Protokoll zur Sitzung vom 22.05.2017

Faridis Alberteris Azar, Betreuer: Dr. Maximilian Hadersbeck

"Optimierung der linguistischen Suche beim XML-annotierten Nachlass von Ludwig Wittgenstein"

Der Vortrag beginnt mit der Einordnung ins Digital-Humanities-Projekt "Wittgenstein in Co-Text" mit der Partneruniversität Bergen (WAB) in Norwegen. Die Uni Bergen besitzt die Rechte zu 20.000 Seiten des Wittgenstein-Nachlasses, von denen sie 5000 der LMU zur Verfügung gestellt haben. Dabei handelt es sich um XML-Dateien in drei Kategorien. Faridis erklärt die verschiedenen Kategorien anhand eines Beispiels: Wittgenstein fügte an einer Textstelle, bei der es um Schmerzen ging, den Zusatz "Zahn" nachträglich hinzu. Die NORM Kategorie enthält also die angepeilte Endfassung des Textes, hier: "Zahn s chmerzen". Die DIPLO Kategorie beschreibt den tatsächlichen Stand des Textes, hier: "Zahn S chmerzen". Bei der ORG Kategorie liegen alle Varianten vor: "Zahn s S chmerzen". Danach stellt sie den verwendeten Tagger vor, den POS TreeTagger (Dr. Schmid). Wichtig für die Arbeit ist dabei, dass der Entscheidungsbaum die Übergangswahrscheinlichkeiten misst. Der Schwerpunkt der Arbeit aber umfasst die Erkennung von Personennamen in den Texten des Nachlasses und eine entsprechende Erweiterung des Lexikon. Alte NORM-Dateien der Universität Bergen enthalten kein spezielles XML-Element für Eigennamen. Der Tagger versucht nun eine Namensform einem Namenslemma zuzuordnen. Es gibt jedoch Fehler: diese Namensformen sind nicht in Trainingsdaten enthalten und können somit nicht erkannt werden. Im März 2017 erreichen aktuelle Daten das CIS. Die Universität Bergen hat daran gearbeitet, neue XML-Elemente ("persName") in die Dateien einzuarbeiten, die einen Personennamen beschreiben. Hier ist das Lemma des Namen als Key aufgeführt. Nach weiteren Experimenten und Training steht fest, dass der Tagger diese neue XML-Information nicht nutzt. Nun bestand die Aufgabe darin, den Fehler zu finden und den Python Parser etree auf den Task anzusetzen. Alle Namen die in NORM-Dateien gefunden wurden, werden einer Liste von Token-Lemma hinzugefügt. Nun wagt man sich an eine semantische Suche mit WittFind heran und versucht Eigennamen mithilfe von Tags zu finden. Hier werden jedoch ebenfalls noch teilweise falsche Treffer geliefert: es wird ein Eigenname gefunden, dieser gehört jedoch nicht zu einer Person, z.B. wird "Venus" als Treffer erkannt, dabei handelt es sich im Kontext um den Planeten. Um das zu umgehen, wird ein neuer Vorschlag formuliert. Die neue Kategorie <+persName> soll nun auch Tags enthalten und wird im CIS-Lexikon eingetragen. Die daraus resultierenden Ergebnisse werden anhand eines Vergleichs mit WittFind vorgestellt. WittFind findet in 13 von 20 Dateien 168 richtige Treffer. Das neue System dagegen findet in allen Dateien 833 richtige Treffer. Dr. Schulz merkt an, dass er erwartet hätte, dass die Treffer nicht mehr sondern präziser werden würden. Weitere Arbeit an dem Projekt umfasst Transkriptionsfehler (falsche XML-Elemente) und Editionsprobleme (Informationen, die beim Editieren in Bergen

verblendet wurden) so gut wie möglich zu beheben und das Lexikon entsprechend erweitern, z.B. indem man die Wortliste bzw. Frequenzliste mit etree statt regex zu erstellt.

Iuliia Khobotova, Betreuer: Wenpeng Yin

"Comparing representation learning over character-level, word-level or a combination of both in NLP tasks"

Iuliia beschäftigt sich in ihrer Arbeit mit der Frage, wie verschiedene Input-Parameter die Genauigkeit von CNNs (convolutional neural network, ein neuronales Netzwerk, das hierarchisch aufgebaut ist) und RNNs (recurrent neural network, ein sequentielles neuronales Netzwerk) beeinflussen. Daraus resultierend versucht sie das beste Parameter-Set und die schnellste Berechnung von Genauigkeit zu bestimmen. All das wendet sie im Hinblick auf Wort- und Zeichen-Embeddings an. Dazu führt sie Experimente mit folgenden, wechselnden Parametereigenschaften durch: der Größe des Embeddings, der Größe der Hidden-Layer und der Größe des Batchs. Der hierfür verwendete Korpus stammt aus dem Projekt Stanford Sentiment Treebank und enthält annotierte Daten aus Kinofilmrezensionen. Dort sind über 215.000 einzigartige Sätze zu finden. Die Open Source Software, die Iuliia verwendet, nennt sich Theano und wird von einer Machine Learning Gruppe in Montreal entwickelt. Es handelt sich dabei um ein Python Framework, das mit numpy-ähnlichen Strukturen rechnet. Input und Output Layer des Systems unterscheiden sich nicht, während das Hidden Layer verschiedene neuronale Netze darstellt. Iuliia beendet ihren Vortrag mit einem Ausblick auf noch ausstehende Aufgaben. Es gibt noch Statistiken, die die verschiedenen Genauigkeiten beschreiben, auszuwerten und diese dann entsprechend graphisch darszustellen.

Alexander Vordermaier, Betreuer: Katharina Kann

"Comparison of Transfer Methods for Low-Resource Morphology"

Nach einem kurzen Überblick, beginnt Alexander seinen Vortrag mit der Motivation für seine Arbeit. Es handelt sich um die Frage, inwiefern man Sprachen mit wenig annotierten Trainingsdaten mit Daten aus anderen aber ähnlichen Sprachen oder aber derselben Sprache ohne Annotation unterstützen und damit den Mangel kompensieren kann. Für seine Arbeit hat er Bulgarisch als High Resource Sprache (mit ausreichend annotierten Daten) und Mazedonisch als Low-Resource Sprache (mit zu wenigen Daten) gewählt. Das Ganze wird anhand von Morphologie getestet. Dr. Schulz wirft hier ein, dass sich bei vielen Ressourcen alle morphologischen Formen für häufige Wörter einfach finden lassen, seltene

Wörter dabei aber ebenso auf der Strecke bleiben. Bei der ersten Methode, der Paradigma-Komplettierung, handelt es sich um die Mischung der Low Resource Trainingsdaten mit den High Resource Trainingsdaten einer anderen Sprache. Hier werden die Sprachdaten für das Trainingsset in verschiedenen Kombinationen (50, 200 für LR und 50, 100, 200, 400, 800, 1600, 3200, 6400, 12800 für HR) jeweils gemischt. Die Daten sind zudem in zwei Inhalte aufgeteilt. Eine File enthält das Lemma eines Wortes und die Tags der gesuchten morphosyntaktischen Form des Wortes, die andere File enthält das gesuchte Wort mit der in den Tags beschriebenen Form. Bei dem Training liest das System also die Zeile und lernt das Lemma und die Tags und bekommt gleichzeitig die Zeile mit dem flektierten Wort vorgelegt. Nach dem Training wird es beim Testen versuchen, die Lemma-Form-Paare der Testdaten richtig zuzuordnen. Die zweite Methode (Auto-Encoding) macht sich nicht-annotierte Daten derselben Sprache zunutze. Hierbei ist das Wort sowohl Input als auch Output und wird nur stumpf kopiert. Auch das erfolgt in den oben genannten Kombinationen. Hier wird aus dem Publikum die Frage laut, was ein solcher Trainingssatz für Erfolge erzielen soll, wenn Input und Output identisch sind, Alex anwortet darauf, dass dies ein reines Experiment war, um zu sehen, ob es denn irgendetwas an der Genauigkeit verändert und dass dann an den Resultaten, die er in kürze vorstellen wird, sichtbar wird. Die dritte und letzte Methode, um ein Resultat bei Low Resource Sprachen zu erzielen, war im Rahmen dieser Arbeit die Kombination der beiden ersten Methoden. Auch hier kommen die bereits genannten Kombinationen ins Spiel, jedoch wird das Verhältnis von High Resource Daten beibehalten (also beispielsweise mit 50 LR Daten sieht das Paket nun so aus: 50 – 200 –200 und nicht 50 – 200). Zur Veranschaulichung der Ergebnisse werden zwei Graphen angeführt, einer der die Experimente mit 50 LR Datensätze zeigt und einer der 200LR Datensätze darstellt. Bei dem 50er Paket wird deutlich, dass die zweite Methode, das Auto-Encoding am schlechtesten abschneidet und nie an 20% Trefferquote herankommt. Das liegt wohl an der Methode selbst, denn schließlich scheint ein bloßes Kopieren des flektierten Wortes nicht auf das Lemma schließen. Die bulgarischen und mazedonischen Wortendungen unterscheiden sich sehr häufig vom Lemma, anders als beispielsweise im Deutschen (der Baum, des Baumes, dem Baum, den Baum). Die erste Methode mit der fremden Sprache als Hilfsmittel erreicht beinahe 50% und die Kombination aus beiden erreicht die 50%-Marke. Bei den 200er Datensätze sehen die Ergebnisse schon vielversprechender aus. Die schlechteste Methode vorhin erreicht nun fast 60%, ein erheblicher Anstieg. Die Kombination erreicht fast 80%, während sich die Paradigma-Komplettierung an diese annähert. Dr. Schulz vermutet, dass die Qualität der Ergebnisse von den gewählten 50 bzw. 200 Datensätzen abhängt. Die restliche Zeit seiner Arbeit wird Alexander damit verbringen, mögliche Fehlerquellen zu finden, sich mit ähnlichen Arbeiten zu befassen und sich genauer mit dem Modell auseinanderzusetzen, das seines Wissens aus drei rekursiven neuronalen Netzwerken (RNN) besteht.