Neural Networks for Named-Entity-Recognition

Der erste Vortrag des Tages wurde von Pascal Guldener gehalten und behandelte Neuronale Netze bei der Named-Entity-Recognition. Zu Begin stellte er SENNA, “Semantic/Syntactic Extraction using a Neural Network Architecture”, vor. Dies ist eine Sofware, die mehrere Linguistische Probleme mit Hilfe von “single learning” löst. Darunter zählen die Named-Entity-Recognition, Part-Of-Speech tagging, Chunking und Semantic-Role-Labeling. Eine Besonderheit von SENNA ist, dass das Programm nahezu keine linguistischen Vorkenntnisse benötigt. Die bisherigen traditionellen Methoden verwenden bereits existierendes linguistisches Wissen. Diese Herangehensweise erweist sich bei einzelnen Tasks als sehr effektiv. Es werden sowohl Task-spezifische Features benutzt, als auch der „Output “ bereits existierender Systeme verwendet. Jedoch haben diese Punkte auch Nachteile. Features, die für einen speziellen Task verwendet wurden, sagen meist nichts über die tatsächliche Bedeutung eines Textes aus. Außerdem sind linguistische Features für den Menschen aufwändig zu erarbeiten. Wenn „Output“ von anderen Systemen verwendet wird, kommt es leider auch oft zu Problemen, die die Laufzeit betreffen.

Es gibt zwei große Architekturen in SENNA. Einmal den „Window approach“ und einmal den „Sentence approach“. In SENNA werden Wörter von einem n-dimensionalen Vektor repräsentiert. „n“ ist ein zu optimierender Parameter. Zuerst findet das Training mit einer zufälligen Initialisierung statt, dann werden die „Embeddings“ unbeobachtet verbessert.

Genauer: Jedem Wort wird ein Index zugewiesen und dann in eine „Lookup“-Tabelle gespeichert. Bei dem „Window approach“ wird ein „Window-Vektor“ durch Konkatenierung erstellt. Auf den „Sentence approach“ wurde nicht weiter eingegangen. Über das relevante Tagset wird dann jeweils ein Score ausgegeben.

Als Trainingsdaten wurden die 100000 häufigsten Wörter aus dem Wall Street Journal verwendet. Zusätzlich wurden die Daten noch etwas angepasst. Zum Beispiel die Ersetzung aller Zahlen durch einen String.

Die Ergebnisse wurden mit einem überwachten Benchmark-System verglichen. Die Resultate waren jedoch nicht sonderlich gut. Schuld daran waren die Embeddings, die von nun an separat trainiert wurden.

Als nächstes wurde ein nicht überwachtes Training auf Sprach-Modellen verwendet. Die Trainingsdaten hierfür waren aus Wikipedia und Reuters. Bei der Evaluation kann man erkennen, dass die Werte der Benchmark bei allen Punkten, außer dem Semantic-Role-Labeling übertroffen wurden.

Abschließend lässt sich sagen, dass Wörter, die in einem Task relevant sind, auch in verwandten Tasks relevant sein können. Zusätzlich werden mächtige Wortrepräsentationen erstellt, die auf nicht annotierten Datenbasieren. Weitere Features hinzuzufügen ist jedoch eine schwierige Aufgabe.