leicht verständlich ist, einen guten logischen Fluss aufweist und referenziertes Material ordnungsgemäß anführt.

Das wissenschaftliche Arbeitsergebnis ist als Text gestaltet, der den Kontext zu den wissenschaftlichen Ideen hinter dem erstellten technischen Arbeitsergebnis liefern soll. Er stützt sich auf glaubwürdige wissenschaftliche Arbeiten mit nützlichen Informationen, um seine Ideen zu untermauern. Das resultierende Arbeitsergebnis ist logischerweise in zwei Teile gegliedert, wobei der erste Teil die Schlüsselbegriffe der wissenschaftlichen Frage analysiert und der zweite Teil die endgültige Antwort auf die Frage enthält.

3.2. Produktion und Bewertung

Das wissenschaftliche Arbeitsergebnis definiert die drei identifizierten Schlüsselbegriffe nacheinander und hält sich dabei an die definierten Bereiche. Diese Unterfragen sind: ”Programm”, ”Frage” und ”Absicht”. Wo es angebracht ist, werden die relevanten Informationen aus den Papieren vollständig genutzt, um einen bestimmten Begriff zu definieren. Im letzten Teil des Textes wird die Schlussfolgerung gezogen, dass ein Programm sehr wohl in der Lage ist, Fragen als Input zu nehmen und sie nach ihrer Intention zu klassifizieren, auch wenn es dies vielleicht nicht vollständig selbst tun kann.

Insgesamt bereitet das wissenschaftliche Arbeitsergebnis den kommenden technischen Teil vor, indem es die definierten Anforderungen erfüllt und die erwarteten Ergebnisse liefert, so dass es seine Ziele erfolgreich erreicht hat.

4. Technisches Arbeitsergebnis - Simulationsprogramm zum Passwortknacken

4.1. Anforderungen und Gestaltung

Auch das technische Arbeitsergebnis sollte bestimmte funktionale und nicht-funktionale Anforderungen erfüllen. Zu den funktionalen Anforderungen gehören das Auffinden von Tweets nach Tweet-ID, die Extraktion von Fragen innerhalb eines Tweets und die Fähigkeit des Programms, die Absicht einer Frage zu bestimmen. Nicht-funktionale Anforderungen sind eine gute visuelle Darstellung wichtiger Informationen und die Verwendung von Dateien zum Lesen oder Speichern von Daten.

Das Arbeitsergebnis nutzt eine tabellarische Darstellung der Daten, wann immer sie benötigt wird. Es wurde mit Hilfe mehrerer Python-Pakete erstellt und ist von mehreren externen Dateien abhängig, um wie vorgesehen zu funktionieren. Diese Dateien werden praktischerweise in einem gemeinsamen Verzeichnis gespeichert, um eine reibungslose Ausführung des Programms zu gewährleisten.

4.2. Produktion und Bewertung

Das erstellte Programm besteht aus mehreren Algorithmen, die nach und nach zur Erfüllung der endgültigen Aufgabe beitragen. Zunächst werden die ID-Rohdaten gelesen und für jede ID wird der entsprechende Tweet extrahiert. Dann identifiziert und extrahiert ein weiterer Algorithmus jede Frage aus dem Tweet-Datensatz. Danach beginnt ein zyklischer Prozess, bei dem die Intents im aktuellen Fragen-Datensatz identifiziert werden und die Trainingsdaten für den Intent-Klassifikator entsprechend aktualisiert werden. Danach verarbeitet der Klassifizierungsalgorithmus jede Frage und die Ergebnisse der Absichtsklassifizierung werden angezeigt. Schließlich werden mit Hilfe eines empirischen Verfahrens die als korrekt klassifizierten Fragen aus dem aktuellen Datensatz entfernt und der zyklische Prozess beginnt von neuem. Dieser Vorgang wird so lange wiederholt, bis alle verbleibenden Fragen richtig klassifiziert sind.

Insgesamt erfüllt das Programm alle definierten Anforderungen. Trotzdem gibt es noch einige kleine Probleme, wie zum Beispiel das Fehlen eines Kontexts bei einigen Tweets, um sie wie beabsichtigt zu klassifizieren, und das Fehlen einer Anzeige des Endergebnisses.

5. Schlussfolgerung

Im Großen und Ganzen hat das Projekt die erwarteten Ergebnisse erbracht. Trotz der kleinen technischen Problemen, die noch bestehen, wird diese Arbeit als erfolgleich angesehen, da sie ziegt, dass ein Programm in der Lage ist, unstrukturierte Eingaben zu verarbeiten und daraus logisch organisierte Ausgaben zu erzeugen.