Protokoll 2 für die Sitzung vom 19.06.2015

Ivana Daskalovska 11139620

Thema: Machine-Learning basierte automatische OCR-Korrektur
Student: Michael Strohmayer
Betreuer: Klaus Schulz

Am Anfang der Präsentation hat Herr Strohmayer den Titel seiner Bachelorarbeit und die Gliederung seiner Arbeit vorgestellt. Danach ist er auf die Motivation eingegangen. Da Schriftbilder, Grammatik und Schreibweisen sich im Laufe der Zeit verändern, erkennen OCR Systeme manche Wörter nicht zuverlässig und werden dann falsch klassifiziert. Wenn so einen Fall auftritt, liefern OCR-Systeme eine Liste an Korrekturvorschlägen von denen der korrekte Vorschlag ausgewählt werden muss.

Ziel der Bachelorarbeit: Entwicklung einer Software zur automatischen Nachkorrektur der eingelesenen OCR-Dokumente, die einen Machine-Learning Klassifikator trainiert und die Ergebnisse auswertet.

Vorgehensweise: Herr Strohmayer hat erstmal die vorhandenen Dokumente eingelesen. Insgesamt waren es zwei Dokumente. Er hat die vom Profiler zurückgegebenen Featurewerte extrahiert. Darüberhinaus hat er auch neue Features hinzugefügt. Er hat zwei verschiedene Machine-Learning Klassifikatoren trainiert und anschließend eine Evaluation durchgeführt.
Der Profiler berechnet historische Schreibvarianten und gibt Korrekturvorschläge für Schreibfehler zurück.
Bei den Dokumenten, die er zur Verfügung hatte, handelt es sich um zwei Kräutertexte: „Paradiesgärtlein“ und „Curiöser Botanicus“. Diese stammen aus dem RIDGES Korpus, der von CIS in Kooperation mit Humboldt Universität in Berlin erstellt wurde. Der Korpus enthält 33 Kräuterkundetexte aus der Zeit zwischen 1484 und 1914. Der Korpus wurde manuell nachkorrigiert. Als eigene zusätzliche Features hat Herr Strohmayer die Längendifferenz zwischen dem OCR Token und dem Korrekturvorschlag implementiert. Darüberhinaus hat Herr Strohmayer auch den Konfidenzwert des Korrekturvorschlags zu Unterstützung verwendet.

Bei dem Training von Machine-Learning Klassifikatoren hat Herr Strohmayer die Scikit-learn (Umfangreiche Bibliothek an Machine-Learning und Data Mining Tools) verwendet um einen Gauß‘schen Naive Bayes Klassifikator zu trainieren. Der zweite Klassifikator baut auf der Libsvm auf, und verwendet eine Support-Vector-Machine zur Klassifizierung von Daten.

Probleme, die im Laufe der Bachelorarbeit aufgetreten sind: Im Anfangsstadium hatte Herr Stromayer enorme Performance Probleme bei der Datenverarbeitung, da die Daten zu groß waren. Er hat hierzu die interne Datenstruktur geändert, indem er den Python Dictonary Datentyp benutzt. Desweiteren hat er eine eigene Klasse für jeden Datensatz erstellt. Ein weiteres Problem waren die Konfidenzwerte in der Ausgabe, die teilweise sehr gering waren und somit falsche Trainingswerte geliefert haben. Diese wurden dann abgeschnitten.

Evaluation: Mit Kreuzevaluierung wurden sehr gute Ergebnisse erzielt. Er hat zum Training dabei 50% Daten von einem Korpus und 50% Daten von dem anderen Korpus genommen und auf dem Rest evaluiert. Er hat die beiden Klassifikatoren miteinander verglichen. Die Klassifikation mittels Naive Bayes lief sehr schnell und deutlich schneller als mit einer Support-Vektor-Maschine. Die Support-Vektor-Maschine lieferte aber die besseren Ergebnisse. Die Implementation der Support-Vektor-Maschine in scikit-learn unterstützt ein Multiklassen- bzw. Einklassen-Modell, Dabei hat das Multiklassen-Modell die besseren Ergebnisse geliefert, obwohl die Klassifikationaufgabe - Korrekturvorschlag entweder korrekt oder nicht korrekt ist - eigentlich binär.

Er hat als Metriken Precision, Accuracy, Recall und F1 verwendet.
Diese Metriken wurde zunächst auf die Basisfeatures, dann Basisfeatures mit Konfidenzwert und zum Schluss auf alle Features angewendet.

Häufige Fehlklassifikationen: der Profiler hat ab und zu einen falschen Konfidenzwert geliefert oder keinen Korrekturvorschlag zurückgeliefert, da keine Informationen in den GrundTruth-Daten vorhanden waren.

Die Automatische Nachkorrektur von OCR Dokumenten hat sich als sinnvoll erwiesen und lieferte gute Ergebnisse. Weitere Schritte wären die Kombination von beiden Klassifikatoren und die Hinzunahme von weiteren Features.