Expose´ zur Diplomarbeit

Identifizierung von Adressangaben in Texten – ohne Verwendung von Wörterbüchern

Nora Popp

Juli 2009

Betreuer: Professor Ulf Leser

HU Berlin, Institut für Informatik

Ziel

In dieser Diplomarbeit soll untersucht werden, inwieweit es möglich ist, Adressangaben und adressierbare Entitäten in natürlichsprachlichen Texten zu identifizieren, ohne dabei auf ein Wörterbuch (eine Sammlung von Adressen) zurückgreifen zu müssen. Mit 'adressierbaren Entitäten' sind Orte oder Gebäude (Museen, Denkmäler, Plätze, Wahrzeichen, öffentliche Gebäude ...) gemeint, die z.B. in Stadtplänen zu finden sind, deren Adressen aber nicht explizit im Text angegeben sind, da sie vom Autor des Textes als bekannt angenommen werden.

Motivation

Aufgrund der immer größer werdenden Informationsmenge, die über das Internet zugänglich ist, wird es immer notwendiger, diese Informationen filtern zu können. Damit die Ergebnisse einer Suche so genau wie möglich dem Gesuchten entsprechen, reicht ein einfacher Stringvergleich der Terme einer Suchanfrage mit Termen einer Internetseite nicht aus. Einerseits wird die Ergebnismenge in den meisten Fällen unüberschaubar groß, andererseits entspricht ein Großteil der gelieferten Ergebnisse eventuell nicht den Erwartungen des Nutzers.
Damit Texte im Internet gezielter nach gewünschten Informationen durchsucht werden können, ist es nötig, diese Texte mit Metadaten zu versehen, Daten, die auch den Inhalt eines Textes maschinell auswertbar machen.

Viele Texte im Internet enthalten Adressangaben, mit denen der Inhalt der Texte bestimmten Orten auf einer Landkarte oder einem Stadtplan zugeordnet werden kann. Damit wäre zum Beispiel eine Suche nach lokalen Ereignissen möglich.
Um einen Text einer Adresse zuordnen zu können, muss diese auch als solche erkannt werden und damit eine Adresse von einem anderen System maschinell erkannt werden kann, muss sie vorher explizit markiert worden sein.

In dieser Diplomarbeit wird es darum gehen, Adressangaben zu identifizieren und zu markieren. Es wird nicht versucht werden, Adressangaben in einem Text genauen

Koordinaten zuzuweisen, sondern allgemeine Merkmale für das Erkennen einer Adresse zu finden, damit diese maschinell markiert werden kann.
Damit soll, wenn möglich, eine Grundlage geschaffen werden, Texte mit Adressangaben, die als solche markiert sind, unter Bezugnahme lokaler Kenntnisse, genauen Koordinaten zuweisen zu können.
Da jede Stadt/Ortschaft eine eigene Liste von Straßennamen und Ortsbezeichnungen hat, und diese je nach Größe des Ortes auch sehr, sehr umfangreich sein kann, soll in dieser Arbeit kein Wörterbuch mit Adressangaben verwendet werden. Ein solches Wörterbuch müsste sehr groß sein und würde vermutlich nie vollständig sein.

Vorgehen

Es soll versucht werden, allgemeine Kriterien für Adressangaben zu finden. Hierfür wird ein Korpus, bestehend aus einer Sammlung von Pressemeldungen der Berliner Polizei [1], verwendet. Diese Art von Text scheint für das Vorhaben geeignet, da in relativ wenigen Sätzen ein Tathergang beschrieben wird, der fast immer einer oder mehreren Adressen zugeordnet wird.
Zunächst muss das Korpus in eine verarbeitbare Form gebracht werden. Danach werden manuell alle Vorkommen von Adressangaben markiert und mit Hilfe des TreeTaggers [9] werden allen Token des Korpus ihre Part-of-Speeches und Lemmata zugeordnet. Anschließend werden die Texte in eine Menge von Trainingsdaten und eine Menge von Testdaten unterteilt. Die Testdaten werden bis zur Evaluierung des entwickelten Models nicht mehr betrachtet. Das Trainingskorpus soll daraufhin einer allgemeinen Untersuchung unterzogen werden: "Wie viele und was für Adressangaben kommen im Korpus vor?", "Welche Worte kommen in Sätzen mit Adressangaben vor?", "Welche Worte kommen nicht in Sätzen mit Adressangaben vor?", "Von 'Geburt' bis 'Tod' kann an einer Adresse alles passieren, aber welche Verben sind typisch (typischer) für Sätze mit Adressausdrücken, welche sprechen eher dagegen?" u.s.w.. Danach werden die Sätze des Trainingskorpus, die Adressangaben enthalten, genauer untersucht: "Welche Kombinationen von Part-of-Speech-Tags können eine Adressangabe bilden?", "Welche Strings oder Teilstrings ('...straße...', '...weg...', '...platz...') kommen in Adressangaben immer wieder vor?", "Was für Part-of-Speeches treten vor und nach Adressangaben auf?" u.s.w..

Mit der Beantwortung dieser und weiterer Fragen sollen soviel wie möglich Merkmale von Sätzen mit Adressangaben und Sätzen ohne Adressangaben bestimmt werden. Diese Merkmale werden daraufhin als Dimensionen von Merkmalsvektoren dienen. Da sich die meisten Merkmale vermutlich auf den Kontext einer Adressangabe beziehen werden, sollen zunächst die Trainingssätze klassifiziert werden und erst anschließend die potentiellen Adressangaben identifiziert werden.

Für die Klassifizierung soll die Arbeitsumgebung von RapidMiner [2] genutzt werden. Sie stellt verschiedene Methoden für Maschinelles Lernen und Data-Mining-Prozesse zur Verfügung. Welche Methoden sich hier am besten eignen, wird untersucht werden. Neben der Wahl der Lernmethode wird die Auswahl der Merkmale, die für die Klassifizierung relevant sind, eine große Rolle spielen. Es muss geprüft werden, welche Merkmale für die Klassifizierung maßgeblich und welche uninteressant sind. Wenn ein Merkmal für die Entscheidung, ob ein Satz eine Adressangabe enthält oder nicht, nichts beitragen kann, ist es unwichtig und verlangsamt nur den Verarbeitungsprozess. Außerdem soll der Einfluss bestimmter Merkmale überprüft werden. Zum Beispiel: "Wie groß ist der Informationsverlust, wenn statt der Originaltoken nur deren Lemmata betrachtet werden?" oder "Welche Rolle spielt die Groß- und Kleinschreibung einzelner Token?"

Anschließend sollen alle Sätze, die potentiell eine Adressangabe enthalten, ausgewertet werden. Auch hierbei werden Merkmalsvektoren gebildet, die dann aber speziell auf das Erkennen eines Adressausdrucks abzielen. Dieser Ausdruck muss dann 'nur noch' gegen die restlichen Token seines Satzes abgegrenzt werden.

Um einen Klassifikator zu trainieren wird die Methode der Cross-Validierung angewandt. Hierzu werden die Trainingsdaten in beispielsweise 10 Teile aufgeteilt. Der Klassifikator lernt dann immer auf 9 von 10 Teilen und wird auf dem nichtbenutzten 10. Teil getestet. Das geschieht 10 mal, wobei immer ein anderer Teil nicht zum Lernen sondern zum Testen verwendet wird. Die erzielten Ergebnisse werden anschließend gemittelt.

Wenn eine endgültige Merkmalsauswahl getroffen wurde, wird der Klassifikator auf allen Trainingsdaten trainiert und anschließend auf das bis dahin nicht betrachteten Testkorpus angewandt.

Ob das Ziel, ein gutes Modell zu entwickeln, das Adressangaben in Texten identifiziert, damit erreicht wird, wird sich anhand von Precision und Recall des Modells zeigen.

Verwandte Arbeiten

Der Versuch Adressen und adressierbare Entitäten zu identifizieren ist in das Problemfeld der Named Entity Recognition (NER) einzuordnen. Die klassischen Kategorien hierfür sind Person, Organisation und Location.

Mikheev et. al. [4] haben ein NER-System entwickelt, dass regel-basierte Grammatiken mit statistischen Modellen (Maximum Entropie) kombiniert. Die Zuordnung einer Named Entity (NE) zu einer der genannten Kategorien erfolgt unter Einbeziehung von Kontextinformation, Eigenschaften des Wortes und Wörterbüchern. In verschiedenen Versuchen wurde dabei der Einfluss unterschiedlich großer Wörterbücher auf die Ergebnisse untersucht. Bei Verwendung von großen Wörterbüchern wurden für alle drei Kategorien Recall-Werte zwischen 90 und 96 und Precision-Werte zwischen 93 und 98 angegeben. Ganz ohne Wörterbücher sinken die Werte bei Organisation und Person etwas (Recall: 86 bis 90, Precision: 85 bis 95), bei Location jedoch auf einen Recall-Wert von 46 und einen Precision-Wert von 59. Bei einem weiteren Versuch wurden Wörterbücher aus einem Trainingskorpus generiert. Das Location-Wörterbuch wurde noch um 200 Namen von Ländern und Kontinenten erweitert. Mit Hilfe dieser Wörterbücher, angewendet auf ein Testkorpus desselben Textgenres, ergaben sich Recall-Werte zwischen 87 und 92 und Precision-Werte zwischen 90 und 97. Die Untersuchungen beziehen sich auf englischsprachige Texte. Mikheev et. al betonen jedoch, dass ihr System nicht ohne weitere Modifikationen auf andere Texttypen angewendet werden kann. M. Rössler [5] glaubt, dass Ergebnisse wie bei Mikheev et.al. nur für Sprachen möglich sind, bei denen es ausreicht nach großgeschriebenen Wörtern zu suchen um Named Entities zu finden und Wörterbücher nur zur Klassifizierung benötigt werden. Da im Deutschen alle Substantive groß geschrieben werden und die Wortstellung in deutschen Sätzen teilweise frei wählbar ist, wäre NER hier stark von verlässlichen Wortlisten abhängig.

U. Quasthoff und C. Biemann [6] versuchen bei ihrer Suche nach Vor- und Nachnamen (auch mit Titeln), in deutschen und englischen Texten, mit möglichst wenig Vorwissen zu beginnen. Mit Hilfe handgemachter Regeln und einer Liste von wenigen Vor- und Nachnamen werden neue Namenskandidaten gesucht und verifiziert. Neugefundene Namen werden der Liste der bekannten Namen hinzugefügt und eine neue Suche wird gestartet. Dieser Vorgang wiederholt sich, bis keine neuen Namen dazukommen. In einem weiteren Versuch wurden, mit größerem Anotationsaufwand, Regeln automatisch

gelernt und danach auf die selbe Weise wie im ersten Ansatz angewandt. Im Laufe der Bearbeitung sinkt die Precision des Algorithmus in beiden Fällen aufgrund von falsch erkannten NEs, die dann zum Erkennen neuer Namen benutzt werden. Im zweiten Versuch sinkt die Precision auch durch die Verwendung von weniger strikten Regeln.
T. Zhang und D. Johnson [8] haben den Einfluss von linguistischen Features auf die Ergebnisse von NER in englischen und deutschen Texte untersucht. Sie betrachten das NER-Problem als ein sequentielles Token-basiertes Taggingproblem. Für jedes betrachtete Token wird ein Feature-Vektor erstellt, der auch Feature ein bis zwei Token links und rechts vom betrachteten Token beinhaltet. Je mehr linguistische Feature (Groß-/Kleinschreibung, TokenPräfixe/-Suffixe, POS-Information etc.) einbezogen wurden, umso besser wurden die Ergebnisse. Unter Einbeziehung von Wörterbuchinformation stieg die Güte der Ergebnisse nur noch um ein paar Prozentpunkte. Die Ergebnisse für englische NEs waren allerdings deutlich besser als für deutsche NEs. Mayfield et.al. [3] erzielen ähnliche Resultate wie Zhang und Johnson. Sie benutzen keine Wörterbuchinformation, dafür aber sehr große Feature-Vektoren, die mit Hilfe einer Support-Vector-Machine verarbeitet werden. Die Feature-Vektoren für englische Texte enthalten ca. 600.000 Feature, die für deutsche Texte über 1Mio Feature.
M. Volk und S. Clematide [7] benutzen zur Klassifizierung von NEs in deutschen Texten einerseits Wortlisten für geografische Namen und Personen-Vornamen und lernen andererseits Personen-Nachnamen und Firmennamen aus einem Korpus. Anders als in anderen Ansätzen werden hier POS-Informationen nicht zur Klassifizierung benutzt, sondern mit der Klassifizierung von NEs soll einem Tagger die Unterscheidung zwischen normalem Nomen und NE erleichtert werden.

Obwohl es sich bei Adressen um geografische Ortsangaben handelt, kann das in dieser Arbeit betrachtete Problem nicht unbedingt mit der Suche nach Named Entities der Kategorie Location gleichgesetzt werden, da in den meisten Ansätzen zu diesem Thema eher nach Städte- und Ländernamen gesucht wird. Ausdrücke für Adressen sind vermutlich etwas komplexer. Insofern ist es fraglich, ob die Ergebnisse des Modells, dass in dieser Arbeit entworfen werden soll, mit Ergebnissen anderer NER-Systeme verglichen werden kann.

Quellen

[1] Der Polizeipräsident in Berlin, Polizeipressestelle, Platz der Luftbrücke 6, 12101 Berlin, http://www.berlin.de/polizei/

[2] Mierswa, I. and Wurst, M. and Klinkenberg, R. and Scholz, M. and Euler, T., Yale (now: RapidMiner): *Rapid Prototyping for Complex Data Mining Tasks.* In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2006), 2006. http://www.rapidminer.com/.

[3] J. Mayfield, P. McNamee, C. Piatko. *Named Entity Recognition using Hundreds of Thousands of Features.* In Proceedings of CoNLL-2003.

[4] Mikheev, A., Moens, M., Grover, C. *Named Entity recognition without gazetteers.* In EACL'99

[5] M. Rössler. *Corpus-based Learning of Lexical Resources for German Named Entity Recognition.* In Proceedings of LREC-2004.

[6] U. Quasthoff and C. Biemann. *Named Entity Learning and Verification: Expectation Maximization in Large Corpora.* In Proceedings of CoNLL-2002.

[7] M. Volk and S. Clematide. *Learn-Filter-Apply-Forget. Mixed Approaches to Named Entity Recognition.* In Proceedings of 6th International Workshop on Applications of Natural Language for Information Systems 2001.

[8] T. Zhang and D. Johnson. *A Robust Risk Minimization based Named Entity Recognition System.* In Proceedings of CoNLL-2003.

[9] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/