Lerneinheit: Named Entity Recognition mit Stanford NER lehren

Mareike Schumacher [1]

1. Universität Regensburg

Thema:	Named Entity Recognition	DOI:	10.48694/fortext.3768
Jahrgang:	1	Ausgabe:	9
Erscheinungsdatum:	30-10-2024	Erstveröffentlichung:	2020-02-07 auf fortext.net
Lizenz:	CC BY SA		open access

Allgemeiner Hinweis: Rot dargestellte Begriffe werden im Glossar am Ende des Beitrags erläutert. Alle externen Links sind auch am Ende des Beitrags aufgeführt.

Eckdaten des Lehrmoduls

- Thema der Sitzung: Referenzierung von Figuren in Prosatexten des fin-de-siècle-Jahres 1899
- Lernziele: Kenntnisse über die Methode der Named Entity Recognition, sicherer Umgang mit StanfordNER, kritische Bewertung der Methode, Einsichten in die Ausgestaltung von Figurenbezeichnungen im fin de siècle
- Phasen: Vorstellung und Diskussion der Methode, Demonstration der Toolfunktionen (vgl. Feature), Gruppenarbeit, Gruppenpräsentation, Abschlussdiskussion
- Sozialform(en): Vortrag, Gruppenarbeit, Diskussion
- Medien/Materialien: Alle Lernenden müssen einen Laptop, den StanfordNER heruntergeladen und ein Mal ausgetestet haben; Lehrende benötigen einen Laptop und einen Beamer. In diesem Lehrmodul werden Materialien für bis zu 30 Lernende bereit gestellt.
- Dauer des Lehrmoduls: 2 x 90 Minuten
- Schwierigkeitsgrad des Tools: leicht

Bausteine

- Verlaufsraster des Lehrmoduls: Aus welchen Phasen setzt sich das Lehrmodul zusammen? Dem Verlaufsplan entnehmen Sie Inhalte und Schwerpunkte.
- Anwendungsbeispiel: Anhand welcher Texte unterrichten Sie Named Entity Recognition? Leiten Sie die Studierenden dazu an, Figurenreferenzen in der Literatur des fin de siècle automatisch zu annotieren.
- Verlauf der Unterrichtseinheit(en): Wie sieht die konkrete Ausgestaltung der Phasen aus und welche Arbeitsschritte werden vorgenommen? Erfahren Sie, wie die Unterrichtseinheit strukturiert ist und welche Beispielaufgaben Sie Ihren Studierenden stellen können.
- Lösungen zu den Beispielaufgaben: Hat die Lerngruppe die Beispielaufgaben richtig gelöst? Hier finden Sie Antworten.

Alle Materialien zu dieser Sitzung stellen wir Ihnen auf Zenodo zum Download zur Verfügung (forTEXT 2020).

Verlaufraster des Lehrmoduls

Phase	Impulse des/der Lehrenden	Erwartete Aktivität der Lernenden	Sozialform	Medien / Materialien
Vorab und Einstieg (*etwa 10 Minuten*)	Was ist Named Entity Recognition? In welchen Disziplinen ist die Methode verankert? An welche literaturwissenschaftlichen Traditionen lässt sich damit anknüpfen?	Fragen zum vorab gelesenen Methodeneintrag Named Entity Recognition (Schumacher 2024a) und zur Video-Fallstudie (forTEXT 2018d), Formulieren erster eigener Ideen, wozu die Methode eingesetzt werden kann.	Diskussion im Plenum	Beamer, Laptop
Problematisierung (*etwa 10 Minuten*)	Welche Kategorien werden mit Named Entity Recognition erkannt? Inwiefern sind diese literaturwissenschaftlich relevant? Wie ist das Verhältnis zwischen Text und generierten Daten?	Beteiligung an der Diskussion; Rückbezug auf Methodeneinträge	Diskussion im Plenum	Beamer, Laptop
Erarbeitung (*ca. 70 Minuten*)	Vorstellung der Toolfunktionen; bei Bedarf Einzelbetreuung der Studierenden	Hands-on Named Entity Recognition im Plenum und in Einzelarbeit	Lehrvortrag und Gruppenarbeit	Beamer, Laptop, StanfordNER, Korpus, Trainingsdaten
Sicherung (*ca. 60 Minuten*)	Moderation der Zusammenführung von Arbeitsergebnissen	Sammeln von Beobachtungen	Diskussion im Plenum	Beamer, Laptop
Reflexion & Transfer (*ca. 30 Minuten*)	Diskussion von Schwierigkeiten; Impulse für Transfer geben	Ergebnisse und Schwierigkeiten diskutieren	Diskussion im Plenum	Beamer, Laptops

Das Verlaufsraster steht Ihnen zum Download als PDF-Datei auf Zenodo zur Verfügung (forTEXT 2020).

1. Anwendungsbeispiel

In zwei Seminarsitzungen lehren Sie die Studierenden die Methode der Named Entity Recognition (Schumacher 2024a) anzuwenden, kritisch zu hinterfragen und durch eigene Modelle für literaturwissenschaftliche Anwendungsfälle zu adaptieren. Sie übertragen damit eine ursprünglich computerlinguistische Methode auf die Literaturwissenschaften (vgl. Domäneadaption). Die Studierenden lernen eine einfach zu beherrschende Machine Learning-Technik und werden dadurch nicht nur literaturwissenschaftlich geschult, sondern auch für aktuelle gesellschaftlich relevante technische Entwicklungen sensibilisiert. Der Gegenstand bleibt aber nah am eigenen fachlichen Interesse, denn die Studierenden betrachten eine zentrale literaturwissenschaftliche Kategorie - die Figur - und deren Darstellung in einem Korpus aus Erzähltexten des ausgehenden 19. Jahrhunderts, genauer des Jahres 1899.

2. Verlauf der Unterrichtseinheiten

2.1 Vorarbeiten

Die Studierenden sollten zur Vorbereitung auf die Sitzung den Methodenbeitrag „Named Entity Recognition“ (Schumacher 2024a) und den Toolbeitrag „StanfordNER“ (Schumacher 2024b) gelesen haben. Sie sollten die Videofallstudie „Konstellationen bei Goethe und Plenzdorf“ (forTEXT 2018d) angeschaut haben. Außerdem sollten sie mit Hilfe der Tutorial-Videos zum Stanford NER (forTEXT 2018b; forTEXT 2018a; forTEXT 2018c) und das dazugehörige deutsche Sprachmodell heruntergeladen und bestenfalls ausprobiert haben. Dazu gehört auch, dass im Bedarfsfall die neueste Version von Java und auf Mac-Betriebssystemen das Hilfsprogramm XQuartz installiert wird. Bitten Sie Ihre Studierenden, Ihnen vorab die dabei auftretenden Fehler mitzuteilen, damit Sie sich auf die Lösung technischer Probleme vorbereiten können. Textgrundlage ist ein Teilkorpus des Prosa-Korpus d-Prose (Gius, Guhr und Adelmann 2020). Aus den insgesamt rund 60 Texten in d-Prose, die aus dem Jahr 1899 stammen, haben wir für Sie ein Teilkorpus, ein Trainingskorpus und einen Testtext erstellt. Das Trainingskorpus ist in einzelne Abschnitte unterteilt, die die Studierenden für das Machine-Learning-Training einzeln annotieren können. Sowohl das in den Fallstudien betrachtete Teilkorpus unter diesem Link als auch die Abschnitte des Trainingkorpus und den Testtext (forTEXT 2020) sollten Sie den Studierenden vorab zur Verfügung stellen.
Die Korpora sind so aufgebaut, dass sich Trainingskorpus, Teilkorpus und Testtext nicht überschneiden. Die Beispielannotation im Testtext zeigt, wie die Kategorie „Figur“ hier annotiert werden könnte. Im Diskussionsteil der Einheit können weitere Figurenkonzepte diskutiert werden. Auch bei Nachfragen zur Annotation sollte immer wieder darauf hingewiesen werden, dass während der Sitzung nur einer von mehreren möglichen Annotationsstandards angewendet wird. Es geht hier nicht darum, einen generischen Goldstandard zu erfüllen, sondern ein Modell zu trainieren, das für eine bestimmte eigene Forschungsfrage optimiert ist.
Sollten Sie mit anderen Textdaten arbeiten wollen, so finden Sie weitere Prosatexte in d-Prose, im Deutschen Textarchiv (Horstmann und Kern 2024) oder bei Textgrid (Horstmann 2024b). Auch für Dramentexte kann Named Entity Recognition Gewinn bringend eingesetzt werden. Möchten Sie sich dieser Gattung zuwenden, so finden Sie in DraCor (Horstmann 2024a) eine gute Quelle. Um Trainingskorpus und Testtext so aufzubereiten, dass StanfordNER die Daten verarbeiten kann, folgen Sie der Anleitung in diesem Anleitungsvideo (forTEXT 2018c).
Als Einstieg in die Sitzung bietet sich eine kurze methodische Einführung an, für die Sie diese Beispielfolien nutzen können, die wir Ihnen auf Zenodo bereitstellen (forTEXT 2020). Grundsätzlich kann dieses Lehrmodul sowohl für Präsenzlehre als auch für virtuelle Ersatzformate genutzt werden. Da Sie in einem virtuellen Raum nicht an die Computer der Lernenden herantreten und auftretende Fehler gemeinsam beheben können und da die Lernenden häufig nur mit einem einzigen Bildschirm arbeiten können, also zwischen Videokonferenzraum und Interface des Tools wechseln müssen, ist es ratsam die Einheit in kleinere Schritte aufzuteilen. Im Folgenden finden Sie darum auch immer Angaben, wo Sie unterbrechen und eine Pause einfügen sollten, in der die Kameras ausgeschaltet werden können, Sie aber für einzelne Rückfragen zur Verfügung stehen.

2.2. Einstieg und Problematisierung

Eröffnen Sie die Einheit zur Named Entity Recognition (NER) mit einem kurzen Impulsvortrag zu den Grundlagen der Methode. NER ist ein Verfahren zur automatischen Erkennung bestimmter, vorher klar definierter Einheiten wie z.B. Namen von Personen, Orten oder Organisationen. Die Forschung zu diesem Verfahren hat gezeigt, dass Implementierungen des maschinellen Lernens die besten Ergebnisse zeigen. Ebenfalls häufig werden Wortlisten-Abgleiche durchgeführt, die allerdings weniger effektiv sind. Beim maschinellen Lernen ermittelt ein Programm anhand einer Reihe von Beispielen Muster vorher festgelegter Worteigenheiten. Dazu gehören Besonderheiten des Kontext wie z.B. häufig vor einem Ausdruck stehende Wörter oder Eigenschaften des Wortes selbst wie z.B. Groß- und Kleinschreibung. Eine dritte Art der Umsetzung von Named Entity Recognition besteht darin, dass eigene (also nicht vom Computer im Lernprozess generierte) grammatikartige Regeln für die Kategorien entwickelt und in den Algorithmus implementiert werden. Mischformen, die z.B. maschinelles Lernen mit einem Listenabgleich kombinieren, sind ebenfalls möglich. Versuchen Sie die Lernenden einerseits dafür zu sensibilisieren, in wie vielen alltäglich gebrauchten Programmen und Applikationen ähnliche Logiken wirken und andererseits zu zeigen, wie Sie selbst die Methode für ihr Studium oder ihre Forschung nutzen können.
Regen Sie nach der methodischen Einführung eine kurze Reflexion zur Verknüpfung der Methode mit eigenen Projekten an. Fragen Sie die Lernenden, ob sie selbst Anwendungen kennen, in denen maschinelles Lernen eine Rolle spielt. Fragen Sie dann, inwiefern sie durch automatische Annotation bei eigenen Projekten unterstützt werden können. Wenn es keine zurückliegenden oder gegenwärtigen Projekte gibt, an denen die Teilnehmenden arbeiten, fragen Sie nach Ideen für zukünftige Projekte. Geben Sie Rückmeldung, ob die Ideen realistisch sind oder nicht. Dabei können Sie folgende Regel anwenden: je klarer die Kategorie definiert ist und je eindeutiger sie auf eine bestimmte Wortgruppe passt, desto höher die Wahrscheinlichkeit, dass sie für einen Computer erlernbar ist. Je ungenauer eine Kategorie und je vielfältiger eine Wortgruppe, zu der sie passt, desto schlechter

werden die Ergebnisse einer automatischen Annotation sein.
Ziel der Einstiegsphase ist, dass die Lernenden ein Grundverständnis der Methode entwickeln. Sie sollen versuchen, eigene Anforderungen für eine automatische Annotation zu formulieren. Dabei soll der Horizont erst einmal eröffnet werden und kreative Ideen entwickelt. Was konkret mittels NER umsetzbar ist, soll die nächste Phase zeigen.

2.3. Erarbeitung

StanforNER muss nicht installiert werden, sondern kann direkt aus dem von der StanfordNER-Webseite heruntergeladenen Ordner gestartet werden. Zeigen Sie auf Ihrem Rechner, wie Sie den heruntergeladenen Ordner öffnen und zum Tool navigieren. Sie finden im heruntergeladenen Ordner mehrere .JAR-Dateien, die das Tool in optimierter Form für unterschiedliche Betriebssysteme enthalten. Bitten Sie die Lernenden, eine der Dateien per Doppelklick zu öffnen. Führen Sie das Lehrmodul in Präsenz durch, so können Sie nun an jeden Computer herantreten, um sicher zu stellen, dass das Tool sich bei jedem korrekt öffnen lässt. Lehren Sie in einer virtuellen Konferenzumgebung, so machen Sie hier 10 Minuten Pause, in denen die Lernenden ihre Kameras ausschalten können und die gezeigten Schritte in Ruhe nachvollziehen können. Stehen Sie in dieser Zeit selbst für Rückfragen zur Verfügung und assistieren sie einzelnen Teilnehmenden im Bedarfsfall.
Zeigen Sie nun, wie Sie aus dem heruntergeladenen Ordner mit den deutschen Sprachmodellen das NER-Modell ins Tool laden, indem Sie zuerst auf die Schaltfläche „load Classifier“ und dann im sich öffnenden Drop-Down-Menü auf „load CRF from file“ klicken. Nun können Sie aus Ihrer Ordner-Struktur das NER-Modell für die deutsche Sprache auswählen. Es sollten sich nach kurzer Zeit am rechten Rand des User Interfaces von StanfordNER die Kategorien „Person“, „Ort“ und „Organisation“ zeigen. Führen Sie nun vor wie Sie auf ähnliche Weise einen Text laden. Klicken Sie auf die Schaltfläche „File“ und dann im sich öffnenden Drop-Down-Menü auf „Open file“. Wählen Sie aus Ihrer Ordnerstruktur einen Text aus dem hier bereitgestellten Kernkorpus. Klicken Sie dann auf die Schaltfläche „Run NER“. Machen Sie erneut eine Pause und assistieren Sie den Lernenden einzeln dabei, diese Schritte selbst auszuführen. Lehren Sie in einer virtuellen Umgebung, so machen Sie eine 10-15-minütige Kamera-Pause, in der die Lernenden zum Interface des Tools zurückkehren und sich auf die Bedienung konzentrieren können. Ermöglichen Sie in der Zeit einzelnen, ihren Bildschirm mit Ihnen zu teilen, damit Sie assistieren können.

Ins Plenum zurück gekehrt, besprechen Sie gemeinsam die folgenden Aufgaben:

Aufgabe 1: Was fällt Ihnen bei Betrachtung der automatischen Annotation auf? Was erkennt das Tool in dieser vortrainierten Weise gut? Was entgeht dem Programm? Können Sie, anknüpfend an die vorbereitend gelesenen Materialien, erklären, warum bestimmte Wörter korrekt annotiert werden und andere nicht?

Beginnen Sie nun zunächst gemeinsam mit der Annotation eines eigenen Trainingskorpus. Teilen Sie dafür jedem Lernenden eine der hier als Trainingsdaten bereitgestellten Tabellen zu. Führen Sie beispielhaft die Annotation des Anfangs einer weiteren Tabelle vor. Annotieren Sie alle Referenzen auf Figuren, indem Sie in der zweiten Spalte hinter einer Figurenreferenz das „O“ (das hier für die Kategorie „other“ steht und bei der Vorbereitung der Trainingsdaten von Stanford NER automatisch so annotiert wird) durch „Figur“ ersetzen. Legen Sie dabei im Plenum fest, wie Sie mit Zweifelsfällen umgehen wollen (z.B. ob Personalpronomen mit annotiert werden sollen oder nicht, ob Sie nur Namen oder auch Bezeichnungen wie „Bruder“ oder „Witwe“ annotieren wollen). In der verbleibenden Zeit dieser Sitzung können die Lernenden anfangen, die ihnen zusortierte Tabelle zu annotieren. Währenddessen können weitere Zweifelsfälle besprochen werden, die sich im Laufe der Annotation zeigen. Die Annotation der Tabelle ist als Vorbereitung der zweiten Sitzung zu Ende zu führen. Nutzen Sie dieses Lehrmodul im Seminarkontext, so haben die Studierenden in der Regel nun eine Woche Zeit, an der Annotation der Trainingsdaten zu arbeiten. Bieten Sie ein Blockseminar oder einen Workshop an, so sollten Sie an dieser Stelle eine längere Pause einplanen. In der Regel können Tabellen von ca. 4.000 Tokens Umfang wie die hier bereitgestellten in ungefähr 60-90 Minuten vollständig annotiert werden. Im Optimalfall geben Sie also Workshop- oder Blockseminarteilnehmenden mindestens 120 Minuten Zeit, damit sie nach der Annotationsaufgabe noch eine Bildschirmpause von 30 Minuten machen können. Wenn ihr Zeitrahmen das nicht zulässt, so geben Sie eine Zeit zum Annotieren vor und arbeiten Sie dann mit dem weiter, was die Lernenden geschafft haben. Verzichten Sie im virtuellen Lehrformat auf keinen Fall auf eine angemessene Pause, da der zweite Teil des Lehrmoduls deutlich mehr Konzentration erfordert als der erste. Nutzen Sie selbst diese Zeit um so viel wie möglich vom Testtext nach den von Ihnen mit den Studierenden erarbeiteten Richtlinien zu annotieren. Der hier zur Verfügung gestellte Testtext beinhaltet einen Ausschnitt von 10.000 Tokens aus Ganghofers Gotteslehen. Um Ihre Annotation zu beschleunigen und zu vereinfachen, haben wir den Testausschnitt mit einem unserer Classifier vorannotiert. Sie müssen die Annotation also lediglich ergänzen und korrigieren. Bitten Sie die Lernenden Ihnen die Tabellen nach Abschluss der Annotationsphase direkt zuzuschicken (ggf. mit einem Hinweis darauf, wie viel annotiert wurde). Kopieren Sie alle annotierten Daten in ein Tabellendokument, das Sie im TSV-Format speichern und vor dem zweiten Teil des Moduls allen Lernenden zur Verfügung stellen.

Beginnen Sie den zweiten Teil des Moduls mit einem kurzen Erfahrungsaustausch. Ist den Lernenden die Annotation leicht gefallen? Worüber sind sie gestolpert? In der Regel fallen einem eine Reihe von Kleinigkeiten auf, wenn man zum ersten Mal mit der Aufbereitung eines Trainingskorpus für Machine Learning konfrontiert wird. Dazu gehört ein intuitiver Beginn der Annotation bis zum Auftauchen eines ersten Zweifelsfalls. Je nach Art der Mehrdeutigkeit, die in einem Ausdruck stecken kann, ist es möglich, dass dieser eine Fall die ganze bisherige Annotationsweise in Frage stellt. In jedem Fall setzt ein Reflexionsprozess ein, der sich an relativ kleinen, unauffälligen Phänomenen aufhängt. Es sind diese kleinen Erfahrungen mit Zweifelsfällen, die häufig ein Umdenken in Bezug auf die Methodik bewirken. Die Lernenden erkennen, mit welcher Sorgfalt in der Aufbereitung von Daten bewusst Interpretationsentscheidungen getroffen werden. Dabei wirkt die einzelne Annotation häufig unbedeutend. Es ist wichtig, dass Sie diese Erfahrung, die den Studierenden mitunter selbst etwas unangenehm als Erkenntnis von etwas eigentlich Offensichtlichem vorkommt, nicht nivellieren oder abwerten. Genau hier beginnt der Weg zum erfahrungsbasierten Verständnis der Vor- und Nachteile der vermittelten Methode.
Bitten Sie die Lernenden nun, die Tabelle mit den Trainingsdaten im selben Ordner wie den StanfordNER abzulegen. Sie haben nun einen lernfähigen Algorithmus und annotierte Beispieldaten, aus denen dieser lernen kann. Was Sie noch brauchen, ist ein Dokument, in dem einerseits festgehalten wird, anhand welcher Wort- und Kontextmerkmale das Tool lernen soll (die sogenannten Features), einen Hinweis darauf, in welchem Dokument die Trainingsdaten abgelegt sind und eine Angabe, unter welchem Namen der aus dem Training resultierende Classifier gespeichert werden soll. All diese Informationen können Sie für das Tool in einer Datei ablegen, die Properties-Datei genannt wird. Um eine solche zu erstellen, bitten Sie die Lernenden, einen Texteditor (TextEdit, Notepad, BBEdit oder Visual Studio Code, nicht Word oder OpenOffice oder andere Programme, die mehr als Reintextverarbeitung anbieten) zu öffnen. Lassen Sie sie folgende Zeilen in ihr Dokument kopieren:

```
trainFile = training-data.tsv
serializeTo = ner-model.ser.gz
map = word=0,answer=1

useClassFeature=true
useWord=true
useNGrams=true
noMidNGrams=true
maxNGramLeng=6
usePrev=true
useNext=true
useSequences=true
usePrevSequences=true
maxLeft=1
useTypeSeqs=true
useTypeSeqs2=true
useTypeySequences=true
wordShape=chris2useLC
useDisjunctive=true
```

Bitten Sie sie „training-data“ durch den Namen der Trainingsdatei und „ner-modell“ durch den Namen zu ersetzen, den das eigene NER-Modell haben soll, z.B. FigurenClassifier. Anschließend muss die Datei im selben Ordner wie der StanfordNER abgelegt werden. Die Datei-Endung muss PROP lauten (z.B. figuren.prop).
Zeigen Sie den Lernenden nun zunächst auf Ihrem Computer, wie sie die Kommandozeile öffnen und zum Ordner navigieren können, in dem die Daten liegen. Gehen Sie dann zu den einzelnen Lernenden hin und assistieren Sie dabei. Im digitalen Raum ist es hier etwas schwierig individuell zu helfen, da die Commandline bei unterschiedlichen Betriebssystemen auch unterschiedlich zu erreichen ist. Geben Sie den Teilnehmenden wieder eine 10-minütige Kamera-Pause und bieten Sie einzeln Hilfe an, indem Sie das Screen-Sharing nutzen. Allgemein gilt: Die Commandline erreichen Sie unter Windows, indem Sie unten links auf das Windows-Symbol klicken und dann in die Suchleiste „cmd“ eingeben. Unter Mac heißt die Commandline „Terminal“ und kann bei den „Dienstprogrammen“ gefunden werden.
Bei Lernenden ohne technische Vorkenntnisse bietet es sich nun an, Schritt für Schritt mit dem cd-Command zum Ordner mit dem StanfordNER zu navigieren. Bei einer Präsenzveranstaltung können die Studierenden parallel auf Ihren eigenen Laptops navigieren, brauchen dabei aber zum Teil etwas Assistenz. Bei einer virtuellen Veranstaltung gehen Sie wie gehabt vor. Zeigen Sie zunächst den Ablauf ein Mal ganz auf Ihrem Bildschirm. Machen Sie dann eine 10-minütige Pause, in der Sie einzelnen Teilnehmenden helfen.
Sind alle über die Commandline im richtigen Ordner angekommen, bitten Sie sie folgende Zeile Code zu kopieren und bei sich einzufügen:

```
java -cp stanford-ner.jar edu.stanford.nlp.ie.crf.CRFClassifier -prop figuren.
    ↪ prop
```

(Ändern Sie gegebenenfalls den Dateinamen der PROP-Datei). Mit der Bestätigung „enter“ beginnt der Algorithmus zu lernen. Dabei können Fehler auftreten. Gängige Fehler sind:

- *argument array differs* - in einer Zeile der Tabelle sind weniger Spalten ausgefüllt als ausgefüllt sein müssten. In der Regel fehlt in einer Zeile eine Annotation. Diesen Fehler können Sie beheben, indem sie in einem Tabellenprogramm nach leeren Zellen suchen und sie z.B. durch „O“ ersetzen. Da Sie alle mit demselben Tabellendokument arbeiten, wird der Fehler bei allen auftreten. Die Fehlerbehebung müssen also auch alle durchführen. Lehren Sie virtuell geben Sie dafür wieder eine Kamera-Pause von 10 Minuten.
- *No such file or directory* - Sie haben einen Tippfehler in der Properties-Datei. Wahrscheinlich stimmt der Name der Trainingsdatei nicht ganz (z.B. Groß- statt Kleinschreibung).

Das gemeinsame Beheben von Fehlern ist ein wichtiger Schritt im Erlernen der Methodik des maschinellen Lernens. Es zeigt, wie präzise hier gearbeitet werden muss. Wir empfehlen darum, nicht von vornherein solche Fehler bei Ihrer Vorbereitung „heimlich“ auszumerzen, sondern Sie in der Lehre passieren zu lassen, um dann gemeinsam mit den Lernenden daran zu arbeiten.

Aufgabe 2: Sorgen Sie dafür, dass der Prozess des maschinellen Lernens korrekt ausgelöst und durchgeführt wird.

Während sich im Kommandozeilen-Fenster zeigt, wie das Tool arbeitet, können Sie noch etwas zur Iterativität des Lernprozesses erklären. Ziehen Sie eine Parallele zu den eigenen Erfahrungen der Studierenden beim manuellen Annotieren.

Aufgabe 3: Diskutieren Sie im Plenum Gemeinsamkeiten und Unterschiede des menschlichen und des maschinellen Lernens. Wie haben Sie die Annotation der Kategorie Figur erlernt? Wie versucht das Programm gerade diese Kategorie zu erlernen?

Ist der Classifier fertig trainiert, so bitten Sie die Lernenden wieder den StanfordNER zu öffnen. Laden Sie wieder denselben Text aus dem Kernkorpus wie in der ersten Sitzung. Laden Sie statt des vortrainierten NER-Classifiers für die deutsche Sprache das eigene NER-Modell. Klicken Sie auf „Run NER“.

2.4. Sicherung

Fragen Sie die Teilnehmenden nun nach Beobachtungen der automatischen Annotation dieses neuen Classifiers. Was wird gut annotiert? Wo wurde fehlerhaft annotiert? Welche Art von Fehlern gibt es? Erwähnen Sie die Unterscheidung von true und false positives und negatives. Bitten Sie die Lernenden diese automatische Annotation mit der in der ersten Sitzung durchgeführten zu vergleichen. Lassen Sie sie am Ende die annotierte Textdatei über „File > save tagged file as“ abspeichern.

2.5. Transfer & Reflexion

Öffnen Sie nun die Diskussion für weitere Erfahrungen und Beobachtungen. Welche Aspekte der Methode hat die Lernenden überrascht? Durch welche Eigenheiten von NER fühlten sie sich in ihren Vorannahmen bestätigt? Kehren Sie zurück zu den Ergebnissen des Brainstormings im ersten Teil des Lehrmoduls. Halten die Lernenden ihre Projektideen immer noch für den Einsatz von NER geeignet? Haben sie neue Ideen? Wird die Methode für ihren zukünftigen Studien- oder Forschungsprozess nützlich sein?

3. Lösungen zu den Beispielaufgaben

In diesem Lehrmodul gibt es nur offene Reflexionsfragen, zu denen es keine eindeutigen Lösungen gibt. Entwickeln Sie stattdessen gemeinsam mit den Lernenden eine Reflexion der Methode.

Externe und weiterführende Links

- Alle Materialien zu diesm Lehrmodul auf Zenodo: https://zenodo.org/records/10519536 (Letzter Zugriff: 07.10.2024)
- Anleitungsvideo zur Aufbereitung des Trainingskorpus und des Testtexts: https://web.archive.org/save/https://zenodo.org/records/10371086 (Letzer Zugriff: 07.10.2024)
- d-Prose Korpus: https://web.archive.org/save/https://zenodo.org/record/5015008#.YPfI05Mza8U (Letzer Zugriff: 07.10.2024)
- Stanford CoreNLP Sprachmodell: https://web.archive.org/save/https://stanfordnlp.github.io/CoreNLP/index.html#download (Letzer Zugriff: 07.10.2024)

- d-Prose Korpus: https://web.archive.org/save/https://zenodo.org/record/5015008#.YPfI05Mza8U (Letzer Zugriff: 07.10.2024)
- Video-Fallstudie zu *Figurenkonstellationen in Goethes Werther und Plenzdorfs neuem Werther*: https://web.archive.org/save/https://doi.org/10.5281/zenodo.10250582 (Letzer Zugriff: 07.10.2024)

Bibliographie

forTEXT. 2018a. Tutorial: Stanford Named Entity Recognizer installieren und deutsche Kategorien laden. Named Entity Recognition und Literaturanalyse. 15. August. doi: 10.5281/zenodo.10372231, https://youtu.be/hYed-ZqEzs8.

———. 2018b. Tutorial: Stanford Named Entity Recognizer zur digitalen Literaturanalyse nutzen. Named Entity Recognition und Literaturanalyse. 31. August. doi: 10.5281/zenodo.10372239, https://youtu.be/n35i4Cy2c7Y.

———. 2018c. Tutorial: Ein eigenes NER-Modell für die digitale Literaturanalyse trainieren. Named Entity Recognition und Literaturanalyse. 14. September. doi: 10.5281/zenodo.10371086, https://zenodo.org/records/10371086.

forTEXT. 2018d. Tutorial: Figurenkonstellationen in Goethes Werther und Plenzdorfs neuem werther. Zenodo, 4. Oktober. doi: 10.5281/zenodo.10250582, https://doi.org/10.5281/zenodo.10250582.

forTEXT. 2020. Named Entity Recognition mit Stanford NER lehren. 7. Februar. doi: 10.5281/zenodo.10519535, https://zenodo.org/records/10519536.

Gius, Evelyn, Svenja Guhr und Benedikt Adelmann. 2020. d-Prose 1870-1920. Zenodo, 15. Dezember. doi: 10.5281/zenodo.4315209, https://zenodo.org/record/4315209 (zugegriffen: 16. Dezember 2020).

Horstmann, Jan. 2024a. Ressourcenbeitrag: DraCor - Drama Corpora Project. Hg. von Evelyn Gius. *forTEXT* 1, Nr. 6. Netzwerkanalyse (30. August). doi: 10.48694/fortext.3785, https://fortext.net/ressourcen/textsammlungen/dracor-drama-corpora-project.

———. 2024b. Ressourcenbeitrag: TextGrid Repository. Hg. von Evelyn Gius. *forTEXT* 1, Nr. 11. Bibliografie (29. November). doi: 10.48694/fortext.3794, https://fortext.net/ressourcen/textsammlungen/textgrid-repository.

Horstmann, Jan und Alexandra Kern. 2024. Ressourcenbeitrag: Deutsches Textarchiv (DTA). Hg. von Evelyn Gius. *forTEXT* 1, Nr. 11. Bibliografie (29. November). doi: 10.48694/fortext.3791, https://fortext.net/ressourcen/textsammlungen/deutsches-textarchiv-dta.

Schumacher, Mareike. 2024a. Methodenbeitrag: Named Entity Recognition (NER). Hg. von Evelyn Gius. *forTEXT* 1, Nr. 9. Named Entity Recognition (30. Oktober). doi: 10.48694/fortext.3765, https://fortext.net/routinen/methoden/named-entity-recognition-ner.

———. 2024b. Toolbeitrag: Stanford Named Entity Recognizer. Hg. von Evelyn Gius. *forTEXT* 1, Nr. 9. Named Entity Recognition (30. Oktober). doi: 10.48694/fortext.3767, https://fortext.net/tools/tools/stanford-named-entity-recognizer.

Glossar

Annotation Annotation beschreibt die manuelle oder automatische Hinzufügung von Zusatzinformationen zu einem Text. Die manuelle Annotation wird händisch durchgeführt, während die (teil-)automatisierte Annotation durch Machine-Learning-Verfahren durchgeführt wird. Ein klassisches Beispiel ist das automatisierte PoS-Tagging (Part-of-Speech-Tagging), welches oftmals als Grundlage (Preprocessing) für weitere Analysen wie Named Entity Recognition (NER) nötig ist. Annotationen können zudem deskriptiv oder analytisch sein.

Domäneadaption Domäneadaption beschreibt die Anpassung einer in einem Fachgebiet entwickelten digitalen Methode an ein anderes Fachgebiet.

Feature Unter Features können Einzelfunktionen eines Tools verstanden werden, die beispielsweise komplexe Funktionen wie die Visualisierung eines Textes als Wordcloud ermöglichen, oder auch kleinere Funktionseinheiten wie den Abgleich einzelner Spracheigenschaften (Properties) mit annotierten Beispieltexten darstellen.

Korpus Ein Textkorpus ist eine Sammlung von Texten. Korpora (Plural für „das Korpus“) sind typischerweise nach Textsorte, Epoche, Sprache oder Autor*in zusammengestellt.

Lemmatisieren Die Lemmatisierung von Textdaten gehört zu den wichtigen Preprocessing-Schritten in der Textverarbeitung. Dabei werden alle Wörter (Token) eines Textes auf ihre Grundform zurückgeführt. So werden beispielsweise Flexionsformen wie „schneller“ und „schnelle“ dem Lemma „schnell“ zugeordnet.

Machine Learning Machine Learning, bzw. maschinelles Lernen im Deutschen, ist ein Teilbereich der künstlichen Intelligenz. Auf Grundlage möglichst vieler (Text-)Daten erkennt und erlernt ein Computer die häufig sehr komplexen Muster und Gesetzmäßigkeiten bestimmter Phänomene. Daraufhin können die aus den Daten gewonnen Erkenntnisse verallgemeinert werden und für neue Problemlösungen oder für die Analyse von bisher unbekannten Daten verwendet werden.

Named Entities Eine Named Entity (NE) ist eine Entität, oft ein Eigenname, die meist in Form einer Nominalphrase zu identifizieren ist. Named Entities können beispielsweise Personen wie „Nils Holgerson“, Organisationen wie „WHO“ oder Orte wie „New York“ sein. Named Entities können durch das Verfahren der Named Entity Recognition (NER) automatisiert ermittelt werden.

POS PoS steht für *Part of Speech*, oder „Wortart“ auf Deutsch. Das PoS- Tagging beschreibt die (automatische) Erfassung und Kennzeichnung von Wortarten in einem Text und ist of ein wichtiger Preprocessing-Schritt, beispielsweise für die Analyse von Named Entities.

Preprocessing Für viele digitale Methoden müssen die zu analysierenden Texte vorab „bereinigt“ oder „vorbereitet“ werden. Für statistische Zwecke werden Texte bspw. häufig in gleich große Segmente unterteilt (*chunking*), Großbuchstaben werden in Kleinbuchstaben verwandelt oder Wörter werden lemmatisiert.

Property Property steht für „Eigenschaft“, „Komponente“ oder „Attribut“. In der automatischen Annotation dienen konkrete Worteigenschaften wie Groß- und Kleinschreibung zur Klassifizierung von Wörtern oder Phrasen. Durch die Berücksichtigung solcher Eigenschaften in den Features eines Tools kann maschinelles Lernen bestimmter Phänomene umgesetzt werden. In der manuellen Annotation können als Properties auch Eigenschaften von Annotationen benannt werden.

TSV TSV ist die englische Abkürzung für *Tab Separated Values* . Es handelt sich um ein Dateiformat zur einheitlichen Darstellung und Speicherung von einfach strukturierten Daten mit dem Kürzel .tsv, sodass diese problemlos zwischen IT-Systemen ausgetauscht werden können. Dabei sind alle Daten zeilenweise angeordnet. Alle Zeilen wiederum sind in einzelne Datenfelder aufgeteilt, welche durch den Tabulatoren als Trennzeichen geordnet werden. In Programmen wie Excel können solche Dateien als Tabelle angezeigt werden.

Type/Token Das Begriffspaar „Type/Token“ wird grundsätzlich zur Unterscheidung von einzelnen Vorkommnissen (Token) und Typen (Types) von Wörtern oder Äußerungen in Texten genutzt. Ein Token ist also ein konkretes Exemplar eines bestimmten Typs, während ein Typ eine im Prinzip unbegrenzte Menge von Exemplaren (Token) umfasst.
Es gibt allerdings etwas divergierende Definitionen zur Type-Token-Unterscheidung. Eine präzise Definition ist daher immer erstrebenswert. Der Satz „Ein Bär ist ein Bär.“ beinhaltet beispielsweise fünf Worttoken („Ein“, „Bär“, „ist“, „ein“, „Bär“) und drei Types, nämlich: „ein“, „Bär“, „ist“. Allerdings könnten auch vier Types, „Ein“, „ein“, „Bär“ und „ist“, als solche identifiziert werden, wenn Großbuchstaben beachtet werden.

Wordcloud Eine *Wordcloud*, oder auch Schlagwortwolke, ist eine Form der Informationsvisualisierung, beispielsweise von Worthäufigkeiten in einem Text oder einer Textsammlung. Dabei werden unterschiedlich gewichtete Wörter, wie die häufigsten Wörter, i.d.R. größer oder auf andere Weise hervorgehoben dargestellt. Die horizontale/vertikale Ausrichtung und die Farbe der dargestellten Wörter hat meistens allerdings keinen semantischen Mehrwert.