Christoph
Flemmig
CIS, LMU München

25.05.2017

Protokoll für das Kolloquium vom 22.5.2017

In dieser Sitzung wurde drei Bachelorarbeiten von den Studenten in Form von kurzen Vorträgen mithilfe einer kleinen Präsentation am Beamer vorgestellt.

1. Vortrag:
Faridis Alberteris Azar, BA-Betreuer: Dr. Maximilian Hadersbeck
Thema: Optimierung der linguistischen Suche beim XML-annotierten Nachlass von Ludwig Wittgenstein

Zu Beginn stellte Faridis die Umstände und Beweggründe für ihre Bachelorarbeit vor. Sie erzählte von der WiTTFind App die am CIS in Zusammenarbeit mit der Universität Bergen entwickelt wurde. Dies ist eine Suchmaschine für den digitalisierten Wittgenstein Nachlass. Faridis erklärte dass sich ihre Arbeit vor allem auf die Optimierung der linguistischen Suche beim XML-annotierten Nachlass von Ludwig Wittgenstein bezieht. Sie stellte außerdem die drei unterschiedlichen Arten von Dokumenten des Wittgenstein Nachlasses vor. Die erste Art dieser Dokumente ist in der Form wie Wittgenstein sie aufgeschrieben hat, die zweite ist in der sogenannten DIPLO-Form und enthält nur die verbesserte Form und das dritte ist die normalisierte Form mit welcher Faridis arbeitet. Des weiteren erwähnte sie die verwendeten Tagger die sie zur Verbesserung der Suchmaschine benutzt. Bei einem handelt es sich um einen probabilistischen POS-Tagger und bei dem anderen um einen von Helmut Schmid entwickelten Tree-Tagger.
Als Hauptproblematik ihrer Arbeit beschrieb Faridis die Erkennung von den unterschiedlichen Formen von Eigennamen, vor allem wenn diese als Adjektiv verwendet werden. Für diese Problematik hat die Universität Bergen ein XML-Element für Eigennamen hinzugefügt, jedoch verwendet der Tagger dieses Element nicht.
Für den ersten Schritt der Lösung des Problems hat sie deshalb mithilfe der Python-Schnittstelle The ElementTree XML API die Fehler beim Tagging gesammelt. Die falsch getaggten Eigennamen die bei dieser Suche herauskamen speicherte sie dafür in einer Liste, einmal mit dem falsch annotierten Form des Namens als Token und die Grundform des Namens wurde als Lemma übergeben.
Im zweiten Schritt versucht sie die semantische Suche in WiTTFind zu verbessern. Auch die neuen Suchmuster führen immer noch zu Fehlerkennungen. So werden zum Beispiel Begriffe wie hellenisch oder Venus nun als Eigennamen markiert. Ihr Lösungsansatz dafür ist eine neue syntaktische Kategorie 'persName' in WiTTFind zu erzeugen.
Mit diesem Ansatz erhielt sie überraschend positive Ergebnisse. So werden nun in 13 Dateien wo davor 168 Eigennamen entdeckt wurden nun 833 entdeckt.
Als weitere mögliche Kapitel für ihre Arbeit nannte sie Beispielsweise noch die Verbesserung des Umgangs der Transkriptionsfehler und Editionsprobleme bei XML-Dateien im Wittgenstein Nachlass und eine Erweiterung des Lexikons unter Verwendung eines Tree-Taggers.

2. Vortrag:
Iuliia Khobotova, BA-Betreuer: Wenpeng Yin
Thema: Comparing representation learning over word-level, character-level and combination of both in NLP tasks

Iuliia stellte zuerst die Fragestellung und Aufgaben ihrer Bachelorarbeit vor. In ihrer Arbeit geht es

um den Einfluss unterschiedlicher Arten von Input auf die Accuracy auf Convolutional Neural Networks. Ihre Aufgabe ist es festzustellen was die besten Sets von Parametern sind sowie herauszufinden wie schnell die Accuracy berechnet wird und wie sich die Kombination von Word- und Character-Embeddings darauf auswirken.
Für diese Aufgabenstellung überprüft sie deren Ergebnisse in unterschiedlichen Aufgaben des Natural Language Processings (NLP).
Als nächstes gab Iuliia einen kurzen Überblick über Convolutional Neural Networks und Recurrent Neural Networks. Diese sind zwei Hauptarten von Neural Networks die für viele unterschiedliche NLP-Tasks genutzt werden.
In der Vorstellung ihres Experiments wurden drei zu verändernde Parameter genannt, embedding size, hidden size und batch size. Ihr Ziel ist es dabei das Set von Parametern zu finden das die höchste Accuracy ergibt.
Sie Arbeitet dabei mit dem „Stanford Sentiment Treebank"-Datenset das aus annotierten Filmkritiken besteht. Es enthält ungefähr 215000 einzigartige Sätze. Sie benutzt das Python Framework Theano.
Ihre Evaluation wird dabei statistisch basierend auf dem Vergleich der Accuracy sein welche sie graphisch ausarbeiten will.

3. Vortrag:
Alexander Vordermeier, BA-Betreuer: Katharina Kann
Thema: Comparison of Transfer Methods for low Ressource Morphology

Alexander begann seinen Vortrag mit der Motivation für seine Arbeit. In der Arbeit geht es um die Paradigmen Komplettierung von Sprachen. Hier ergibt sich eine Problematik für Low Ressource Sprachen da nicht genug Material für die Paradigmen Komplettierung vorhanden ist. Seine Arbeit soll nun versuchen herauszufinden ob man wenn man eine High Ressource Sprache die der Low Ressource Sprache sehr ähnlich ist zur Hilfe nimmt bessere Ergebnisse erhält. Als High Ressource Sprache hat er hierfür Bulgarisch gewählt, als Low Ressource Sprache Mazedonisch.
Dafür verwendet er die drei Methoden Sprachübergreifende Paradigmen Komplettierung, Auto Encoding und eine Kombination aus beiden.
Daraufhin ging Alexander nun näher auf die unterschiedlichen Methoden ein. Als erstes stellte er Sprachübergreifende Paradigmen Komplettierung genauer vor. Hierfür benötigt man wie schon erwähnt zu einer Low Ressource Sprache noch eine möglichst ähnliche High Ressource Sprache. Die annotierten Daten der beiden Sprache werden werden dabei in unterschiedlich großen Paaren von Datenpaketen vermischt wobei die Datenpakete der Low Ressource Sprache kleiner sind. Darauf wird dann ein Modell trainiert.
Die zweite Methode auf die Alexander näher einging war das Auto Encoding. Hierbei handelt es sich um eine Methode bei der die Eingabe auch gleichzeitig die Ausgabe ist, die Hoffnung ist dass viele Flexionen der Wörter gleich sind. Dies wäre jedoch laut Alexander selten der Fall.
Als dritte Methode wurde die Kombination aus den beiden vorherigen Methoden vorgestellt. Hierbei wird das Modell wieder auf vermischte Datensätze trainiert, diesmal jedoch bestehend aus drei unterschiedlichen Datensätzen anstelle von zwei. Es werden zusätzlich zu den annotierten Daten auch noch nicht annotierte Daten aus der Low Ressource Sprache verwendet.
Im Anschluss an die Methoden stellte Alexander die nötige Form der Daten vor die das Modell benötigt. Dies sind Test-, Development- und Trainingsset die hier als „Source"-Dateien übergeben werden und „Target" Dateien in welchen die Lösung steht. Alexander erklärte außerdem genauer welche Informationen in Source übergeben werden, nämlich die Sprache, die Wortart, ein Tag und das Lemma.
Danach wurde einige Ergebnisse präsentiert. So ließ sich erkennen dass Auto Encoding deutlich am schlechtesten der drei Methoden abschnitt was auch den Erwartungen entsprach. An zweiter Stelle stand die Paradigmen Komplettierung, die immerhin eine Accuracy von fast 50% erzielte. Das beste Ergebnis erzielte die Kombination der beiden Methoden wobei man sagen muss dass dieses nur

geringfügig besser war. Mit größeren Datenpaketen erzielten jedoch alle Methoden ein höheres Ergebnis wobei sich an der Rangfolge nichts änderte.
Am Ende seiner Präsentation gab Alexander noch eine kurze Fehleranalyse. Grund für Fehler wären zum Beispiel falsch verwendete Endungen. Des weiteren treten viele Fehler beim Auto Encoding aufgrund der Vorgehensweise auf. Schlussendlich erwähnte er noch das es eine schwierige Aufgabe für Leute die keine Muttersprachler sind ist. Als weitere Aufgaben für seine Arbeit erwähnte er noch die Identifikation von Fehlerquellen, besseres Verständnis des Modells und die Betrachtung anderer gängiger Verfahren.