Katja Bertholdt
CIS, LMU München
15.05.17

Protokoll zur Sitzung vom 15.05.17 – Computerlinguistik Kolloquium

1. Vortrag: Tobias Eder, „Exploiting Bilingual Word Embeddings to Establish Translational Equivalence“

Den ersten Vortrag des Tages hält Tobias Eder. Er stellt das Thema seiner Bachelorarbeit vor, welche von Dr. Fraser und Dr. Braune betreut wird. Seine Arbeit befasst sich mit „Exploiting Bilingual Word Embeddings to Establish Translational Equivalence“. Die Motivation seiner Arbeit ist die Idee, Übersetzungen auch ohne vorhandenes Wörterbuch tätigen zu können. In bestimmten Domains, wie beispielsweise im medizinischen Bereich, liegen häufig keine Wörterbücher vor und es wäre wünschenswert, trotzdem möglichst genau übersetzen zu können.
Eder macht auf die generelle Schwierigkeit aufmerksam, auf die man stößt, sobald man sich mit Textdateien auseinander setzt. Bei Audio- und Imagedaten handelt es sich um sehr dichte Daten, wohingegen bei Texten das Problem besteht, dass Wörter als atomare Einheiten betrachtet werden. Daher bietet sich die Umwandlung in Word Embeddings und eine Repräsentation im Vektorraummodell an. Hierbei gibt es die Annahme einer Distributionshypothese, das heißt, dass durch die Abbildung eines Wortes im Vektorraum syntaktische oder semantische Beziehungen zu anderen Vektoren aufrecht erhalten bleiben und veranschaulicht werden können. Zudem wird angenommen, dass bei einem ähnlichen Kontext der Wörter eine syntaktische Abhängigkeit besteht. Diese Annahme wurde bereits mehrfach in der Forschung überprüft und dabei wurde festgestellt, dass gewisse Abhängigkeiten durch Richtungen im Vektorraum zu erkennen sind. Diese waren nicht nur syntaktischer, sondern auch semantischer Natur.
Da ein Vektorraum in der Regel sehr hochdimensional ist, muss er für die bildliche Darstellung heruntergerechnet werden. Durch eine Darstellung im zweidimensionalen Raum wird ein cluster verwandter Wörter erwartet. Liegen Wörter nah beieinander, dann können Zusammenhänge festgestellt werden und mit der Kosinusähnlichkeit berechnet werden, wie ähnlich sich diese Worte sind.
(An dieser Stelle wird aus dem Publikum angemerkt, dass mit einer Darstellung der Wörter in einem Vektorraum das Sparseness Problem nicht gelöst wird. Kommt ein Wort im Korpus nur einmal vor, ist es nicht repräsentativ, dieses im Vektorraum darzustellen. Allerdings gilt das Sparseness Problem als inhärent und es gibt bisher keinen Weg, dieses zu lösen.)
Eder stellt zwei Vektorraummodelle vor. Das erste heißt Word2Vec, welches 2013 im Rahmen von Google veröffentlicht wurde. Es beinhaltet zwei Modelle. Das erste Modell heißt CBOW Modell und versucht anhand eines gegebenen Kontexts das passende Wort vorherzusagen. Das zweite Modell heißt Skipgram Modell und versucht zu einem gegebenen Wort den Kontext vorherzusagen.
Das zweite Vektorraummodell heißt „fastText“. Es wurde von Facebook Research 2016 herausgegeben. Hierbei werden Wörter nicht als atomare Einheiten verstanden, sondern Wörter mit gleichem Stamm, also morphologischer Ähnlichkeit, werden im selben Raum verortet. Durch ein Word-Vectors OOV Wörterbuch können dann eventuell ähnliche Wörter gefunden werden, wenn das gewünschte Wort selbst nicht enthalten ist.
Es ist nun das Ziel, lineare Abbildungen zwischen unterschiedlichen Vektorräumen unterschiedlicher Sprachen zu finden. Das bedeutet, dass semantische Beziehungen erhalten bleiben bei der Abbildung von einer Sprache in eine Andere. Diese Lineare Abbildung wird mithilfe einer ridge Regression (L2-Regularisierung) gemacht. Das bedeutet, dass große Gewichte in der Matrix bestraft werden.
Zu den Korpora und dem Experimentaufbau stellt Eder kurz vier Korpora vor, die er verwendet. Der erste ist ein General Korpus, er besteht aus Wikipedia Artikeln und anderen Websiten und

umfasst 110 Millionen Token und 300.000 Types. Der zweite Korpus heißt Medical Big und beinhaltet Texte aus der Medizin. Der dritte Korpus heißt EMEA und ist ein pharmazeutischer Korpus, der unter anderem viele chemische Formeln enthält. Der letzte Korpus beinhaltet Transkripte gesprochener Sprache und heißt TED Talks. Eder beschäftigt sich mit einer Englisch-Deutsch Übersetzung. Pro Korpus wurden dazu ein kleiner paralleler Teil mit ca. 5.000 Wörtern mithilfe des moses toolkit übersetzt. Hierbei wurden bisher im General Korpus eine 95%ige Korrektheit erzielt, beim EMEA Korpus 78%.
Das Experiment besteht darin, ca. 1.000 Wörter aus einem Korpus auszusuchen, die nicht im Parallelkorpus auftauchen. Dann soll mit Hilfe einer Abbildung mit dem Regressionsmodell geprüft werden, wie gut die Übersetzung von einem Vektorraum in den anderen ist. Bei diesen Domänenspezifischen Testsets wird dann manuell geprüft, ob die Übersetzung passt. Hierbei ist einer unterschiedliche Performance der Modelle zu erwarten. Das CBOW sollte auf den medizinischen Daten besser funktionieren, da in dieser Domäne in der Regel sehr viele verschiedene Types verwendet werden.
Zum Schluss seines Vortrags gibt Eder einen Ausblick auf seine nächsten Schritte. Er wird sich auch mit den niedrig frequentierten Wörtern beschäftigen und die Genauigkeit deren Übersetzung betrachten. Außerdem wird er sich Gedanken darüber machen, wie man die Abbildungen durch alternative Regularisierungsmethoden optimieren kann. Außerdem möchte er eine Evaluation auf OOV-Wörtern in fastText durchführen.

2. Vortrag: Joseph Birkner, „Ranking with neural Network derived Document Vectors“

Der zweite Vortrag wird von Joseph Birkner gehalten. Seine Arbeit schreibt er bei PMS, welches zum Institut für Informatik angehört. Dort wird er von M.Sc. Wang und Prof. Dr. Francois betreut. Seine Arbeit beschäftigt sich mit dem Thema „Ranking with neural Network derived Document Vecotrs“. Er hält seinen Vortrag auf Englisch. Der Name des Projekts, in dem Birkner seine Arbeit schreibt ist „IROM“ und bedeutet „Intelligent Recommendatition of massive Open online courses“. Die Aufgabe von Birkner ist es, eine Suchmaschine für Onlinekurse zu erstellen und deren Output für den Nutzer sinnvoll zu gestalten. Durch die Suchanfrage des Nutzers besteht eine „information-need“, die durch das Finden relevanter Kurse aus dem Datensatz (=Textkorpus) beantwortet wird. Der Textkorpus beinhaltet Kursbeschreibungen zu den Kursen.
Birkner zeigt eine Grafik zur Veranschaulichung der geplanten Suchmaschine. Darauf ist eine Person zu sehen, die eine query formuliert. Diese durchläuft dann einen Rankingalgorithmus bezüglich einer Datenbank und der Person werden letztendlich passende Ergebnisse ausgegeben. Dieser Rankingalgorithmus kann durch Meta-Daten, beispielsweise Geschlecht, Alter und vergangene Suchanfragen des Nutzers, verbessert werden, um genauere Ergebnisse zu erzielen.
Birkners Arbeit gehört einem sehr neuen Forschungsbereich des Information Retrieval an. Dieser Bereich hat sich erst vor ca. drei Jahren entwickelt und heißt „Neural Information Retrieval“(NIR). NIR ist in mehrere Teilbereiche unterteilt. Der erste Teil heißt „Representation, Optimization“. Hierbei geht es darum, die Repräsentation der Suchanfrage (des Inputs) zu optimieren. Dieser Bereich wird auch „Deep semantic structured Matching“ genannt. Durch den zweiten Teil „Deep Relevants matching Models“ sollen bessere Rankingergebnisse durch die gegebenen Repräsentationen getroffen werden. Birkners Arbeit beschäftigt sich mit dem ersten Teil der NIR, also mit Representation and Optimization.
Birkner beschreibt die Motivation seiner Arbeit mithilfe eines Axioms: „We need efficient document representations on instantaneously rank recommended courses based on student need“. Eine gute Dokumentrepräsentation besteht darin, dass sie klein genug ist für den Algorithmus, aber dennoch effektiv genug ist, damit alle nötigen Informationen für den Algorithmus enthalten sind. Von einer Dokumentrepräsentation durch tf-idf Gewichtung wird hier abgesehen, da in diesem Modell die Wortreihenfolge nicht berücksichtigt wird und es außerdem davon ausgegangen wird, dass jeder Term des Dokuments eine eigene Bedeutung hat. Birkner unterstreicht an dieser Stelle

das Problem durch ein kleines Beispiel. Hierbei werden zwei Sätze aus völlig unterschiedlichen Kursbeschreibungen gezeigt, deren Vokabular sich aber stark überschneidet. Mithilfe der tf-idf Gewichtung erhalten sie die gleichen Werte, obwohl nicht beide Kurse thematisch zur Suchanfrage des Nutzers passen.
Eine Lösung für die Repräsentation von Wörtern bietet Word2Vec. Mit dieser Methode erhält man semantic-space für Wörter. Hier wird allerdings ein semantic-space für Dokumente benötigt.
Ein System, das Embedded Word Representations für Wörter generiert, wird durch ein „Autoencoder-Network“ trainiert. Birkner erklärt die Grundfunktion eines Autoencoders: dem Neural Network wird ein Kontext als Input übergeben. Der Encoder soll dann das Wort vorhersagen, welches in den Kontext gehört. Da zu jedem Wort nur der Kontext bekannt sein muss, kann die Technik auch auf „unlabeld data“ trainiert werden, was sehr hilfreich ist. In seiner Bachelorarbeit will Birkner einen Autoencoder für die Repräsentation von Dokumenten und nicht für Wörter einsetzen. Dafür werden die tf-idf Werte der Dokumente aus dem Korpus nebeneinander in einer Matrix gespeichert (jede Zeile repräsentiert einen Term, jede Spalte ein Dokument). An dieser Matrix kann die Gauß-Methode angewendet werden, um sie auf ihre hauptsächlichen Vektoren zu reduzieren. Auf diese Weise kann eine niedrigerdimensionale Repräsentation erzielt werden. Diese Vorgehensweise wird LSA (Latent Semantic Analyses) genannt. Das Problem hierbei ist, dass immer der gesamte Korpus verwendet werden muss. Ein Vorteli ist, dass im Vergleich zur tf-idf Gewichtung hier nicht die Annahme getroffen wird, dass jedes Wort eine eigenständige Bedeutung hat. Die Reihenfolge der Wörter wird aber weiterhin nicht beachtet.
Birkner möchte als Teil seiner Arbeit einen „sequence to sequence Autoencoder“ bilden. Dazu wird dem Neural Network kein fester Kontext als Input mit Hilfe eines übergegeben, sondern alle Terme des Dokuments nacheinander. Das Neural Network soll das Dokument zu der Hidden-Repräsentation vorhersagen, durch welches diese hergestellt wurde. Dieser Vorgang wird „Doc2Vec“ oder „sequence to sequence“ genannt.
Birkner gibt einen Ausblick auf seine nächsten Arbeitsschritte. Er hat bereits einen Prototyp entworfen und Dokumentvektoren erstellt. Diese produzierten Vektoren sind 30 dimensional. Der verwendete Korpus besteht aus Kursbeschreibungen. An dieser Stelle gibt Birkner einen Einblick in den Output seines Prototyps. Zu sehen ist ein zweidimensionaler Vektorraum, in dem viele verschiedene Dokumente durch farbliche Punkte verortet sind. Die Farben der Punkte entsprechen Labels zu unterschiedlichen Themenbereichen der Kurse. Birkner erhoffte sich, dass Punkte der gleichen Farbe möglichst nah beieinander verortet werden. Bisher ist aber nur ein Teilerfolg zu verzeichnen, denn es sind nur ein paar Cluster aufzufinden.
Seinen ersten geplanten Arbeitsschritt, nämlich das Training des Seq2Seq- Models hat er bereits abgeschlossen. Als nächsten Schritt wird Birkner durch einen Autoencoder Dokumentvektoren erzeugen. Danach wird ein Ranking erstellt, das ähnliche Dokumentvektoren ausgibt anhand eines gegebenen Dokumentvektors. Zu diesem Zweck wird die Kosinusähnlichkeit berechnet. Anschließend werden einige Evaluationsschritte folgen.

3. Vortrag: Kristina Smirnov, „Comparison of transfer methods for low-resource morphology“

Den letzten Vortrag des Tages hält Kristina Smirnov. Ihre Arbeit behandelt das Thema „Comparison of transfer methods for low-resource morphology“ und wird von Katharina Kann betreut. Zu Beginn stellt Smirnov die Gliederung ihrer Bachelorarbeit vor.
Das Thema der Bachelorarbeit ist im Rahmen der SIGMORPHON 2016 aufgekommen. Dort wurde ein Encoder-Decoder namens MED entwickelt. Die Aufgabe des MED ist es, aus einem Dataset Paare aus Lemma und Flektierter Wortform zu finden, um Rückschlüsse auf die Morphologie zu gewinnen. Es wurde festgestellt, dass bei einem Mangel an Trainingsdaten oder Daten einer bestimmten Sprache auch Daten einer anderen Sprache oder nicht annotierte Daten derselben Sprache verwendet werden können. Smirnov möchte in ihrer Arbeit herausfinden, ob diese beiden

Vorgehensweisen (das Verwenden von Daten einer anderen Sprache bzw. nicht annotierter Daten der gleichen Sprache) kombiniert werden können. Das Model MED wurde von Hinrich Schütze und Katharina Kann an der LMU entwickelt. Dieses Model verfügt über drei Tasks. Das erste ist die Voraussage der flektierten Form (=target), wenn ein Lemma und die flektierte Form gegeben sind (An dieser Stelle scheint es einen Fehler in der Präsentation zu geben. Es wird vermutet, dass nicht die flektierte Form gegeben ist, sondern eine Beschreibung dieser Form). Das zweite und dritte Task wird hier nicht näher beschrieben.
Smirnov wird mit diesem Model trainieren. Dazu sind drei Trainingseinheiten angedacht. Zunächst wird ein Set aus russisch-annotierte Daten mit annotierten ukrainischen Daten verwendet. Im zweiten Training wird ein Dataset aus annotiertem Russisch mit nicht-annotiertem Russisch eingesetzt. Im dritten und letzten Training wird ein Dataset herangezogen, das aus allen drei möglichen Formen der Daten besteht.
Die Daten für ihre Arbeit hat Smirnov von CONLL-SIGMORPHON erhalten. In diesen Daten liegt jeweils das Lemma vor, die dazugehörige flektierte Form und eine morphosyntaktische Berschreibung. Aus dem Publikum wird angemerkt, dass bei morphologischen Formen regelmäßig eine Ambiguität aufzufinden ist. Smirnov erklärt daraufhin, dass in den Daten pro target-Form alle möglichen morphosyntaktischen Interpretationen aufgeführt werden.
Der Input, der dem Modell übergeben wird, wird an einem Beispiel präsentiert: LANG=uk IN= LEMMA OUT=N OUT= GEN OUT=GEN OUT=SG *(Angabe eines ukrainischen Worts)*. Es werden hier also Angaben über die Sprache und das Lemma des Inputs gemacht und der gewünschte Output beschrieben. Das Target (der gewünschte Output) besteht aus dem flektierten Wort, in diesem Fall Nomen Genitiv Singular des angegebenen ukrainischen Worts.
Die Daten, die das Modell benötigt sind in Ordnern angelegt. Smirnov verdeutlicht den Aufbau dieser Ordner durch eine Zeichnung an der Tafel. Zunächst gibt es eine „Low-Resource“(LR), in Smirnovs Fall besteht diese aus russischen Daten. Diese bestehen aus 50 oder 200 annotierten russischen Samples. Diese Samples werden mit den „Higher-Resources“ (HR) kombiniert. Die HR bestehen zum einen aus Ukrainischen Daten, zum anderen aus nicht-annotiertem Russisch. Aus diesen Daten werden Sets unterschiedlicher Größe gebildet und jeweils mit den 50 bzw. 200 annotierten LR-Daten kombiniert, sodass alle Kombinationen aus den verschieden großen Sets berücksichtigt werden. Für jeden Ordner existiert ein Train-Set, ein Test-Set und ein Dev-Set, die sich nicht überschneiden dürfen. Dort gibt es jeweils eine Source-File und eine Target-File. Die Source-File enthält alle Lemmas zur Target-File und die Target-File enthält alle flektierten Formen zum Source-File. Eine Source-File besteht jeweils aus den Kombinationen der LR und HR Daten.
Die Erstellung der Daten hat laut Smirnov viel Zeit in Anspruch genommen. Die Resultate der Daten hat sie noch nicht erhalten. Sobald sie sie vorliegen hat, wird sie untersuchen, wie sich die accuracy in den jeweiligen Szenarien (Kombinationen der Daten: nicht-annotiertes Russisch kombiniert mit annotiertem Russisch, nicht-annotiertes Russsich mit annotiertem Ukrainisch oder nicht-annotiertes Russisch mit annotiertem Russisch und annotiertem Ukrainisch) verhält. Smirnov erhofft sich eine bessere accuracy im dritten Fall, also in der Kombination aus beiden HR und der LR. Da Smirnov Russisch spricht, möchte sie ihre Ergebnisse evaluieren und Fehler identifizieren, um eventuelle Muster erkennen zu können und sich Gedanken über eine mögliche Verbesserung des Models zu machen.