GOETHE UNIVERSITY FRANKFURT

Institute for Computer Science

MASTER THESIS

Natural Language Processing auf mehrsprachigen Münzdatensätzen

-

Untersuchung der Qualität, Datenqualität und Übertragbarkeit auf andere Datensätze

Chrisowalandis Deligio

&

Kerim Gencer

01.01.2021

Betreut von

Prof. Dott.- Ing Roberto V. Zicari

Databases and Information Systems (DBIS)

Inhalt

1. Einleitung

Motivation Kaum eine Freizeitbeschäftigung ist wohl so alt und beliebt wie das Sammeln von historischen Münzen. Man berichtet davon, dass es selbst unter den alten römischen Aristokraten Münzensammler gab. Vermutlich bekanntester Vertreter hierbei ist, laut den Überlieferungen von Sueton und Plinius, Kaiser Augustus, der vor mehr als 2000 Jahren lebte (Kroha 1968). Er pflegte es alte königliche und ausländische Münzen zu sammeln und beschenkte nahestehende Personen zu Neujahrstagen mit historischen Münzen (Haymann 2016, Einführung).

Seit Jahrhunderten dienten und dienen Münzen als Zahlungsmittel und Tauschobjekte. Doch darüber hinaus haben sie einen viel größeren Wert – nämlich den, als unvergleichbare Zeitzeugen, die uns ansatzweise ein Bild über den Zeitgeist unsere Vorfahren vermitteln. Der Österreicher Numismatiker Robert Göbl bezeichnete diese besondere Eigenschaft der Münzen als ihren »Doppelcharakter« (Göbl 1987, S. 20ff.).
Die Numismatik (aus dem lateinischem »*nómisma*« für Münze), auch Münzkunde genannt, hat sich dieses Medium zur Wissenschaft gemacht. Dabei liefern Münzen als historische Primärquellen Erkenntnisse über Wirtschafts- und Kulturgeschichte, besonders für das römische und griechische Altertum. Eine über Jahrhunderte anhaltende historische Kontinuität hat diesem Medium die Besonderheit gegeben, dass den numismatischen Überlieferungen kaum ein Ereignis entgeht. Es wird geschätzt, dass aus dem griechischen, römischen, keltischen und byzantinischem Raum bis heute noch ca. 3000- 10000 Typen von Münzen überlebt haben (Haymann 2016, Einführung).

Corpus Nummorum Online[1] (CNO) bzw. ehemals *Corpus Nummorum Thracorum* (CNT) ist ein Projekt in Form eines Webportals, das sich mit Münzen aus Moesien, Thrakien, Mysien und Troas beschäftigt. Die Zusammenstellung dieser antiken griechischen Münzen soll für verschiedene Forschungszwecke und zum Erhalt des Kulturerbes dienen. Alle Münzen wurden durch standardisierte Kriterien mit einer Beschreibung ihrer Münzbilder versehen. Diese existieren sowohl in deutscher und als auch englischer Sprache. Durch CNO ist es möglich die Münzen nach Münzstätten, Münzsorten und auch Münzstempel zu

[1] https://www.corpus-nummorum.eu/about (25.07.20)

gruppieren oder zu sortieren. Mit Hilfe der *Coin Advanced Search*[2] ist es möglich, durch Suchkriterien wie Person, ihrer Funktion oder der Epoche nach konkreten Münzen zu suchen. Hierfür werden die Schlüsselbegriffe aus den Ikonographen der Münzen extrahiert. Beim Datenbankdump vom 5. Januar 2020 stehen hierbei 5542 ikonographische Beschreibungen zur Verfügung, die alle in einer relationalen MySQL Datenbank eingelagert wurden. Dabei sehen die Beschreibungen, jeweils in englischer und deutscher Sprache, beispielsweise wie folgt aus:

> »Athena standing facing, head left, wearing long garment and helmet, holding patera in outstretched right hand and spear in left arm; at her feet, shield placed on ground.«[3]

> »Athena stehend von vorn, Kopf nach links, im langem Chiton und mit Helm, in der vorgestreckten Rechten Patera, in der Linken Speer haltend; vor ihr, Schild.«.

Die Schlüsselbegriffe in diesem Satz sind der Name »Athena« und die Objekte »Chiton«, »Helm«, »Patera«, »Speer« und »Schild«. Diese Begriffe gehören alle zur Klasse von Entitäten, die in die Unterklassen Personen, Objekte, Tiere und Pflanzen aufzuteilen sind. Diese Entitäten wurden genau wie die Ikonographen in die Datenbank eingespeist. Das Extrahieren und Einspeichern dieser essentiellen Schlüsselbegriffe erfolgte jedoch durch manuelle Handarbeit und nicht automatisch.

Die so genannte *Information Extraction* (IE) ist eine Methode, um die Bedeutung oder den Sinngehalt eines Textes zu erfassen (Bird, Klein und Loper 2009, Kap. 7.1). Hierbei sind besonders zwei Schritte der IE genauer zu betrachten; die Entitätenerkennung, *named entity recognition* (NER) und die Relationsextraktion, *relation extraction* (RE). Unter *named entity* (NE) ist ein reales Objekt mit Eigennamen zu verstehen (Vasiliev 2020, Kap. 2). Dies können Personen, Organisationen, Orte oder auch andere Entitäten sein. Bird, Klein und Loper beschreiben NEs auch als Nominalphrasen, die auf bestimmte Typen von Individuen verweisen (Bird, Klein und Loper 2009, Kap. 7.5). Zwischen diesen NEs können Relationen festgestellt und extrahiert werden. Dabei ist das Ziel, die Beziehungen der Entitäten, an

[2] https://www.corpus-nummorum.eu/advancedSearch (25.07.20)
[3] https://www.corpus-nummorum.eu/coins?id=815 (25.07.20)

denen sie beteiligt sind, zu identifizieren – was informell so viel bedeutet wie: »Wer« hat »Wem« »Was«, »Wann« und »Warum« angetan (Marquez et al. 2008, Introduction). Die Relationen werden dabei in einer Tripelform annotiert (NE_1, α, NE_2), wobei α die Wortfolge ist, die die beiden Entitäten NE_1 und NE_2 mit einander verbindet bzw. zwischen beiden eingreift (Bird, Klein und Loper 2009, Kap. 7.6).

Diese manuell-händische Arbeit zu reduzieren, um ein verfeinertes Suchen für die numismatische Gemeinschaft zu ermöglichen, war das Ziel der Bachelorarbeit »*Natural Language Processing to enable semantic search on numismatic descriptions*« von Patricia Klinger (Klinger 2018). Mit Hilfe eines *Natural Language Processing* (NLP) Ansatzes unter der Programmiersprache *Python* wurde ein NER und RE umgesetzt. In Anbetracht der Notwendigkeit von unkomplizierter Erstellung eigener Entitäten, fiel P. Klingers Entscheidung auf *spaCy*[4]. Die Entitäten beschränkten sich dabei auf historische als auch mythologische Personen und Objekte der englischen Beschreibungen. Dementsprechend wurden auch nur die Relationen zwischen Personen und Objekten betrachtet und bearbeitet. Die Entitätserkennungsrate gemessen als F-Maß (siehe **Kapitel 3.3**) erreichten ganze 97%. Bei den zu erkennenden Relationen wurde das »Wer«, »Wem« und »Was« untersucht. Das F-Maß erreichte für das RE 88%. An jenen Ansatz knüpft diese Masterarbeit an.

Aufgabenstellung NLP als ein Teilbereich der künstlichen Intelligenz setzt es sich zur Aufgabe, natürliche Sprachen analysieren und verarbeiten zu können. Somit bildet es für diese Arbeit die passende Grundlage (Bird, Klein und Loper 2009, Kap. 1). Ziel dieser Arbeit ist es Erweiterungen als auch Optimierungen in den Bereichen des NERs und des REs umzusetzen. Als Grundlage dient die Arbeit von P. Klinger. Die Leistung dieser Arbeit lässt sich mit folgenden vier Aspekten beschreiben:

I. An erster Stelle gilt es, die bereits bestehenden Entitätsklassen zu ergänzen.
   Im konkreten bedeutet dies, dass die Entitätsklassen Tier und Pflanze implementiert werden, sodass insgesamt vier Klassen existieren: Personen, Objekte, Tiere und Pflanzen. Durch die neu hinzugefügten Klassen entstehen zwischen Elementen

[4] https://spacy.io/ (25.07.20)

ebenjener Klassen, neue Relationen, die es zu erkennen gilt. Während sich zuvor die Relationen auf (Person, α, Objekt) beschränkten, kommen nun einige neue dazu, wie in:

»Hygieia standing right, feeding serpent held in right arm from patera in outstretched left hand. Ground Line.«[5]

annotiert als das Tripel (Hygieia, *feeding, serpent*) mit Hygieia ∈ Personen und *serpent* ∈ Tiere. Mehr Entitätskombinationen bedeuten in diesem Fall, dass das aus der Person-Objekt-Relation entstandene Klassifikationsproblem überarbeitet werden muss (siehe **Kapitel 4.3.2**). Die Frage, die hierbei zu klären ist, ist die des Berechtigungsdaseins von P. Klinger eingeführten Klassen, für das Klassifikationsproblem bzw. die der notwendigen Erweiterungen. Durch die neue Ausgangslage für das NER und RE, muss das Thema Performance erneut analysiert werden – dies umfasst das Beobachten als auch Optimieren der Vorhersagen des NLPs.

II. Eine neue Erweiterung, die in dieser Arbeit hinzukommt, ist ein Modell, das sich auf das Erkennen von Entität-Verb-Beziehungen fokussiert. Da dies im Fall der Münzbeschreibungen häufiger vorkommt, ist es nützlich, über Tripel- Relationen hinaus auch an Entitäten geknüpfte Verben, ohne ein weiteres Objekt erkennen und annotieren zu können. Dies soll Suchanfragen in der Form von (NE_1, α) mit α ∈ Verben ermöglichen.

III. Der zweite Teil dieser Arbeit befasst sich mit der Anwendbarkeit des NLP-Modells auf eine weitere Sprache. Da die Ikonographen der CNO- Datenbank sowohl in englischer als auch deutscher Sprache zur Verfügung stehen, wird die Übertragbarkeit auf den deutschen Datensatz angewendet. Beobachtet werden hierbei die Schwierigkeiten und Herausforderungen, die mit dem Übertragen auf

[5] https://www.corpus-nummorum.eu/coins?id=824 (25.07.20)

die deutsche Sprache einhergehen. Weiter ist die Performance des Modells in den jeweiligen Sprachen zu vergleichen.

IV. Zusätzlich ist das Thema der Datenqualität zu klären. Davon umfasst ist die Frage, in welcher Form die Daten vorliegen müssen. Eng mit diesem Gesichtspunkt verknüpft wird im Anschluss die Übertragbarkeit dieses Modells auf andere Numismatik-Datensätze an Beispielen demonstriert.

Struktur Im **zweiten Kapitel** wird das Webportal CNO, das als Datensatzgrundlage für diese Arbeit gilt, vorgestellt. In **Kapitel drei** wird auf die Grundlagen eingegangen, um ein notwendiges Verständnis über die genutzten Methoden der IE zu vermitteln. Dabei werden insbesondere die Methoden NER und RE der IE und ihre Funktionsweisen im Allgemeinen näher beleuchtet. Im letzten Abschnitt des dritten Kapitels werden darüber hinaus die grundlegenden Unterschiede der englischen und deutschen Sprache vermittelt. Diese werden für das fünfte Kapitel benötigt. Im **vierten Kapitel** wird die Umsetzung des NLPs, Erweiterungen des NERs und REs auf die englischsprachigen Datensätze gezeigt. Die vorliegende Datenqualität wird diskutiert und auf die dabei aufgetretenen Probleme und Herausforderungen inklusive ihrer Lösungsansätze eingegangen. Zusätzlich wird eine neue (NE, Verb) – Erweiterung präsentiert. Beendet wird das Kapitel mit einer Evaluierung des englischsprachigen Modelles. Das **fünfte Kapitel** beschäftigt sich mit dem Modell für die deutsche Sprache. Die Umsetzung des angepassten Modells wird vorgestellt. Die Datenqualität des deutschen Datensatzes wird analysiert und hierbei auftretende Herausforderungen werden mit Lösungsansätzen diskutiert. Auch dieses Modell wird evaluiert und im Anschluss wird noch die (NE, Verb) – Erweiterung auf den deutschen Datensatz ausgeführt. In **Kapitel sechs** wird aus der Erkenntnis des vorherigen Kapitels eine Bewertung bezüglich der Übertragbarkeit des Modells auf mehrsprachige Datensätze abgegeben. Im **siebten Kapitel** werden die wichtigsten Ergebnisse zusammengefasst und ein Ausblick für zukünftige Forschungen gegeben.

2. Das numismatische Webportal

2.1 Corpus Nummorum Online

Corpus Nummorum Online[6] (CNO), vorher *Corpus Nummorum Thracorum* (CNT) ist ein Webportal, das Münzen aus Moesien, Thrakien, Mysien und Troas erfasst. Die ursprüngliche Grundlage bildeten Münzen des Berliner Münzkabinetts und Gipsabdrücke von Münzen der Berlin-Brandenburgischen Akademie der Wissenschaften. Über die Zeit wurde die Sammlung nach und nach ergänzt und umfasst heute über 100 verschiedene Sammlungen aus 25 Ländern[7]. Dabei ist die Erstellung einer numismatischen Typologie das Ziel des Projektes. CNO ermöglicht die wissenschaftliche Klassifizierung der einzelnen Münzen. Münzen werden in Typen gruppiert, bei denen einige sogar in Subtypen geordnet werden können. Die Ikonographen der Münzen wurden nach standardisierten Kriterien[8] in menschlicher Sprache verfasst. In englischer als auch deutscher Sprache stehen diese zur Verfügung. Laut Datenbankdump vom 5. Januar 2020 besteht der Datensatz aus 5542 ikonographischen Beschreibungen, welche in einer MySQL Datenbank eingelagert wurden.

Um die Form der Ikonographen augenscheinlich zu machen, betrachte man folgende Beschreibung als Beispiel:

> »Artemis standing facing, head left, holding patera in outstretched right hand, left resting on long torch, with quiver over shoulder; hound at side to left; club to right.«[9]

> »Artemis stehend von vorn, Kopf nach links, in der vorgestreckten Rechten Patera und in der Linken lange Fackel haltend, Köcher am Rücken; zu ihren Füßen Hund nach links und Keule nach rechts.«

[6] https://www.corpus-nummorum.eu/about (01.08.20)
[7] https://www.corpus-nummorum.eu/collections (01.08.20)
[8] https://www.corpus-nummorum.eu/pdf/ExternalCoinEntry.pdf (01.08.20)
[9] https://www.corpus-nummorum.eu/coins?id=3531 (01.08.20)

Die Regelung für die Ikonographen besagt, dass Personen und Personifikationen die ersten Elemente einer Münze sind, die es zu identifizieren gilt. Gefolgt von ihrer Kleidung und ihrer Haltung. Wenn die abgebildete Person mit Objekten o. Ä. interagiert, sei es in Form von »etw. haltend«, wird als erstes die rechte, dann die linke Hand beschrieben. Mit Kommas wird die Identifikation, die beschriebene Kleidung und die Richtung, in die die Person sitzt oder steht, getrennt. Folgende Form wird als »*Figure types*« angegeben:

(Nude) / hair style/ figure / verb / orientation/ place/ head orientation/ clothing / attribute right to left / field / (ground line.) / border.

Semikolons werden hierbei verwendet, um die Beschreibung von individuellen Figuren bzw. Elementen zu separieren.

Aktuell ist es möglich, mit Hilfe des *Coin Advanced Search*, nach bestimmten Schlüsselwörtern zu filtern und zu suchen. Die Auswahl der möglichen Schlüsselwörter ist dabei durch ein *Drop Down* Menü eingeschränkt.

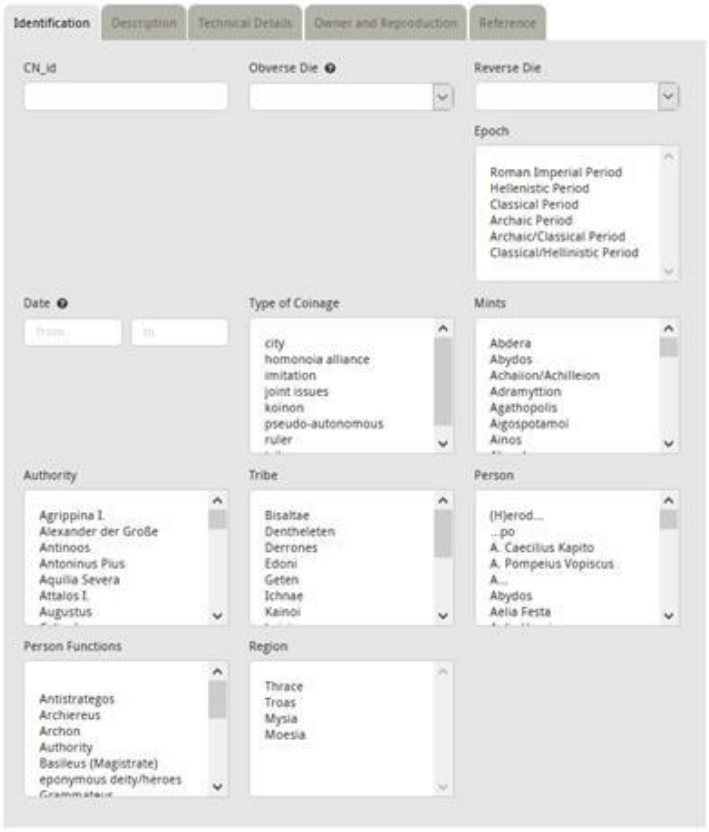

Abbildung 1: Coin Advanced Search[10]

[10] https://www.corpus-nummorum.eu/advancedSearch (01.08.20)

Von den manuell extrahierten Schlüsselwörtern, welche in der relationalen Datenbank mitgeführt werden, sind aktuell nicht alle im *Coin Advanced Search* auszuwählen. Aus dem Beispielsatz von oben wären die Schlüsselwörter – *Artemis, patera, torch, quiver, hound* und *club*, nicht in der Suchauswahl-Maske vorzufinden.

2.2 Verwandte Arbeiten

Online Coins of the Roman Empire (OCRE)[11] ist ein Webportal, das sich ausschließlich mit reichrömischen Münzprägungen beschäftigt. Ins Leben gerufen wurde sie als Gemeinschaftsprojekt der *American Numismatic Society* und *des Institute for the Study of the Ancient World* der *New York University*. Heute gehört unter anderem noch das Münzkabinett der Staatlichen Museen von Berlin zu den Kooperationspartnern. Das Projekt umfasst alle jemals veröffentlichten römischen Münzen von Kaiser Augustus 31 v. Chr. bis zum Tod Zenos 491 n. Chr. Von den insgesamt 100.000 Münzprägungen, sind circa 50% auf der Plattform vertreten.

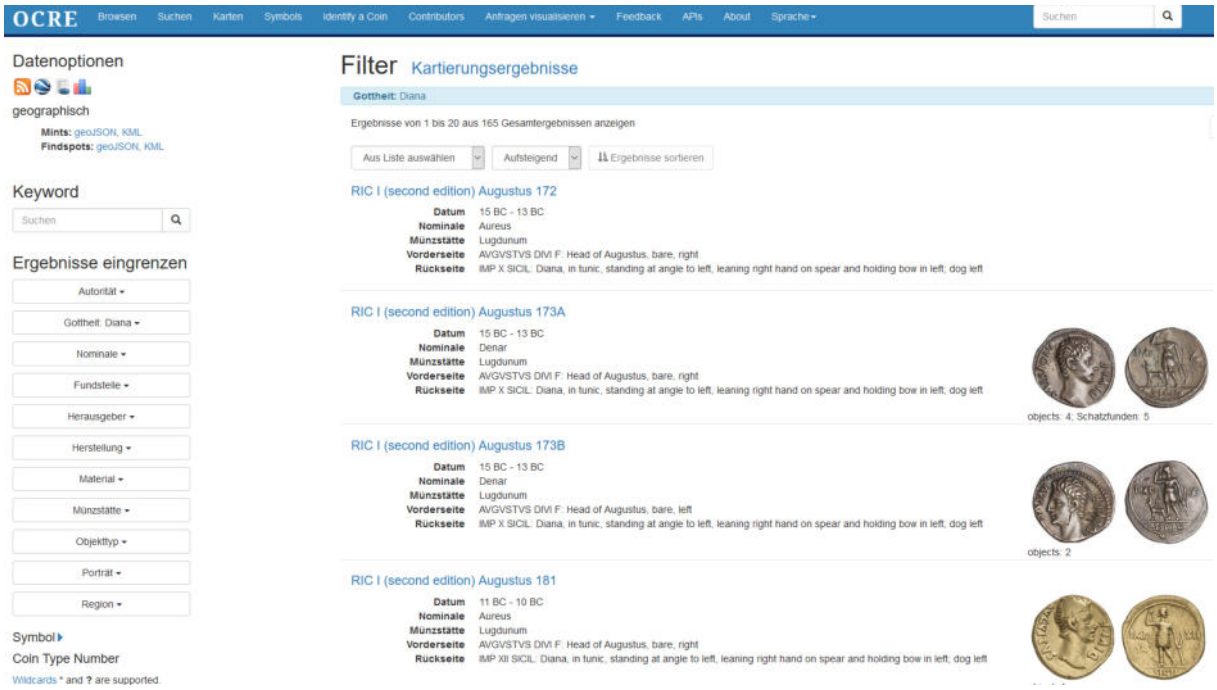

Abbildung 2: OCRE Suche – Auswahl der Eingrenzung durch Drop Down Menü auf der linken Seite[12]

[11] http://numismatics.org/ocre/?lang=de (01.08.20)
[12] http://numismatics.org/ocre/results?q=deity_facet%3A%22Diana%22&lang=de (01.08.20)

Genau wie in der CNO Suche ist es in OCRE möglich, nach Schlüsselbegriffen zu suchen bzw. die Suche einzugrenzen. Statt verallgemeinert nach Personen zu suchen, bietet OCRE die zusätzliche Spezifikation nach Autoritäten, Gottheiten oder Nominalen zu suchen. In Abbildung 2 ist die Suchanfrage nach Münzen mit der abgebildeten Gottheit Diana (Artemis in der römischen Mythologie) zu sehen. OCRE bietet hierbei besonders ausführliche Informationen zu Fundortverteilungen der einzelnen Münzprägungen.

Im letzten Kapitel dieser Arbeit wird im Rahmen der Evaluierung der Übertragbarkeit die Datensätze der OCRE Datenbank mit Implementierungen dieser Arbeit begutachtet (siehe **Kapitel 6**).

3. Grundlagen

In diesem Kapitel wird ein Überblick über die angewandten Methoden und nötigen Hintergrundwissen verschafft. Dabei handelt es sich in erster Linie um Grundlagen des maschinellen Lernens sowie um die genutzten Klassen aus der Pythonbibliothek spaCy[13]. Da der zweite Teil dieser Arbeit eine neue Sprache für das Modell vorsieht, ist auch ein gewisses linguistisches Vorwissen abzudecken.

Abschnitt 3.1 beschäftigt sich mit den Arten des maschinellen Lernens und einem der Grundprobleme dieser Arbeit; der Klassifikation. Kapitel 3.2 geht auf NLP und den Bereichen NER und RE der IE ein. In 3.3 werden die Metriken vorgestellt, die genutzt werden um Modelle dieser Art zu evaluieren. Anschließend wird in Kapitel 3.4 die Python Bibliothek *spaCy* vorgestellt. Letztendlich wird in Kapitel 3.5 das nötige Grundwissen über die sprachlichen Unterschiede der englischen und deutschen Sprache vermittelt, auf welches ab **Kapitel 5** eingegangen wird.

3.1 Maschinelles Lernen

Maschinelles Lernen bzw. *Machine Learning* als Teilgebiet der künstlichen Intelligenz hat sich zum Ziel gesetzt, selbständig Informationen anhand einer Menge von Daten zu

[13] https://spacy.io/ (13.09.20)

gewinnen. Man könnte auch davon sprechen, »künstliches Wissen aus Erfahrung« zu generieren.[14] Dabei gibt es verschiedene Arten des maschinellen Lernens (Brownlee 2016, Kap. 5).

Supervised Beim überwachten Lernen (*supervised learning*) wird ein Trainingsdatensatz übergeben, der bereits korrekt annotiert ist bzw. dessen Ausgabe bekannt ist. Ist keine korrekte Annotation bzw. Ausgabe vorhanden, muss diese manuell erstellt werden. Der manuelle Prozess kann sehr viel Zeit in Anspruch nehmen. Das Ziel hierbei ist es, das Modell anhand der Trainingsdaten zu optimieren, so dass es auf neue Daten eingesetzt werden und vorhersagen getroffen werden können. Das Modell wird dabei solange optimiert bis es ein zufriedenstellendes Ergebnis erzielt.

Unsupervised Beim unbewachtem Lernen (*unsupervised learning*) ist, im Gegensatz zum überwachten Lernen, die Ausgabe nicht bekannt. Das gewählte Modell entscheidet und lernt selbst die Struktur der Daten und versucht Erkenntnisse zu extrahieren. Ein Problem ist das *Clustering*, dabei wird versucht die Daten in Gruppen mit Ähnlichkeiten zusammenzufassen. Ein weiteres Problem ist das Assoziieren bei dem versucht wird Folgerungen zu erkennen.

Semivised Beim halbüberwachten Lernen besitzt nur ein Teil des Datensatzes bereits die korrekte Ausgabe. Hier versucht das Modell anhand der kleineren, bereits annotierten Mengen, Schlussfolgerungen und Erkenntnisse auf den Rest der Daten zu gewinnen.

3.1.1 Klassifikation

Bei der Klassifizierung wird eine bestimmte Eingabe einer bestimmten Klasse zugeordnet, die Anzahl der Klassen ist hierbei endlich (Bishop 2006, Kap.1). Ein Beispiel wäre hierfür die Erkennung der Bedeutung des Wortes »Ball« im jeweiligen Kontext. Es könnte der »Ball« zum Spielen gemeint sein oder aber auch eine Tanzveranstaltung, dies wäre ein

[14] https://www.bigdata-insider.de/was-ist-machine-learning-a-592092/ (13.09.20)

Klassifikationsproblem mit zwei Klassen. Diese Arbeit behandelt ein Multiklassenproblem, daher könnte eine Eingabe auch mehreren Klassen zugeordnet werden (Bird, Klein und Loper 2009, Kap. 6.1). Für die Klassifikation wird ein Trainingsdatensatz genutzt dessen Ausgabe bereits bekannt ist, die Klassifikation ist daher Teil des überwachten Lernens. Der Trainingsdatensatz muss daher bereits annotiert sein, dies wird für den verwendeten Datensatz manuell getätigt. Die Trainingsdaten werden anschließend in zwei Datensätzen aufgeteilt, der erste Datensatz wird benutzt, um das Modell zu trainieren, der zweite wird genutzt, um das nun trainierte Modell zu nutzen und Vorhersagen zu treffen, die anschließend evaluiert werden (Bishop 2006, Kap. 1). Um das Klassifikationsproblem zu lösen gibt es verschiedene Klassifikator. Eine sehr gängige Methode ist die logistische Regression (LR). Diese hat sich in der Arbeit von P. Klinger auch als sehr effizient erwiesen. Weitere Klassifikator, die genutzt werden sind *Support Vector Machines* (SVM) und *Random Forest Classifer* (RFC).

3.1.2 Classifier

Logistische Regression Die logistische Regression wird genutzt, wenn die Zielvariable kategorisierbar ist und wird meistens für ein Zweiklassenproblem eingesetzt[15], wie bspw. die Kontexterkennung des Wortes »Ball«. Das Ergebnis zu welcher Klasse die Eingabe gehört wird als 0 oder 1 bestimmt. Dies wird durch die Sigmoidfunktion berechnet. Die Sigmoidfunktion hat einen S-förmigen Graphen, mit Werten zwischen 0 und 1. Abhängig von einem Schwellenwert werden dann die Werte des Graphen als 0 oder 1 klassifiziert. Der Schwellenwert ist standardmäßig auf 0,5 gesetzt, dies bedeutet das Werte im Graphen die größer oder gleich 0,5 sind, den Wert 1 erhalten bzw. 0, wenn die Werte kleiner sind. Die Funktion nimmt dabei einen Wert entgegen und berechnet die Wahrscheinlichkeit des Wertes, dass dieser in der Klasse liegt. Die Wahrscheinlichkeit, dass die Eingabe in der anderen Klasse ist, erhält man durch die Berechnung der Gegenwahrscheinlichkeit.

[15] https://www.datacamp.com/community/tutorials/understanding-logistic-regression-python (15.09.20)

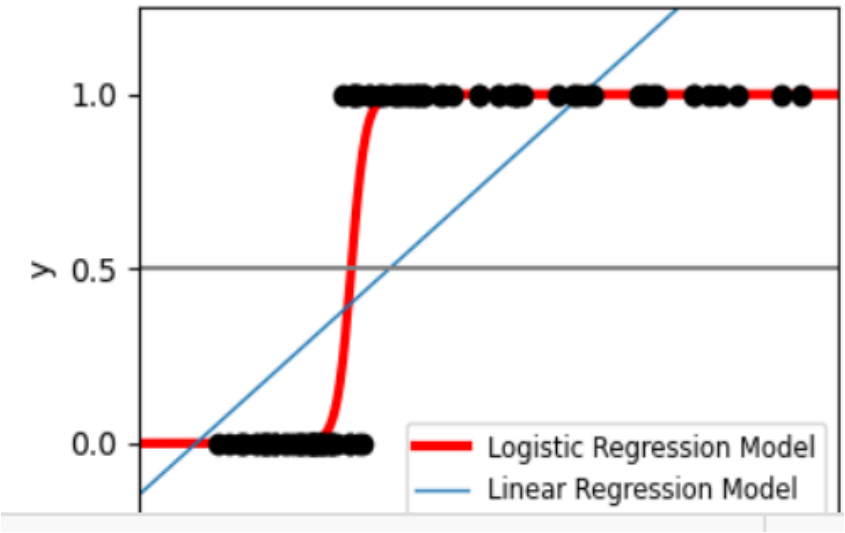

Abbildung 3: logistic curve[16]

Die logistische Regression kann aber auch für Klassifikationsprobleme mit mehr als zwei Klassen benutzt werden. Um nun ein Problem mit mehr als zwei Klassen zu lösen und zu klassifizieren gibt es mehrere Varianten. Für die hier genutzten Klassifikator aus der Python Bibliothek *scikit-learn* sind folgende drei relevant:

I. One-vs-Rest
II. One-vs-One
III. Softmaxfunktion

Bei der *One-vs-Rest* Strategie werden mehrere Klassifizierungssets erstellt und jede Klasse wird gegen die Menge der anderen Klassen trainiert. Wenn es die Klassen »PERSON«, »OBJECT«, und »ANIMAL« gibt werden folgende Klassifizierungsprobleme erstellt:

PERSON vs. [OBJECT, ANIMAL]

OBJECT vs. [PERSON, ANIMAL]

ANIMAL vs. [PERSON, OBJECT]

[16] https://scikit-learn.org/stable/auto_examples/linear_model/plot_logistic.html (13.09.20)

Dabei wird für jedes Problem ein Modell erstellt. Eine andere Möglichkeit ist *One-vs-One* Strategie, hierbei wird jede Klasse gegen eine andere Klasse trainiert, mit den obigen Klassen würde dies folgendermaßen aussehen:

PERSON vs. OBJECT

PERSON vs. ANIMAL

OBJECT vs. ANIMAL

Es wird jedes Modell ausgewertet und die Vorhersage wird anhand der Anzahl der Entscheidungen, für die jeweilige Klasse, festgemacht[17]. Das bedeutet entscheiden sich zwei Modelle für die Klasse PERSON, so wird diese als Ergebnis gewählt. Die dritte Variante wird durch die Softmaxfunktion umgesetzt, diese wird auch als Standard für die logistische Regression mit Multiklassenproblemen genutzt[18]. Die Softmaxfunkton nimmt im Vergleich zur Sigmoidfunktion nicht einen Wert, sondern einen Vektor entgegen und berechnet daher jede Wahrscheinlichkeit für die jeweilige Klasse, die Summe der Wahrscheinlichkeiten ergibt 1. Die Sigmoidfunktion kann für die anderen beiden Strategien verwendet werden, da der Vergleich nur jeweils zwischen zwei Klassen berechnet wird.

Support Vector Machine *Support Vector Machine* (SVM) wird ebenfalls dazu eingesetzt ein Zweiklassenproblem zu lösen. Eine SVM bestimmt die Zuordnung der Klasse anhand einer Trenngeraden. Es werden *Cluster* gebildet und zwischen diesen wird eine Trenngerade berechnet, diese soll dabei die Eigenschaft haben den längst möglichen Abstand zu dem nächst möglichen Punkten der *Cluster* zu besitzen. Für Multiklassenprobleme nutzt die Python Bibliothek *scikit-learn* die *One-Vs-One* Variante als Standard[19].

Random Forest Classifier *Random Forest* kann sowohl für die Klassifikation als auch Regression genutzt werden und kann ohne Anpassungen direkt für Multikassenprobleme eingesetzt werden (Hänsch und Hellwich 2015, Kap. 1-2). Ein *Random Forest* Modell

[17] https://machinelearningmastery.com/one-vs-rest-and-one-vs-one-for-multi-class-classification (18.09.20)
[18] https://scikitlearn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html (11.09.20)
[19] https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html (11.09.20)

besteht aus vielen Entscheidungsbäumen, dabei sagt jeder Baum eine Klasse voraus und die Klasse mit der größten Anzahl an Vorhersagen aller Bäume wird vom Modell gewählt[20]. Die Bäume treffen ihre Entscheidungen zufällig, somit haben die Bäume zueinander eine geringe Korrelation (Brownlee 2016, Kap. 17). Eine geringe Korrelation führt dazu, dass die Genauigkeit der Vorhersage zunimmt, haben die Bäume eine hohe Korrelation hat dies zur Folge das alle zur gleichen Entscheidung tendieren bzw. das der *Random Forest* nur aus ähnlichen Entscheidungsbäumen besteht (Hänsch und Hellwich 2015, Kap. 1). Außerdem führt eine niedrige Korrelation dazu, dass sich die Bäume gegenseitig vor Fehlern schützen und sich gegenseitig ausgleichen, was die Genauigkeit der Vorhersage erhöht. Das wird dadurch erreicht das zum einen jeder Entscheidungsbaum eine andere Verteilung der Eingabe erhält, die Größe dieser Verteilung entspricht dabei die der Eingabe und zum anderen jeder Entscheidungsbaum die Merkmale, die er nutzt, ebenfalls zufällig wählt[20].

Entscheidungsbäume besitzen eine Wurzel und beliebig viele innere Knoten sowie mindestens zwei Blätter. Ein Entscheidungsbaum besteht dabei aus Fragen bzw. Tests und Antworten. Die inneren Knoten repräsentieren dabei die Fragen und Blätter die möglichen Antworten auf das zu lösende Problem (Hänsch und Hellwich 2015, Kap. 1.2).

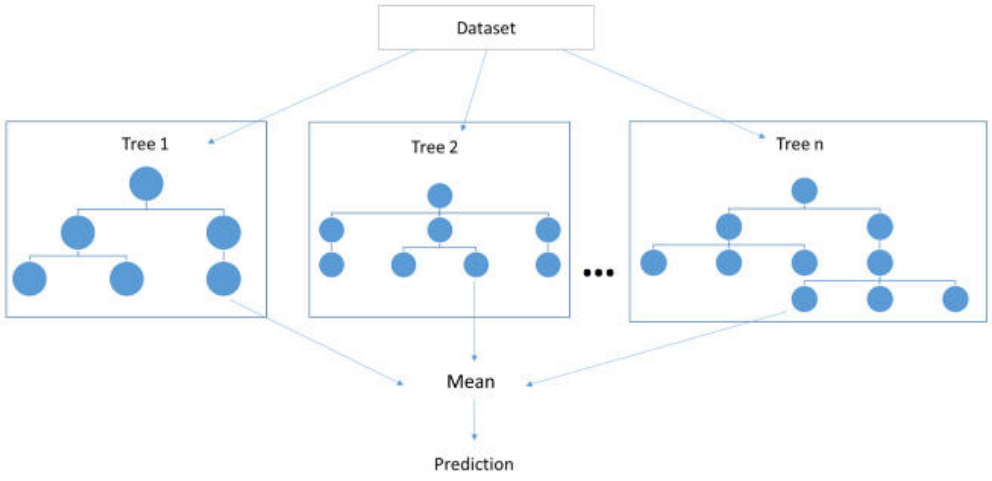

Abbildung 4: N Entscheidungsbäume führen zur Vorhersage des Modells[21]

[20] https://towardsdatascience.com/understanding-random-forest-58381e0602d2 (13.09.2020)
[21] https://towardsdatascience.com/random-forests-and-decision-trees-from-scratch-in-python-3e4fa5ae4249 (13.09.2020)

Mehr zu den Möglichkeiten für das Klassifizieren findet man unter anderem in der Literatur »Master Machine Learning Algorithms« von J. Brownlee (2016).

3.2 Natural Language Processing

Natural Language Processing (NLP) beschäftigt sich mit der Erfassung, dem Analysieren und Verarbeiten einer natürlichen Sprache, so dass der Computer dieses Wissen nutzen kann, um zu reagieren bzw. antworten. Im Vergleich zu einer Programmiersprache wie Python, dessen Eingaben und Reaktionen festgelegt sind, sind diese in einer natürlichen Sprache nicht klar definiert und hängen vom Kontext bzw. dem Entwickler ab. Das heißt der Computer muss den Kontext erkennen können und anhand dessen die Entscheidung treffen und nicht nach einem festgelegten Muster, wie es in einer Programmiersprache üblich ist (Hobson, Howard und Hapke 2019, Kap. 1.1). Ziel ist es daher ein Modell zu entwickeln, das für das angewendete Gebiet die Eingabe erfassen, interpretieren und eine gewünschte Reaktion zurückgeben kann. Eine natürliche Sprache wird nicht direkt vom Computer verstanden, um dies erst zu ermöglichen durchläuft das NLP verschiedene Schritte der Transformation und Berechnung, um die Eingabe einer natürlichen Sprache und die gewünschte Ausgabe zu erhalten. Dieser Prozess wird als *Pipeline* bezeichnet, dabei geht es darum die wichtigen Informationen aus dem gegebenen Kontext zu extrahieren und verwendbar zu machen (Hobson, Howard und Hapke 2019, Kap. 1.1- 1.2.1).

Bei dieser Arbeit ist es das Ziel, bestehende Relationen zwischen Entitäten in numismatischen Beschreibungen zu erkennen bzw. extrahieren. Dies wird in zwei Schritten unterteilt NER und RE. Die Umsetzung dieser beiden Schritte wird mithilfe des Python Moduls *spaCy* umgesetzt, mehr dazu im **Kapitel 3.4**.

Named Entity Recognition Eine Methode, um Informationen zu extrahieren ist die Eigennamenerkennung (Hobson, Howard und Hapke 2019, Kap. 11.1). Mit dem Ziel bestehende Relationen zu erkennen, muss erstmal erkannt werden um welche Entitätstypen es sich handelt. Die zu erkennenden Entitätstypen hängen dabei natürlich auch von der gewünschten Umgebung bzw. Absicht ab. In dieser Arbeit werden die folgenden Entitätstypen definiert: »PERSON«, »OBJECT«, »ANIMAL« und »PLANT«. Für das RE ist es daher wichtig diese Entitätstypen korrekt zu erkennen.

»Nude Apollo **PERSON** advancing right, holding arrow **OBJECT** and drawing bow **OBJECT** in his left hand.«[22]

Im obigen Beispielsatz kann man sehen wie die Ausgabe eines NER aussieht. Damit die selbst erstellten Entitätstypen erkannt werden, wird zunächst ein Datensatz mit einer korrekten Zuordnung als Trainingsgrundlage genutzt, die korrekte Zuordnung findet automatisiert statt. Die Grundlage der Zuordnung, welche Elemente in welche Klasse gehören, findet jedoch manuell statt. Es werden Tabellen erstellt, die Elemente der jeweiligen Klasse enthalten. Die erstellten Tabellen müssen für ein verbessertes Training und ständigem Lernen mit neuen Elementen erweitert werden, da immer neue Ikonographen in die Datenbank aufgenommen werden und es daher Neues zu erkennen gibt. Mit diesem Grundwissen kann nun das Modell lernen und für neue Daten eingesetzt werden. Das trainierte Modell wird mithilfe der Grundlage neue Elemente auf den neuen Daten vorhersagen und einer Klasse zuordnen. Die Auswertung dieser Vorhersagen findet manuell statt, dies hilft dabei den Datensatz zu vergrößern, das heißt richtige Vorhersagen werden in die Datenbank aufgenommen.

Das NE bildet hier auch die Grundlage für das RE, da erst die Entitäten erkannt werden müssen um anschließend die Relation dieser zu erkennen.

Relation Extraction Das RE befasst sich mit der Aufgabe Relationen in einem Text zwischen mehreren Entitäten zu erkennen, daher wie zwei oder mehrere Entitäten einen Bezug zueinander haben. Im oben erwähnten Beispiel ist zu erkennen das Apollo einen Pfeil hält bzw. einen Bogen zieht.

Ziel dieser Arbeit ist es zu untersuchen welche Beziehungen zwischen den einzelnen Entitäten existieren und zwar zwischen dem Subjekt und Objekt des Satzes. Es wird daher das Tripel (Subjekt, Relation, Objekt) gesucht, wichtig ist dabei die Reihenfolge, da unterschiedliche Beziehungen zwischen den Klassen existieren, abhängig ob eine Person das Subjekt und ein Gegenstand das Objekt ist oder vice versa. Betrachtet man das Tripel (*Apollo, holding, sword*), besteht die Beziehung aus »Apollo hält ein Schwert«. Würde die

[22] DesignID = 38

*Alle zukünftig genannten Ikonographen sind im Anhang beigefügt.

Reihenfolge nun keine Rolle spielen, so könnte man auch die Beziehung »Schwert hält Apollo« herleiten, was wiederrum falsch wäre, da solch eine Beziehung nicht vorkommt.

In dieser Arbeit wird der Ansatz von P. Klinger fortgeführt, sie setzt die Relationserkennung als Multiklassenproblem um. Der Gedanke dabei ist, sich zu fragen, ob eine gewisse Relation zwischen einem Paar, bestehend aus einer PERSON als Subjekt und einem OBJECT als Objekt, besteht oder nicht (Klinger 2018, Kap. 4.2). In dieser Arbeit wird dieses Wissen nun genutzt, um die Relation aller Paare zwischen den erstellten Klassen zu extrahieren. Der Trainingsdatensatz wird dabei manuell erstellt, dabei wird ein Teil der Sätze aus der Datenbank genommen und es werden alle Relationen pro Satz annotiert. Ziel ist es Tripel in der Form (Subjekt, Relation, Objekt) zu erhalten, sodass es im *Relation Description Format* RDF ist (Hobson, Howard und Hapke 2019, Kap. 11.4).

Der Prozess, um eine Relation zu extrahieren besteht aus mehreren Schritten. Der erste Schritt ist es, wie bereits erwähnt, Entitäten zu erkennen. Danach muss die Information zwischen diesen ermittelt werden, das heißt welches Verb verbindet das betrachtete Subjekt bzw. Objekt miteinander. Dieses Resultat wird dann an den Klassifikator weitergegeben, um dies vorzubereiten werden Methoden benötigt, um die Verbindung bzw. den Pfad zwischen den betrachteten Entitäten zu ermitteln und anschließend Methoden, die diese Information in einem Vektor umwandeln.

Der Pfad zwischen zwei Wörtern in einem Satz wird ermittelt, indem, angefangen bei einem Wort, der letzte Vorfahre des Wortes gesucht wird und anschließend dasselbe für das andere Wort. Besitzen beide Wörter denselben Vorfahren so haben diese eine Verbindung bzw. Pfad zueinander. Dies Erkennung des Pfades wird mithilfe von *spaCys DependancyParser* umgesetzt (siehe **Kap. 3.4**).

Folgende Methoden werden, wie bereits in der Grundlage, für die Erstellung der Feature genutzt (Klinger 2018):

Verb2Str fügt jedes Verb, das im Dokument vorkommt, als Feature hinzu, die Verben werden durch den PoS-Tag »Verb« bestimmt. In der Praxis hängt dies stark von spaCys PoS-Tagger ab, oft werden Verben bei den Ikonographien als Nomen markiert, wie zum Beispiel im Satz »Dolphin swimming upwards.«[23] (siehe **Kapitel 4.4**).

[23] DesignID = 423

Path2Str nutzt den gefundenen Pfad zwischen zwei Entitäten und wandelt diesen in einen String um, außerdem können zwei Parameter gewählt werden pos (*Part of Speech*) und dep (Dependency) welche an jedem Element des Pfades hinzugefügt werden können. Diese Information wird durch spaCy zur Verfügung gestellt, mehr dazu und wie diese Information abrufbar ist in **Kapitel 3.4**. Es folgt Beispiel, um die Transformation zu verdeutlichen. In der Ikonographie

> »Ares standing right, wearing helmet, holding spear and shield set down at his feet.«[24]

wird zwischen »Ares« und »*helmet*« folgender Pfad erkannt:

[Ares, standing, wearing, helmet]

Fügt man den PoS-Tag hinzu so wird der Pfad folgendermaßen extrahiert:

Ares\PROPN standing\VERB wearing\VERB helmet\NOUN

Doc2Str wandelt das gesamte Dokument in einen String um, um diesen anschließend zu vektorisieren. Dieses Feature nutzt nicht das NER, daher ist keine hohe Performance zu erwarten (Klinger 2018).

Um die Daten nun für das maschinelle Lernen vorzubereiten, da Strings bzw. die Ikonographen in verschiedenen Größen vorkommen, müssen diese nun in einen Vektor umgewandelt werden. Für diesen Ansatz sind zwei Methoden weit verbreitet[25]. Die erste Variante nennt sich »Bag-of-Words« (BoW). Diese Methode erstellt für ein Dokument ein Vokabular bestehend aus den einzigartigen Worten des Dokumentes. Anschließend wird dieses angewendet, um das Auftreten der vorhandenen Elemente des Vokabulars in einer

[24] DesignID = 4200
[25] https://medium.com/@sewwandikaus.13/bow-vs-tf-idf-in-information-retrieval-a325b5e61984 (07.11.20)

Eingabe zu ermitteln und als Vektor auszugeben. Um dies zu verdeutlichen wird dies anhand eines Beispiels gezeigt:

I. »Head of Athena, right, wearing helmet.«

II. »Ares standing right, wearing helmet.«

Aus den beiden Ikonographen wird folgendes Vokabular erstellt:

{Head, of, Athena, wearing, helmet, Ares, standing, right}

Danach wird dieses auf die Ikonographen angewendet und es entsteht ein Vektor, der das Aufkommen des Vokabulars im Satz darstellt.

	Ares	Athena	Head	helmet	of	right	standing	wearing
I.	0	1	1	1	1	1	0	1
II.	1	0	0	1	0	1	1	1

Tabelle 1: Bag-of-Words

Zu beachten ist das die BoW Variante den Nachteil hat, dass die Reihenfolge im Satz nicht beachtet wird und das die unterschiedlichen Seltenheiten des Aufkommens ebenfalls keinen Einfluss haben[23].

Die BoW Variante wird in dieser Arbeit mit der Klasse *CountVectorizer* aus der Python Bibliothek *scikit-learn* umgesetzt. Diese bringt einen weiteren Parameter mit, der die diese Variante nochmal erweitert. Durch den Parameter ngram ist es möglich festzulegen wie viele Wortkombinationen möglich sind. In der obigen Tabelle kommen nur einzelne Wörter vor (unigram), wird der Parameter auf 1,2 gesetzt so sind zusätzlich auch Bigramme erlaubt. Das heißt, es werden Kombinationen wie »Ares standing«, »standing right« oder »wearing helmet« in das Vokabular mit aufgenommen. Die Nutzung von Bigrammen oder auch Trigrammen (bestehend aus drei Wörtern) hilft es den Kontext zusammenzuhalten, da wie bereits erwähnt, die Reihenfolge verloren geht.

Eine weitere Variante ist TF-IDF, dies steht für *term frequency times inverse document frequency*. Bei dieser Variante wird zunächst ebenfalls der obige Ansatz getätigt

und das Auftreten der Wörter ermittelt, aber das Auftreten wird nochmal mit der Häufigkeit dieser gewichtet, sodass häufig aufkommende Wörter einen kleineren Einfluss ausüben. Dies geschieht dadurch, dass der Wert logarithmiert wird. Folgende Formel wird benutzt $Log\left(\frac{n}{df(t)}\right) + 1$, dabei wird der Logarithmus der Gesamtanzahl der Dokumente(n) durch die Anzahl der Dokumente, die das Wort beinhalten(df(t)) berechnet +1[26]. Die +1 ist nötig wenn ein Wort in jedem Dokument auftritt, da man damit einen Log(0) haben würde. Diese Variante wird mit der Klasse *TfIdfVectorizer* implementiert, ebenfalls aus der Bibliothek *scikit-learn*[27].

3.3 Metriken

Nachdem ein Modell erstellt worden ist, muss betrachtet werden wie gut das Modell arbeitet. Um ein Modell zu evaluieren stehen verschiedene Metriken zur Verfügung. Die Ergebnisse der Evaluation sind unteranderem dafür wichtig, um Modifikationen am Modell vorzunehmen und diese dann zu vergleichen, aber auch dafür, dass gezeigt wird wie vertrauenswürdig es ist (Bird, Klein und Loper 2009, Kap. 6.3). Eine Evaluation findet anhand eines Testdatensatzes statt, dieser besitzt eine korrekte Annotation und entspricht demselben Format wie dem des Trainingsdatensatzes. Der Testdatensatz wird nicht als Training für das Modell genutzt, daher hat das Modell keine Informationen darüber und trifft anhand diesem seine Vorhersagen, die dann evaluiert werden können.

$$\text{Accuracy} = \frac{\#correctPrediction}{\#totalPrediction}$$

Accuracy Die Genauigkeit stellt eine simple Metrik dar, diese wird durch das korrekte Vorhersagen des Modells berechnet, die korrekten Vorhersagen werden dabei anhand der existierenden Annotationen überprüft. Existiert das Objekt Schwert 100 Mal im

[26] https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfTransformer.html#sklearn.feature_extraction.text.TfidfTransformer (07.11.20)
[27] https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html (07.11.20)

Testdatensatz und das Modell erkennt 80 davon, so hat das Modell eine Genauigkeit von 80/100 = 80%. Ein relevanter Einfluss für die Genauigkeit ist, wie oft das Gesuchte vorkommt. Kommt das Objekt zu 50% im betrachteten Umfeld vor, so ist es weniger aussagekräftig als wenn das Objekt zu 10% vorkomme, dementsprechend ist auch die Größe des Training- beziehungsweise Testdatensatzes wichtig. Eine Genauigkeit von 80% des Gesuchten auf 1000 Daten ist natürlich beachtlicher als auf 200 Daten, da die Rate des Auftretens geringer ist. Die Genauigkeit ist daher keine gute Metrik, wenn Klassen existieren, deren Anzahl im Vergleich zu anderen überwiegen, wenn also die Klassen ungleichmäßig repräsentiert sind[28]. Für dieses Problem werden üblich Metriken gewählt die aus der Konfusionsmatrix berechnet werden. Eine Konfusionsmatrix ist eine 2x2 Matrix, die aus Vorhersagen besteht und vier Fälle abdeckt[29].

	True	*False*
Prediction True	True Positive	False Positive
Prediction False	False Negative	True Negative

Tabelle 2: Konfusionsmatrix

Dabei steht *True Positive* (TP) für die richtig vorhergesagten der Positivklasse, *False Positive* (FP) für die falsch vorhergesagten der Positivklasse, *False Negative* (FN) für die falschen vorhergesagten der Negativklasse und *True Negative* (TN) für die richtig vorhergesagten der Negativklasse. Die Konfusionsmatrix enthält dabei die Anzahl der vier Fälle, teilt man jede Zeile der Matrix durch die gesamte Anzahl an Vorhersagen für die jeweilige Reihe, so erhält man eine normierte Matrix, diese enthält dann die Rate der Elemente und kann besser zum Einschätzen der Performance genutzt werden (Tabelle 4)[12].

[28] https://www.saracus.com/blog/performance-metriken-klassifikation-2-2/ (17.09.20)
[29] https://developers.google.com/machine-learning/crash-course/classification/true-false-positive-negative (17.09.2020)

TP	Relevante Elemente, die korrekt als relevant klassifiziert worden sind.
TN	Nicht relevante Elemente, die korrekt als nicht relevant klassifiziert worden sind.
FP	Nicht relevante Elemente, die fälschlicherweise als relevant klassifiziert worden sind.
FN	Relevante Elemente, die fälschlicherweise als nicht relevant klassifiziert worden sind.

Tabelle 3: Erläuterung der möglichen Resultate (Bird, Klein und Loper 2009, Kap. 6.3)

	True	*False*
Prediction True	$\frac{\#\text{True Positive}}{\#total\ Positive}$	$\frac{\#\text{False Positive}}{\#total\ Positive}$
Prediction False	$\frac{\#\text{False Negative}}{\#total\ Negativ}$	$\frac{\#\text{True Negativ}}{\#total\ Negativ}$

Tabelle 4: Normierte Konfusionsmatrix

Aus der Konfusionsmatrix (Tabelle 2) kann man nun die *Precision* und den *Recall* berechnen.

Precision Die *Precision* (Präzision) gibt an wie viele der Vorhersagen, die das Modell als richtig bestimmt hat, korrekt sind (Tabelle 2 *Prediction* True – True). Die Präzision sagt daher wie genau das Modell ist. Eine Präzision von 70% bedeutet, dass wenn das Modell ein Element als Positiv bestimmt, es zu 70% richtig liegt.[30]

$$\text{Precision} = \frac{TP}{TP+FN}$$

Recall Der *Recall* gibt an wie groß der Anteil der TP anhand der gesamten Anzahl der als positiv klassifizierten Elemente TP+FN ist. Der *Recall* gibt daher an wie viele der als positiv klassifizierten Elemente wirklich korrekt sind. Hat man Insgesamt 4 korrekte

[30] https://developers.google.com/machine-learning/crash-course/classification/precision-and-recall (08.12.2020)

Elemente und das Modell erkennt nur eins davon, so ist der Recall 1/4 – 25%, das Modell erkennt also 25% der richtigen Klasse. [30]

$$\text{Recall} = \frac{TP}{TP+FN}$$

F-Maß *Precision* und *Recall* harmonieren nicht gut miteinander, wird die *Precision* verbessert, so sinkt der *Recall* und vice versa. Damit ein gutes Mittelmaß der beiden gefunden wird, wird das F-Maß (auch F genannt) genutzt. Das F-Maß berechnet sich aus *Precision* und *Recall* und bildet eine Ausgewogenheit zwischen den beiden. Das F-Maß kann dabei zwischen 0 und 1 liegen, wobei näher zur 1 eine hohe *Precision* und *Recall* bedeuten.

$$F = \frac{2 * Precision * Recall}{Precision + Recall}$$

Gridsearch Um die beste Kombination zwischen den oben vorgestellten Algorithmen, Methoden und den möglichen Parametern zu ermitteln, wird ein sogenannter *Gridsearch* ausgeführt. Hierbei werden die verschiedenen Algorithmen bzw. Methoden mit ihren jeweils unterschiedlichen Hyperparametern, Parameter die vor dem Training festgelegt werden müssen, getestet und ausgewertet. Damit kann die beste Konstellation für den vorhanden Datensatz gefunden werden[31].

3.4 SpaCy

Als Grundlage für das NLP wird, für die Umsetzung, die Python Bibliothek *spaCy* verwendet. Im folgenden Abschnitt werden Grundlagen erklärt, die für das NLP relevant sind und wie diese in *spaCy* arbeiten. Außerdem werden zwei Klassen vorgestellt, die eine Umsetzung des NER und RE ermöglichen. Für diese Arbeit wird die, zu dieser Arbeit aktuellen, *spaCy* Version 2.3.1 verwendet. Die Unterschiede im Vergleich zur vorher genutzten Version 2.1 werden in diesem Abschnitt behandelt.

[31] https://towardsdatascience.com/grid-search-for-model-tuning-3319b259367e (08.11.2020)

Tokenizer Beim Tokenisieren wird der Satz in seine Einzelteile zerlegt, ein Token ist ein Wort oder ein Satzzeichen des Satzes[32]. Der Satz »Athena mit Patera.« wird in folgende Token geteilt:

Athena

mit

Patera

Part of Speech (POS) Beim Verwenden von POS wird jedem Baustein des Satzes eine Wortkategorie zugewiesen, wie zum Beispiel Nomen, Verb, Adjektiv usw. Die Zuweisung geschieht in Anbetracht der Wortdefinition und im Zusammenhang des Kontextes. Wendet man POS auf den obigen Satz an, nachdem dieser tokenisiert worden ist, so erhalten die Tokens folgende Zuweisung:

Athena	mit	Patera.
PROPN	VERB	PROPN

DependancyParser Durch den *DependancyParser* wird die Struktur des Satzes analysiert, heißt: welche Abhängigkeiten zwischen den einzelnen Bausteinen besteht. Im Anschluss wird ein Abhängigkeitsbaum dazu generiert. Die Baumwurzel stellt dabei das Wort dar der den Satz zusammenhält, also in den meisten Fällen das Verb[33].

Versionsvergleich Das Upgrade von Version 2.1 auf 2.3.1 verspricht folgende Vorteile. Zum einen wurde die Performance verbessert Entitätstypen zu erkennen die ein geringes Aufkommen besitzen. Es wird nun immer dann eine Fehlermeldung ausgegeben, wenn einem Wort mehr als ein Entitätstyp zugeordnet wird. Dies betrifft Wörter, die aus mehreren einzelnen Wörter bestehen, wie zum Beispiel »laurel branch«. Dies wurde in früheren *spaCy* Versionen ignoriert bzw. außer Acht gelassen[34,35]. Das Objekt »laurel

[32] https://spacy.io/api/ (08.11.2020)
[33] https://web.stanford.edu/~jurafsky/slp3/15.pdf (08.11.2020)
[34] https://github.com/explosion/spaCy/issues/3608 (10.09.2020)
[35] https://spacy.io/usage/v2-3 (10.09.2020)

branch« besteht aus zwei Worten »laurel« und »branch«. Betrachtet werden soll aber nur das gesamte Wort »laurel branch« und nicht dessen Teilworte. Setzt sich also ein Wort aus mehreren Worten zusammen, so wird für das NLP nur noch der gesamte Begriff an *spacys EntityRecognizer* weitergeben. Dies geschieht dadurch, dass während des Lernprozesses die erkannten Worte durchsucht und Überschneidungen aus dem Prozess gelöscht werden. Das heißt, sollte eine mögliche Überschneidung auftauchen, so wird nur das längste Wort behalten bzw. das Wort das beide Teilwörter enthält. Werden also wie im obigen Beispiel »laurel«, »branch« und »laurel branch« als Möglichkeit bereitgestellt, so wird zunächst verglichen ob die Indizes im gleichen Bereich liegen, heißt Anfang bzw. Endbuchstabe verglichen und anschließend wird das längste der drei Wörter behalten. Dieser Prozess wurde in der alten *spaCy* Version einfach ignoriert, sprich der zuletzt zugewiesene Entitätstyp wurde behalten.

3.5 Grundsätzlicher Unterschied der deutschen zur englischen Sprache

Bevor das NLP auf einen deutschen Datensatz angewendet wird, sollte man sich die Unterschiede der deutschen zur englischen Sprache vor Augen führen. Dies gilt jedoch genauso, für die Gemeinsamkeiten beider Sprachen. So gehören beide Sprachen derselben Sprachfamilie an. Die englische als auch deutsche Sprache zählen zu den germanischen Sprachen (Schmidt 2013, Kap. 1). Eine weitere nötige Grundgemeinsamkeit ist, offensichtlich, die Nutzung derselben 26 Buchstaben des lateinischen Alphabets. Wobei hier schon auf die ersten Unterschiede gestoßen wird, denn die deutsche Sprache besitzt darüber hinaus noch die Umlaute ä, ö, ü und das sogenannte scharfe S (ß)[36]. Auch grammatikalisch sind noch einige Gemeinsamkeiten zu erkennen – so sind die Regeln der temporalen Konjunktionen der Verben ähnlich. Das deutsche Pendant zum englischen »*drink, drank, drunk*« wäre beispielsweise »trinkt, trank, getrunken«. Wenn man nun jedoch nach dem Pendant des englischen Artikels »*the*« suche, hätte man im Deutschen hierfür ganze 6 bis 12 Möglichkeiten[37].

[36] https://grammis.ids-mannheim.de/rechtschreibung/6144 (10.09.2020)
[37] https://front-runner.de/gemeinsamkeiten-unterschiede-englisch-deutsch/ (13.09.2020)

»*My philological studies have satisfied me that a gifted person ought to learn English (barring spelling and pronouncing) in thirty hours, French in thirty days, and German in thirty years. It seems manifest, then, that the latter tongue ought to be trimmed down and repaired. If it is to remain as it is, it ought to be gently and reverently set aside among the dead languages, for only the dead have time to learn it.* «

- Mark Twain

Der amerikanische Schriftsteller Mark Twain lernte die deutsche Sprache und setzte sich ausführlich mit dieser auseinander. Seine zum Teil frustrierenden Erfahrungen beim Erlernen der deutschen Sprache als englischsprachiger Mann brachte er in seinem humoristisch-satirischem Werk „*The Awful German Language*“ auf Papier (Twain 1880). Twain behauptet unter anderem, dass es keine andere Sprache gebe, die so ungeordnet und unsystematisch sei wie die deutsche Sprache. Betrachtet man die Wortstellung, so ist die deutsche nicht wie die englische Sprache mit Subjekt - Prädikat - Objekt (S-P-O) zu beschreiben (Imo 2016, Kap. 3). Zu erwähnen ist hierbei, dass dies für englische Haupt- als auch Nebensätze gilt. Zwar lassen sich Sätze mit dieser Stellung auch im Deutschen einfach finden, doch es wird schnell klar, dass sich beide Sprachen in dem Punkt Wortstellung stark unterscheiden. Man betrachte folgendes Beispiel:

(1)

»Der Lehrer (Subjekt) gibt (Prädikat) dem Schüler (Dativobjekt) die Noten (Akkusativobjekt).«

»The teacher (Subjekt) give (Prädikat) the student (Dativobjekt) the grades (Akkusativobjekt).«

Es scheinen die Stellungen identisch zu sein. Der Unterschied wird erst dann deutlich, wenn man nun das Objekt an erster Stelle erwähnt:

(2)

»Die Noten (Akkusativobjekt) gibt (Prädikat) der Lehrer (Subjekt) dem Schüler (Dativobjekt).«

»The grades (Akkusativobjekt) give (Prädikat) the teacher (Subjekt) the student (Dativobjekt).«

oder

(3)

»Dem Studenten (Dativobjekt) gibt (Prädikat) der Lehrer (Subjekt) die Noten (Akkusativobjekt).«

»The student (Dativobjekt) give (Prädikat) the teacher (Subjekt) the grades (Akkusativobjekt).«

So sind in diesen Beispielen die jeweils englischen Sätze von (2) und (3) so in der Form nicht möglich. Man sieht, im Deutschen existiert keine feste Abfolge von Satzgliedern. Der deutsche Hauptsatz lässt sich viel eher durch folgende, aus drei Satzgrundmustern bestehende, Struktur beschreiben (Imo 2016, Kap. 3.3.1):

Vorfeld – finites Verb – Mittelfeld – infinites Verb (wenn vorhanden) – Nachfeld (wv.)

Diese Struktur, eigentlich genannt »Feldermodell«, sagt, anders als bei der S-P-O Satzstruktur, wenig über die Position des Subjektes aus (Imo 2016, Kap. 10). So kann sich das Subjekt legitimerweise am Ende, in der Mitte oder am Anfang des Satzes befinden:

»Auf dem Dach (Vorfeld) steht (finites Verb) ein Rabe (Mittelfeld).«
»Über dieses Problem (Vorfeld) werden (finites Verb) wir uns noch (Mittelfeld) unterhalten (infinites Verb), wenn das Meeting anfängt (Nachfeld).«
»Die Klausur (Vorfeld) hat (finites Verb) heute mal wieder unglaublich viel (Mittelfeld) abgefordert (infinites Verb).«

Besonders diese Eigenschaft ist im Hinblick dieser Ausarbeitung von Bedeutung, aber dazu später mehr (vgl. **Kapitel 5.3.1, Die Wortstellung**).

Mark Twains wohl größter Kritikpunkt an der deutschen Sprache betrifft die Stellung des Verbes. Verben befinden sich im Deutschen häufig am Ende des Satzes. Zum einen betrifft dies jeden Nebensatz. Zum anderen ist dies auch der Fall im Hauptsatz, immer dann, wenn ein Prädikat aus mehreren Teilen besteht (Imo 2016, Kap. 10). Kurz zur Wiederholung angemerkt; das Prädikat kann aus mehreren Verbteilen bestehen. Dem Vollverb, dem Hilfsverb (Auxiliarverb) und dem Modalverb (Imo 2016, Kap. 9.1.1). Ein Beispiel mit einem dreiteiligen Prädikat wäre der Satz »In zwei Wochen soll (Modalverb) ich die Ausarbeitung abgegeben (Vollverb) haben (Hilfsverb).« Diese Besonderheit des »auseinandergerissenes Verbes«, wird als Verbalklammer bezeichnet.

Beim Betrachten der Verbstellung, ist nämlich zu erkennen, dass die deutsche Sprache eine »Klammersprachen« ähnliche Syntax aufweist. Dies ist eben am deutlichsten bei den Verben zu beobachten. Steht das Verb nicht im Präsens (Gegenwartsform) oder Präteritum (Vergangenheitsform), so bildet es eine Satzklammer. »Tom hat ihn nicht erwischt« oder »Tom wurde von seinem Vater getadelt.«. Partikelverben wie »aufmachen, zumachen, anreisen, abladen, austrinken« sind hier die Ausnahme und bilden sogar im Präsens und Präteritum eine Klammer (Imo 2016, Kap. 5.2). »Tom schaltet den Fernseher an.« bzw. »Tom schaltete den Fernseher an.«. Dies wird durch das »auseinanderreißen« des Verbes ermöglicht. Die Klammersyntax wird selbst dann eingehalten, wenn Gebrauch von Modalverben gemacht wird. »Tom kann Hunde nicht leiden.«. Dabei bilden das Modal- und Vollverb die Klammern. Um eben diese, fast einzigartige (ähnlich wäre niederländisch), sprachlich-syntaktische Besonderheit erfassen zu können, wurde genau dieses Grammatikmodell in Form des »Feldermodells« eingeführt.

Ein weiterer, für diese Arbeit relevanter Punkt der genauer beleuchtet werden sollte, ist das schon erwähnte »auseinandergerissene Verb«. Laut Mark Twain, würde man durch das lange Warten auf das Verb, als nicht deutscher Muttersprachler, schnell den Überblick über den Satz verlieren. Außerdem würden deutsche Sätze durch die Klammerstruktur der Verben absurd wirken; dies zeigt er mit einem Beispielsatz aus einem deutschen Roman, welchen er anschließend ins Englische übersetzt (Imo 2016, Kap. 10 & Twain 1880, S.13).

»Da die Koffer nun bereit waren, REISTE er, nachdem er seine Mutter und Schwestern geküsst und noch einmal sein angebetetes Gretchen an den Busen gedrückt hatte, die, in schlichten weißen Musselin gekleidet, mit einer einzigen Teerose in den weiten Wellen ihres üppigen braunen Haares, kraftlos die Stufen herabgewankt war, noch bleich von der Angst und Aufregung des vergangenen Abends, aber voller Sehnsucht, ihren armen, schmerzenden Kopf noch einmal an die Brust dessen zu legen, den sie inniger liebte als ihr Leben, AB.«

»The trunks being now ready, he DE- after kissing his mother and sisters, and once more pressing to his bosom his adored Gretchen, who, dressed in simple white muslin, with a single tuberose in the ample folds of her rich brown hair, had tottered feebly down the stairs, still pale from the terror and excitement of the past evening, but longing to lay her poor aching headyet once again upon the breast of him whom she loved more dearly than life itself, PARTED.«

Das Konjugieren des Substantives, ist der letzte Unterschied auf den eingegangen werden sollte. Im Deutschen sind Substantive Worte, die aus distributioneller Sicht, als artikelfähig gelten. Das bedeutet, dass sich vor diese Worte Artikel platzieren lassen. Zusätzlich kann man zwischen Artikel und Substantiv ein Adjektiv setzen (Imo 2016, Kap. 6.1). Ein Beispiel für ein Substantiv wäre das Wort »Auto«, denn es gilt: »Das Auto« bzw. »Das grüne Auto«. Aus morphosyntaktischer Sicht müssen Substantive deklinierbar sein. Dabei beinhalten die Deklinationen das Genus (Geschlecht), den Numerus (Singular und Plural) und den Kasus (Nominativ, Genitiv, Akkusativ und Dativ). Im Gegensatz zum Deutschen, werden im Englischen alle Substantive, ausgenommen von Eigennamen, klein geschrieben.
In der deutschen Sprache verfügen Substantive außerdem immer über einen festen Genus, ein sogenanntes grammatisches Geschlecht und lassen sich in drei Gruppen unterteilen. Maskulina, also männliche Substantive; Feminina, weibliche Substantive und Neutra, sächliche Substantive. Das Genus existiert in der englischen Sprache nicht. Aus »der Hund«, »die Katze« und »das Auto« wird einfach nur »*the dog*«, »*the cat*« und »*the ship*«.[38]

[38] https://www.studienkreis.de/englisch/substantiv-nomen/ (15.09.2020)

Schaut man sich den Numerus an, sieht man im Englischen, dass die meisten Plurale mit der Endung -s an den Singular gebildet werden. Die Ausnahme hier sind Substantive die auf die Buchstaben -s, -x, -ch, -sh, -z enden. An diese wird zum bilden des Plurals ein -es angehängt. Natürlich gibt es auch unregelmäßige Pluralbildungen, die es auswendig zu lernen gilt. Dazu gehören Worte die auf Konsonanten + -y enden, wie *party - parties;* Endung auf Konsonant oder einzelnem Vokal + -f(e), wie *leaf - leaves*; Endungen auf -o, wie *potato - potatoes*. Hinzukommen kommen weitere Sonderformen wie *man - men, mouse - mice, child - children* und Worte, die im Singular als auch Plural gleichbleiben, so beispielsweise *fish* oder *sheep*. Im Deutschen hingegen gibt es insgesamt fünf Möglichkeiten den Plural eines Wortes zu bilden und hängen mit dem Kasus zusammen. Ein Element, das beide Sprachen bezüglich des Bilden des Plurals gemeinsam haben, sind zählbare und nicht zählbare Wörter (im Englischen *countable-* und *uncountable nouns*). Wenn man im Englischen den Plural eines nicht zählbaren Wortes bilden möchte, muss man Gebrauch von sogenannten *partititve nouns* machen. Den Plural von *water* (bzw. *bottle of water)* als Beispiel könnte man so mit *bottles of water* ausdrücken. Im Deutschen ist dies ähnlich. Nicht zählbare Worte können üblicherweise nur im Singular stehen. Wobei es auch hier Ausnahmen gibt, im Form von Redewendungen oder Ähnliches. Im Fall Wasser existiert bspw. die Redewendung »mit allen Wassern gewaschen sein« (Imo 2016, Kap. 6.1).

Bei der Deklination des Substantives unter einem der vier Fälle des Kasus, unterscheiden sich die beiden Sprachen am stärksten. Im generellen werden durch den Kasus, im Sinne der Verständlichkeit des Satzes, semantische Relationen zwischen Wörtern, kodiert. So kann beispielsweise der Nominativ, also der »Wer-Fall«, dafür verwendet werden, den »Agens« (lat. »*agere*« für handeln) – also die handelnde Person zu ermitteln. Während der Akkusativ, das »Wen«, dann für den »Patiens« (lat. »*patiens*« für erleiden) verwendet wird. Man kann der deutschen Sprache die Eigenschaft zu sprechen, einen gut erkennbaren Kasus zu haben. Dies ist in der englischen Sprache nicht der Fall. Bevor dies genauer beleuchtet wird, wird noch auf eine Gemeinsamkeit eingegangen. Wenn es ums reine Anzeigen des Besitzverhältnisses geht, wird der Genitiv, erfragt mit »Wessen«, im Englischen als auch im Deutschen ähnlich simpel gebildet. Im Englischen wird dies immer mit Apostroph + s angezeigt, im Plural hingegen entfällt das + s.[38] »This is Tom's restaurant.«. Im Deutschen ist es nur die Endung + s, auch wenn das Setzen des Apostrophs

ein gängiger Fehler ist. »Das ist Toms Restaurant.«. Darüber hinaus verfügt das Englische jedoch über so gut wie keine weitere Kasusmarkierung. Dies wird in folgenden Beispielsätzen deutlich:

»Der Polizist (Agens im Nominativ) schlägt den Gauner (Patiens im Akkusativ).«

»The policeman (Agens im Nominativ) beats the rogue (Patiens im Akkusativ).«

Soweit so gut, im Deutschen als auch Englischen sind beide Sätze grammatikalisch möglich und stellen den Polizisten als Handelnden dar. In einer umgekehrten Relation lassen sich die Sätze auch wie folgt darstellen:

»Der Gauner (Agens im Nominativ) schlägt den Polizisten (Patiens im Akkusativ).«

»The rogue (Agens im Nominativ) beats the policeman (Patiens im Akkusativ).«

Durch den nicht erkennbaren Kasus, sind diese beiden Sätze im Englischen die einzigen beiden möglichen Wortstellungen. Da der Nominativ und Akkusativ (gilt auch für den Dativ, »Wem«) nicht zu erkennen ist, ist das erste Substantiv immer im Nominativ. Denn, wie schon erwähnt, gilt immer S-P-O. Deutsch ist von solch einer Regelung nicht betroffen, eben dadurch, dass sich Nominativ und Akkusativ unabhängig der Wortstellung erkennen lassen. Daher sind zusätzlich diese beiden Sätze möglich:

»Den Polizisten (Patiens im Akkusativ) schlägt der Gauner (Agens im Nominativ).«
»Den Gauner (Patiens im Akkusativ) schlägt der Polizist (Agens im Nominativ).«

Die deutsche Sprache verfügt dem entsprechend über eine freiere Wortstellung und lässt so unterschiedlichste Varianten zu. Im Englischen können und werden die Beziehungen rein nach der Reihenfolge der Wörter im Satz bestimmt. Relationen in deutschen Sätzen, trotz solcher Wortstellungsfreiheit zu erkennen, setzt voraus, alle Kasusendungen der Sprache, ob die der Artikel oder die der Substantive, zu kennen – für Substantive allein bedeutet dies

schon drei grundlegende Deklinationsklassen (Feminina Deklination, starke Deklination und schwache Deklination) beherrschen zu müssen.

4. Das englische Modell

Der erste Teil dieser Arbeit beinhaltet das englischsprachige Modell, aus der hervorgegangenen Bachelorarbeit, zu erweitern. Dabei wird zuerst die Grundlage betrachtet und im Anschluss die neuen Erweiterungen erläutert. Das neue erweiterte Modell wird vorgestellt und im Folgenden werden die neu gewonnenen Erkenntnisse untersucht. Anschließend wird die Evaluation des neuen Modells analysiert. Abschließend wird die Umsetzung des alternativen Modells vorgestellt und dieses evaluiert.

4.1 Grundlage

Das grundlegende Modell von P. Klinger beschränkt darauf nur Personen und Objekte sowie die Relationen zwischen diesen, zu erkennen. Dabei wurde ausschließlich die Relation ausgehend von Person zu Objekt betrachtet (Person, Relation, Objekt). Das heißt als Subjekt wurden Elemente aus der Klasse PERSON und als Objekt Elemente der Klasse OBJECT betrachtet. Um Die Relation zwischen diesen beiden Entitätstypen zu erkennen wurde ein Ansatz gewählt, der nicht etwa überprüft welche Relation zwischen den betrachteten Entitäten besteht, sondern ob eine Relation zwischen diesen besteht (Klinger 2016, Kapitel 4.2). Durch diesen gewählten Ansatz kann die Relationsextraktion als Multiklassenproblem umgesetzt werden. Es sind 11 verschiedene Klassen zustande gekommen, wobei letztere Kombinationen sind, die keine Relation zueinander haben (siehe Tabelle 5).

relation	semantic cluster
holding	"holding", "carrying" (garment), "brandishing" (spear), "shouldering" (rudder), "playing" (lyre, aulos), "raising" (shield), "cradling" (torch)
wearing	"wearing", "covered with" (lion-skin)
resting_on	"resting on", "leaning on"
seated_on	"seated on", "seated in"
standing	"standing in" (biga), "driving" (biga), "standing on" (galley)
drawing	"drawing" (arrow)
stepping_on	"stepping on" (helmet)
grasping	"scooping" (gold), "reach out for" (person), "plucking" (chiton)
lying	"lying on"
hurling	"hurling" (thunderbolt)
no_existing_relation	

Tabelle 5: Klassifikation (Klinger 2018, Kap. 4.2)

4.2 Erweiterungen

Die Erweiterung der Grundlage beschäftigt sich mit folgenden fünf Schritten:

1. Tiere und Pflanzen als neue Entitätstypen einführen
2. Alle neu entstehenden Relationskombinationen analysieren
3. Neue Klassifikation erstellen
4. Analyse der Auswirkung auf das Modell
5. Datenqualität untersuchen

Das erste Ziel bzw. die erste Erweiterung ist es die Menge der zu erkennenden Entitätstypen von Personen und Objekten auf Personen, Objekte, Tiere und Pflanzen zu erweitern (**1.**). Darüber hinaus soll die Relationsrichtung, welche sich aktuell von Person zu Objekt beschränkt, erweitert werden, sodass, ausgehend von den Entitätsklassen PERSON, OBJECT und ANIMAL als Subjekt, Relationen ausgehen können. PLANT wird hierbei nicht betrachtet, da Relationen, die aus dieser Klasse ausgehen, im Datensatz zu gering auftauchen. Das heißt es werden zwei weitere Entitätstypen hinzugefügt und alle dabei

entstehenden neuen Relationen betrachtet (**2.**). Es soll beobachtet werden, welche neuen Relationen dabei entstehen bzw. vorkommen und wie sich diese auf die alte Klassifikation auswirken bzw. ob eine neue Klassifikation erstellt werden muss (**3.**). Weiterhin soll beobachtet werden, wie sich die Erweiterungen auf die bereits vorhandene Klassifikation und auf das Modell auswirkt (**4.**). Als Grundlage für das Modell wird ebenfalls die Datenbank CNO verwendet. Diese beinhaltet aktuell circa 2000 Ikonographen mehr als zum Zeitpunkt der Implementierung des ersten Modells. Da nun zwei weitere Entitätstypen betrachtet werden, müssen die Ikonographen erneut untersucht werden.

Die CNO- Datenbank hat Richtlinien wie eine ikonographische Beschreibung formatiert werden soll, das heißt welche Wörter bei gewissen Darstellungen zu wählen sind und welche Wörter grundsätzlich bei gewissen Entitäten gewählt werden sollen. Zum Beispiel soll das Wort »dog« durch »hound« ersetzt werden und auch die Nutzung eines Bindestriches wird untersagt. So soll zum Beispiel statt »ears-of-corn« »ears of corn« verwendet werden[39]. Die Vorgabe dieser Richtlinien unterstützt die Vorbereitung des Trainingsdatensatzes, da ein Wort nur in einer Form vorkommen sollte. Vor der Nutzung der vorhandenen Ikonographen werden diese analysiert und den Richtlinien angepasst. Aufgetretene Unterschiede wurden bei der Vorbereitung angepasst. Beispielsweise taucht das Wort »ivy wreath« sowohl als »ivy wreath« als auch in der Form »ivy-wreath« auf. Dies wird einheitlich nach den Vorgaben angepasst (**5.**). Alle Unterschiede, die auftauchen und wie diese angepasst werden, werden in **Kapitel 4.4** thematisiert.

Durch das Untersuchen der Designs wurde auch festgestellt, dass viele Elemente der zwei neuen Entitätstypen keine Relation zu einem zweiten Element besitzen. Das heißt Tiere oder Pflanzen, die in einem Satz das Subjekt bilden, besitzen kein Objekt, auf das sich ihre Relation bezieht. Ein Beispiel wäre folgende Ikonographie:

> »Eagle flying left, within linear square; all within incuse square.«[40]

Damit diese Information des »fliegenden Adlers« nicht verloren geht, wird zum bestehenden Modell ein weiteres Modell entworfen, das sich mit der Erkennung und Extrahierung der Relationen zwischen Subjekt und Verb beschäftigt (siehe **Kapitel 4.6**).

[39] https://www.corpus-nummorum.eu/pdf/ExternalCoinEntry.pdf (02.11.2020)
[40] DesignID = 5943

4.3 Implementierung

Das NER wird mit der Python Bibliothek *spaCy* realisiert. Dabei wird *spaCys EntityRecognizer* (siehe **Kapitel 3.4**) verwendet. Durch die vorbereitete Grundlage, der bereits manuell annotierten Tabellen für die verschiedenen Entitätstypen und nach den Anpassungen der vorhandenen Designs an den Richtlinien, ist es möglich den NER Prozess an den Designs anzuwenden. Ziel ist es hierbei eine möglichst hohe Erkennungsrate zu erzielen, da die spätere Relationserkennung von den erkannten Entitäten abhängig ist. Das heißt das NER bildet die obere Schranke für die Performance des REs.

Bevor das Modell erstellt werden kann, werden die Designs in einem Trainings- und Testsatz getrennt, dabei wird die Methode *sklearn.model_selection.train_test_split* verwendet. Der Trainingssatz besteht aus 75% der vorhandenen Ikonographen und der Testsatz dementsprechend aus 25%. Damit zukünftige Performanceauswirkungen nach Modifikationen des Modells erkannt werden, wird eine feste Trennung gewählt. Dies wurde mit dem Parameter *random_state* erreicht. Durch das Wählen einer bestimmten Auftrennung des Datensatzes ist es möglich Verbesserung an der Performance zu erkennen, da, bei Veränderungen im Code, gewährleistet wird, dass auf demselben Datensatz trainiert bzw. vorhergesagt wird. Ohne eine bestimmte Auftrennung variiert der Trainings- bzw. Testsatz, was zur Folge hat, dass *Precision*, *Recall* und F-Maß variieren und Veränderungen nicht nachvollzogen werden können. Nachdem zufriedenstellende Werte erreicht werden, werden weitere Durchläufe ohne eine feste Auftrennung des Datensatzes durchgeführt, um die erzielten Werte, durch verschiedene Konstellationen, zu überprüfen. Im folgenden Abschnitt wird nun die Realisierung der erweiterten NER und RE Pipeline vorgestellt.

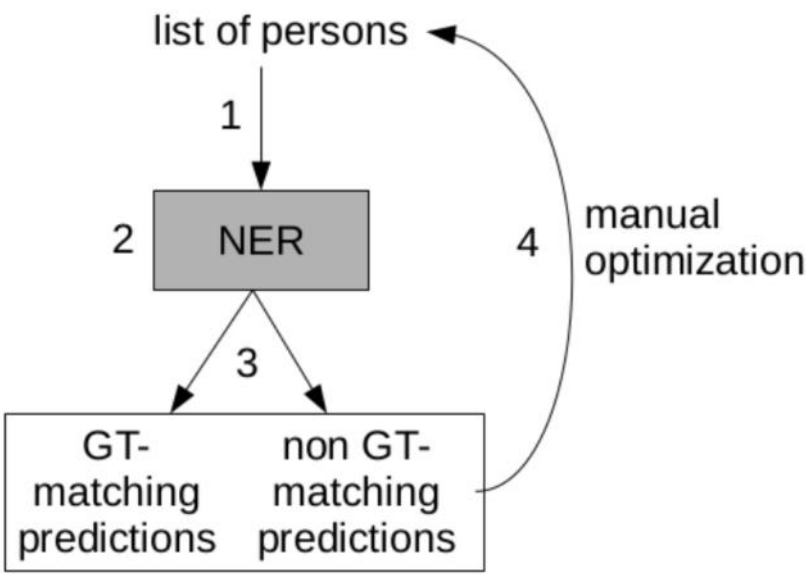

Abbildung 5 NER workflow (Klinger 2018, Kap. 4.1)

4.3.1 Named Entity Recognition

Die Eingabe für das NER besteht aus einer Ikonographie und dessen Annotation. Die Annotation hat folgende Form [(start, stop, label)]. Dabei steht »start« für die Position des ersten Zeichens des Elements, »stop« für die Position des letzten Zeichens und »label« für die Entität zu dem das Element gehört (Klinger 2018). Im Folgenden ist ein Beispiel, um die Struktur der Eingabe zu veranschaulichen:

> »Ares standing left, wearing helmet, holding patera in right hand and spear and shield in left hand.«[41]

Der Satz wird wie folgt annotiert:

> [(0, 4, PERSON), (29, 34, OBJECT), (44, 50, OBJECT), (69, 74, OBJECT), (79, 85, OBJECT)].

[41] DesignID = 6782

Nach dem Training wird das Modell auf den Testdatensatz angewendet. Es werden Vorhersagen getroffen, aus denen zwei Mengen entstehen (Abbildung 6). Die erste Menge beinhaltet die korrekten Vorhersagen, also die, die mit der Annotation übereinstimmen bzw. der *Ground Truth* (GT) entsprechen. Die zweite Menge beinhaltet die falschen Vorhersagen bzw. die, die nicht mit der Annotation übereinstimmen (non GT). In der zweiten Menge können aber Vorhersagen auftreten, die eigentlich korrekt sind. Das kommt dadurch zustande, dass die Annotation, also die *Ground Truth,* durch einen manuellen Prozess erstellt wird, der nur den aktuellen Wissenstand über akzeptierte Entitäten widerspiegelt. Diesen Wissensstand gilt es zu erweitern. Dies kann erreicht werden durch manuelles Betrachten der zweiten Menge, da *False Positive* Elemente enthalten sein können, die eigentlich tatsächlich ein neu entdecktes Element einer Entität beinhalten. Um diesen manuellen Arbeitsschritt vorzuzeigen, werden nun die falschen Vorhergesagten Personen betrachtet, nachdem das Modell, bei einem Durchlauf, auf dem Testsatz angewendet wurde.

> [Lucius Aelius, Sadales II, Marc Aurel, Maron, Altar, Pegasus, Smintheus, Acteon, Drusus Caesar, Sella curulis, Acteon, Maron]

Jedes dieser Worte liegt mit der DesignID und der Satzposition vor und kann nun in der Datenbank überprüft werden. Beginnt man nun den ersten Eintrag »Lucius Aelius« zu betrachten, kommt dieser zweimal als *False Positive* vor.

> »Bare-headed bust **OBJECT** of Lucius Aelius **PERSON**, right, wearing cuirass **OBJECT** and paludamentum **OBJECT**.«[42]

Nach Recherche in der OCRE[43] Münzsammlung und im Web[44] handelt es sich hierbei höchstwahrscheinlich um »Lucius Aelius Caesar«. Dieser kann nun als neue Entität aufgenommen werden oder als Alternativname hinzugefügt werden, falls dieser unter anderem Namen oder einer anderen Schreibweise bereits in der Datenbank vorhanden ist.

[42] DesignID = 6712
[43] http://numismatics.org/ocre/id/ric.2_3(2).hdn.2621 (02.12.2020)
[44] http://www.hellenicaworld.com/Italy/Person/de/LuciusAeliusCaesar.html (02.12.2020)

»Lucius Aelius« ist nicht in der Annotation vorhanden gewesen und wurde durch das Modell erkannt. Daher ist es ratsam die zweite Menge manuell zu überprüfen, mögliche richtige Vorhersagen in die Liste der Annotation aufzunehmen und ein erneutes Training durchzuführen. Dieser Prozess (Abbildung 5) wird solange wiederholt bis eine zufriedenstellende Erkennungsrate der vier Entitätstypen PERSON, OBJECT, ANIMAL und PLANT erreicht wurde. Eine Analyse der Erkennungsrate anhand der vorgestellten Metriken wird im **Kapitel 4.5** dargestellt.

4.3.2 Relation Extraction

Diese Arbeit behandelt ein Multiklassenproblem, das heißt, eine Eingabe kann mehreren Klassen zugeordnet werden. Der erste Schritt bei der Erstellung des Multiklassenproblems ist es, die Daten manuell zu untersuchen, die Relationen zwischen den verschiedenen Entitäten zu vermerken und anschließend Cluster zu bilden, die eine gemeinsame Klasse repräsentieren. In der folgenden Abbildung sind die verschiedenen Klassen zu sehen, die während des manuellen Annotierens des Trainingssatzes entstanden sind. Dabei wurden 1000 Sätze aus der CNO-Datenbank genommen, es wurde darauf geachtet, dass Entitätstypen wie ANIMAL und PLANT genügend oft vertreten sind. Es wurden zunächst die verschiedenen Relationen vermerkt und anschließend mit der Grundlage von P. Klinger verglichen und anhand der neu gefundenen Relationen erweitert bzw. angepasst. Ein Cluster repräsentiert dabei Verben mit einer ähnlichen Bedeutung, sprich: semantische Äquivalente. Cluster mit nur einem Element sind Verben, die spezifisch sind und dementsprechend nicht weiter »geclustert« werden können. Insgesamt sind so 18 Klassen erstellt worden (siehe Tabelle 6). Bereits vorhandene Klassen, wie »*holding*«, haben mehr Synonyme erhalten und sind darüber hinaus im Vergleich zur vorhandenen Klassifikation sieben neue Relationsklassen hinzugekommen. Die neu gefundenen Relationen sind »coiling«*, »feeding«, »breaking«, »receiving«, »pushing«, »flying_over«, »escorted_by«, »pointing_at«* und *»crowning«.* Auch wenn diese Relationen, im Vergleich zu »*holding«* und »*wearing«*, keine große Vertretung im Datensatz haben (Abbildung Verteilung), stellen diese einzigartige Relationen dar, die nur durch bestimmte Entitäten vorkommen und deswegen in den Trainingsprozess aufgenommen werden. So kommt »*flying_over«* nur bei fliegenden Tieren vor, *»coiling«* steht nur mit Schlangen und »*breaking«* steht größtenteils

nur mit einem Löwen in Verbindung. Die vorher vorhandene Klasse »*drawing*« wurde in die Klasse »*holding*« mit aufgenommen, dies geschieht durch den Aspekt, dass durch das Aufkommen vieler Relationen, die Klassenanzahl gering zu halten. Wenn ein Pfeil »gezogen« wird, so wird dieser in dem Moment »gehalten«, daher wurde hier diese Entscheidung getroffen. Generell wurde bei der Erstellung der Klasse die Wahl ob sich Elemente ähneln, aus dem oben erwähnten Aspekt, nicht so streng behandelt. In der folgenden Tabelle wird kenntlich gemacht, welche Klassen neu sind bzw. welche semantisch äquivalenten Worte in eine Klasse aufgenommen werden (rot) und welche (alte) Klassen jetzt Teil einer anderen Klasse sind (grün).

Klasse	**Semantisch Äquivalent**
holding	holding, brandishing, carrying, playing, cradling, supporting, pouring, drawing, picking, touching, containing, carying, drawing_out, raising, removing, collecting, hanging_on, shouldering
wearing	wearing, covered_with
resting_on	resting_on, reclining_on, leaning_on, setting_on, set_on
seated_on	sitting_on, sitting, seated_in, seated_on, riding, riding_on
grasping	grasping, scooping, reach_out, plucking, clasping, strangling, placing
standing	standing, standing_on, standing_in, driving, in (in biga)
crowning	crowning
coiling	(serpent) coiling, twining
feeding	(Athena, Hygieia) feeding (serpent)
breaking	(lion) breaking
pushing	pushing
receiving	receiving
flying_over	(eagle) flying_over
escorted_by	escorted_by
pointing_at	pointing_at, pointing
lying	lying_on
hurling	hurling
no_existing_relation	

Tabelle 6: erweiterte Klassifikation für das englische Modell

Nach der Erstellung der zu bestimmenden Klassen wurde der Trainingssatz anhand dieser manuell annotiert. Eine Annotation hat folgende Form (NE_1 ,α, NE_2), mit NE_1 ε aus [PERSON, OBJECT, ANIMAL], α ε aus den in Abbildung X.Y vorgestellten Klassen und NE_2 ε aus [PERSON, OBJECT, ANIMAL, PLANT]. Die Entität PLANT wird nicht als Subjekt betrachtet, da in dem vorhandenen Datensatz nicht genügend Relationen auftauchen, die von Elementen der Entität PLANT ausgehen.

Die Häufigkeit, des Auftretens der Relationen im vorbereiteten Datensatz, ist in der folgenden Abbildung zu sehen:

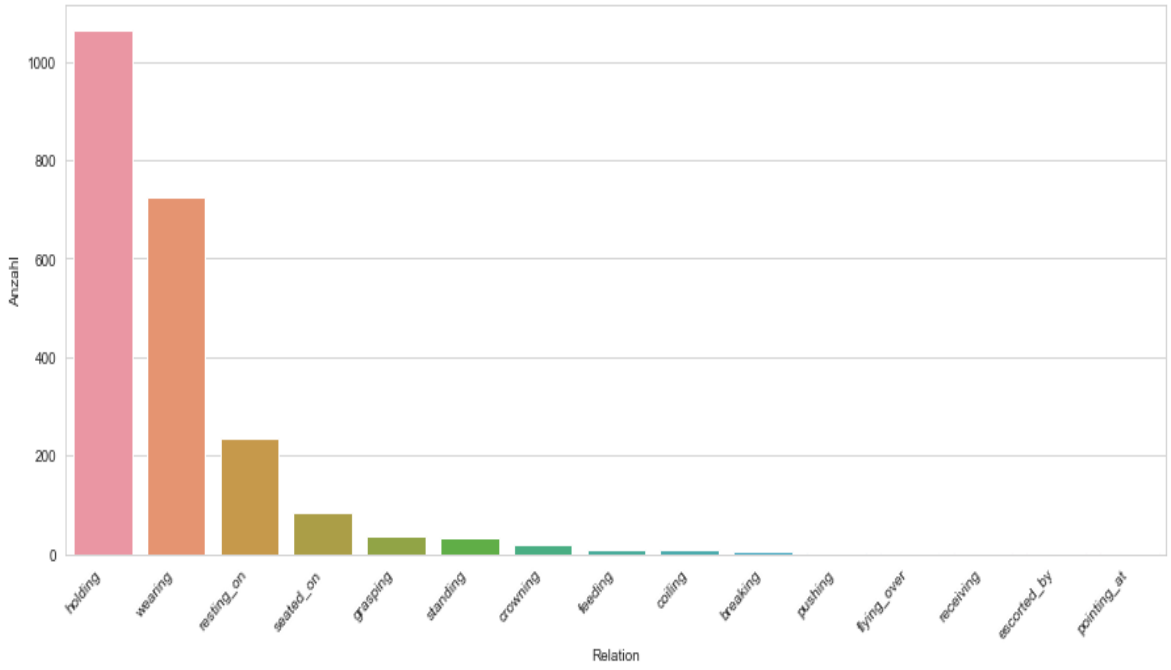

Abbildung 6: Verteilung der Klassen im genutzten Datensatz (englisches Modell)

Die Klasse »*holding*« kommt 1063 Mal im vorbereitetem Datensatz vor, »*wearing*« 723 Mal. Die beiden Klassen überwiegen im Vergleich zu den anderen Klassen. »*Resting_on*« kommt 234 Mal vor, unter 100 Mal kommen die Klassen »*seated_on*« (85), »*grasping*« (38), »*standing*« (32), »*crowning*« (18) vor. Die Restlichen tauchen weniger als 10 Mal auf. Dass die Klassen »*holding*« und »*wearing*« überwiegen, liegt zum einen daran, dass in Anbetracht der Grundlage, nämlich einer Münzdatenbank, es gängig ist, Personen abzubilden, die etwas halten oder tragen. Außerdem ist das Erkennen bzw. Abbilden einer etwas haltenden oder etwas tragenden Entität einfacher, als das Darstellen von spezifischeren bzw. detaillierteren Handlungen, wie z.B. Interaktionen zwischen Mensch

und Tier (wobei reiten/stehen die simpelste Interaktion ist). Darüber hinaus ist ein weiterer Grund für diese Verteilung, dass die Entitätstypen PERSON und OBJECT deutlich öfter vertreten sind als ANIMAL und PLANT und somit auch deren Relationen häufiger auftauchen. Für die Ikonographie

> »Athena seated left, wearing helmet, holding patera in outstretched right hand and spear in left hand; shield behind.«[45]

sieht die Annotation wie folgt aus:

> [[Athena, wearing, helmet], [Athena, holding, patera], [Athena, holding, spear]].

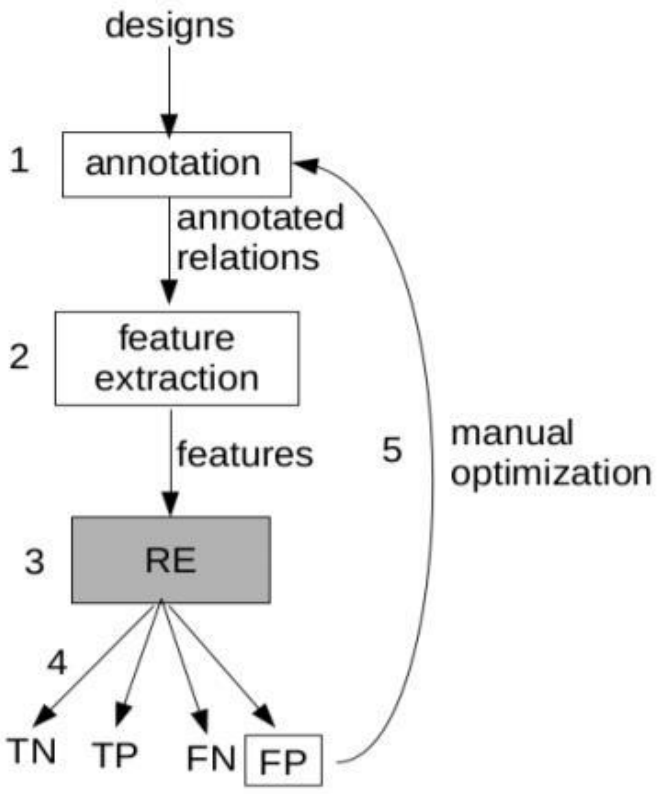

Abbildung 7: RE workflow (Klinger 2018, Kap. 4.2)

Nachdem der Datensatz annotiert worden ist, folgt das Training (Schritt 1, Abbildung 7). Bevor der Datensatz für das Training jedoch verwendet werden kann, werden Methoden

[45] DesignID = 174

gewählt, um die Features für den Trainingsprozess zu extrahieren (Schritt 2, Abbildung 7). Dieser Schritt ist notwendig, um die Eingabe für maschinelle Lernalgorithmen vorzubereiten. Nachdem die Eingabe transformiert worden ist, wird diese als Eingabe für den Klassifikator genutzt. Um die bestmögliche Kombination, der in **Kapitel 3** vorgestellten Methoden und Algorithmen, zu finden, wird ein *Gridsearch* ausgeführt. Beim *Gridsearch* werden alle möglichen Kombinationen getestet und anschließend ausgewertet, mit dem Ziel, die ideale Kombination für den vorhandenen Datensatz zu finden. Die ersten drei Schritte werden anhand einer Pipeline realisiert, bei dem die Eingabe der vorbereitete Datensatz ist. Die Pipeline besteht aus dem NER Prozess, der Feature Extrahierung, sprich unter anderem die Ermittlung des Pfades, und der abschließenden Relationsextrahierung, die sich wiederrum in weiteren Schritten aufteilt. Die Relationsextrahierung besteht aus der Eingabe der Schritte 1 und 2 sowie der Wahl der Methoden der Featureverarbeitung sowie des gewählten Algorithmus. Die Wahl wird durch ein *Gridsearch* ermittelt, die Auswertung des Gridsearchalgorithmus wird in **Abschnitt 4.5** gezeigt. Die besten Ergebnisse liefert folgende Kombination bzw. Pipeline und besteht aus den Schritten:

1. *Named Entity Recognition*
2. *Feature Extraction*
3. *Relationen Extraction*
    a. Path2Str
    b. CountVectorizer
    c. LogisticRegression

Die einzelnen Schritte werden im Folgenden erklärt, insbesondere welche Transformationen stattfinden bis zur letztendlichen Eingabe für den Trainingsprozess.

Der erste Schritt ist es, die Entitäten zu erkennen, dafür wird das vorbereitete Modell genutzt, das im vorherigen Abschnitt vorgestellt worden ist. Nach dem ersten Schritt ist die Ausgabe, für ein Element aus den Designs, eine Liste von Tripeln, bestehend aus Design, Subjekt und Objekt.

Folgende Ikonographie:

> »Ares standing left, wearing helmet, holding patera in right hand and spear and shield in left hand.«[46]

besitzt die Ausgabe:

> [(Ares standing left, wearing helmet, holding patera in right hand and spear and shield in left hand., Ares, helmet),
> (Ares standing left, wearing helmet, holding patera in right hand and spear and shield in left hand., Ares, patera),
> (Ares standing left, wearing helmet, holding patera in right hand and spear and shield in left hand., Ares, spear),
> (Ares standing left, wearing helmet, holding patera in right hand and spear and shield in left hand., Ares, shield),
> …
> (Ares standing left, wearing helmet, holding patera in right hand and spear and shield in left hand., spear, shield)].

Im zweiten Schritt wird nun die Relation zwischen einer gebildeten Kombination untersucht. Dies geschieht durch die Untersuchung des Pfades im Abhängigkeitsbaum. (Abbildung 8) [47].

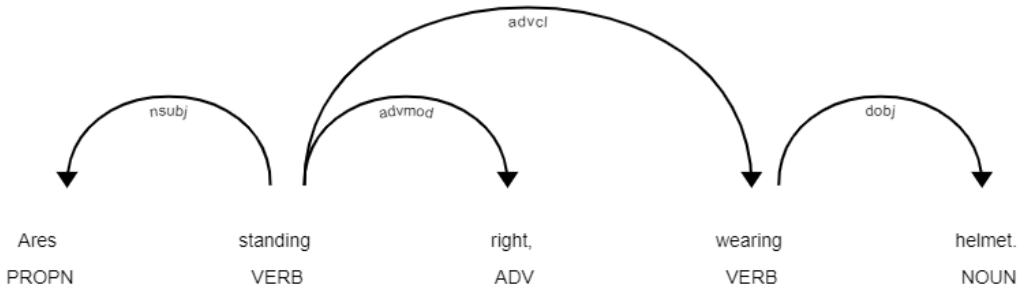

Abbildung 8: Abhängigkeitsbaum auf einem verkürzten Satz, zwecks Darstellung

[46] DesignID = 6782
[47] https://spacy.io/usage/linguistic-features *Legende zu den POS- und DEP- Tags

Der Pfad zwischen der betrachteten Kombination wird zurückverfolgt, Ziel ist es hierbei das Bindungsglied der beiden Entitäten zu finden. Das Bindungsglied ist hierbei meistens ein Verb. Es wird also vom Objekt betrachtet angefangen. Dabei wird der Pfad bis zum letztmöglichen Vorfahren zurückverfolgt und alle Satzbausteine gespeichert, die auf diesem Pfad liegen. Dasselbe Verfahren wird auch ausgehend vom Subjekt angewandt. Anschließend hat man den vollständigen Pfad zwischen Subjekt und Objekt. Dieser extrahierte Pfad enthält alle wichtigen Informationen, die für das maschinelle Lernen benötigt werden. Er enthält die Informationen wie diese Kombination auftritt und welches Wort sie zusammenknüpft. Dieser Pfad wird nun für den nächsten Schritt weitergegeben. Der nächste und letzte Schritt ist die Relationsextrahierung, dieser teilt sich in drei weitere Schritte auf. In Schritt 3.a wird *Path2Str* (siehe **Kapitel 3.2**) verwendet. Wird der Pfad aus der obigen Abbildung zwischen »Ares« und »helmet« betrachtet, entsteht folgender Pfad:

[Ares, standing, wearing, helmet]

Path2Str(ent) liefert dann folgende Transformation:

Ares\PERSON standing\VERB wearing\VERB helmet\OBJECT

Diese Transformation wird nun in dem nächsten Schritt 3.b weiterverarbeitet. Hier gilt es nun die Eingabe, die als String vorliegt, in einen Vektor umzuwandeln, um diese für das maschinelle Lernen zu verwenden. Für die ermittelte vielversprechendste Kombination wird hier die Funktion *sklearn.feature_extraction.text.CountVectorizer* genutzt (siehe **Kapitel 3.2**). Der letzte Schritt 3.c ist nun die Eingabe dem Klassifikator zu übergeben, in diesem Fall geschieht dies mithilfe der Logistischen Regression (siehe **Kapitel 3.1**).

4.4 Herausforderungen bzw. Anpassungen und Erweiterung der Daten

Bevor die Daten verwendet werden können, gilt es zunächst die Ikonographen zu analysieren. Dabei ist aufgefallen, dass Worte sowohl ohne Bindestrich vorkommen als auch mit, daher wurde das Bindestrich aus einem Großteil der Worte entfernt, dies sollte

auch laut Richtlinien des CNO vermieden werden »lion skin, not lion-skin«[48]. Worte, bei dem dies der Fall war, sind unteranderem Folgende:

> »Ivy-wreath, Laurel-branch, Lion-skin, Water-urn, Cubit-role, Serpent-staff, Wine-skin, Wheat-ears, Apis-bull, Palm-tree«

Alle Designs wurden anhand dieser Richtlinie angepasst.

Durch die neu hinzugefügten Entitätstypen ANIMAL und PLANT sowie die neu entstandenen Relationskombinationen musste eine neue *Ground Truth* Annotation erstellt werden. Diese wurde anhand von 1000 englischen Ikonographen angefertigt. Ein Arbeitsaufwand von circa 4 Tagen. Weiterhin ist beim Betrachten der Ikonographen aufgefallen, dass viele Elemente keine Relation zu anderen Elementen besitzen. Vor allem tritt dies bei Tieren auf. Häufig kommt ein Tier vor, das nur mit einem Verb in Verbindung steht.

»Horse prancing left; above, owl flying. Border of dots.«[49]

So kommt in der obigen Ikonographie das Pferd, das herumtanzt und die Eule, die fliegt, vor, dass eine Beziehung zu einem anderen Element fehlt. Damit diese Information nicht verloren geht, ist ein weiteres Modell geplant, das Subjekte und dessen Relationen zu Verben erkennt. Dieses Modell soll daher erkennen, mit welchen Verben die Entitäten in Verbindung stehen, Ziel ist daher, zusätzlich zur Erweiterung des erst vorgestellten Modells, ein weiteres Modell, das ermöglicht Entitäten zu erkennen und deren »Tun«, beispielsweise die Eule die fliegt, der Kaiser der etwas hält etc. Aus dem eben genannten Problem ist die Idee eines zweiten Modells entstanden, welches ebenfalls eine positive neue Erweiterung für das erste Modell mit sich bringt. Bei der Betrachtung der PoS-Tags und der Verben ist zu sehen, dass diese oft als Nomen erkannt werden. Das hat den Grund, dass die Struktur der Ikonographen bzw. der Aufbau des Satzes von einer normalen Struktur abweicht. Ein daraus entstehendes Problem ist, dass bei PoS-Tags Verben in den

[48] https://www.corpus-nummorum.eu/pdf/ExternalCoinEntry.pdf (19.10.20)
[49] DesignID = 1573

Ikonographen häufig als Nomen markiert werden. Beispielsweise wird in der Ikonographie »Dolphin swimming upwards.« »swimming« als Nomen markiert.

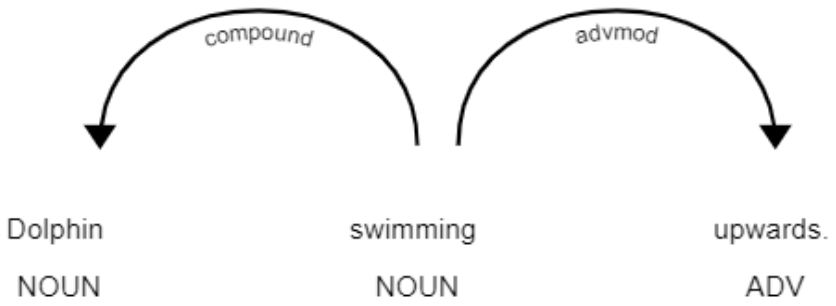

Abbildung 9: spaCys PoS-Tags

Um dieses Problem zu lösen wird das NER um eine Entität erweitert. Es wird die Entität VERB mit in das NER Modell aufgenommen. Diese Variante wird implementiert, da die POS-Tags aufgrund der Struktur der Ikonographen nicht korrekt bestimmt werden. Diese Entscheidung hat den Vorteil, dass Verben korrekt erkannt werden. Der Nachteil dieser Variante ist, dass wie bei den Entitätstypen PERSON, OBJECT, ANIMAL und PLANT eine manuell erstelle Tabelle mitgeführt werden muss. Zusätzlich wird die Methode Path2Str um einen Parameter erweitert und zwar um die Möglichkeit den Entitätstag als Information hinzuzufügen. So würde der Pfad [Ares, standing, wearing, helmet] wie folgt aussehen:

Ares\PERSON standing\VERB wearing\VERB helmet\OBJECT

Diese neue Erkenntnis und Umsetzung haben sich als bestes Feature erwiesen. Wie man im nächsten **Kapitel 4.5** Evaluation sehen wird.

4.5 Analyse und Evaluation

In diesen Abschnitt wird nun das NER analysiert und evaluiert, darauffolgend das RE. Es wird ein *Gridsearch* ausgeführt, um die beste Konstellation für das RE zu finden. Abschließend wird eine Stichprobe analysiert, um den Durchlauf sowohl auf menschliche

Fehler bei der Annotation zu untersuchen als auch Hindernisse für das Modell zu entdecken.

4.5.1 NER Auswertung

Zunächst werden die vorhandenen Ikonographen analysiert. Betrachtet werden zunächst die vier Entitätstypen PERSON, OBJECT, ANIMAL, PLANT. Es werden insgesamt 21027 Entitäten annotiert.

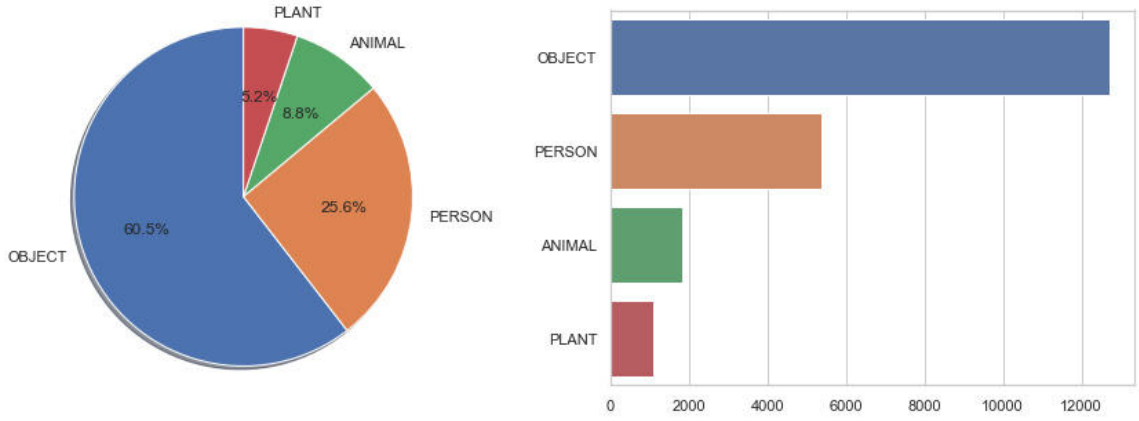

Abbildung 10: Die Verteilung der Entitäten auf den gesamten englischen Datensatz

Die überwiegende Klasse stellt dabei die Klasse OBJECT mit 12721 Elementen dar, dies entspricht 60.5% der gesamten annotierten Elemente. Nach OBJECT stellt PERSON, mit 5376 Elementen (25.6%), die zweitgrößte Klasse dar, gefolgt von ANIMAL (1844, 8.8%) und PLANT (1086, 5.2%). In der folgenden Abbildung sind die jeweils 15 häufigsten Elemente der jeweiligen Klasse zu sehen.

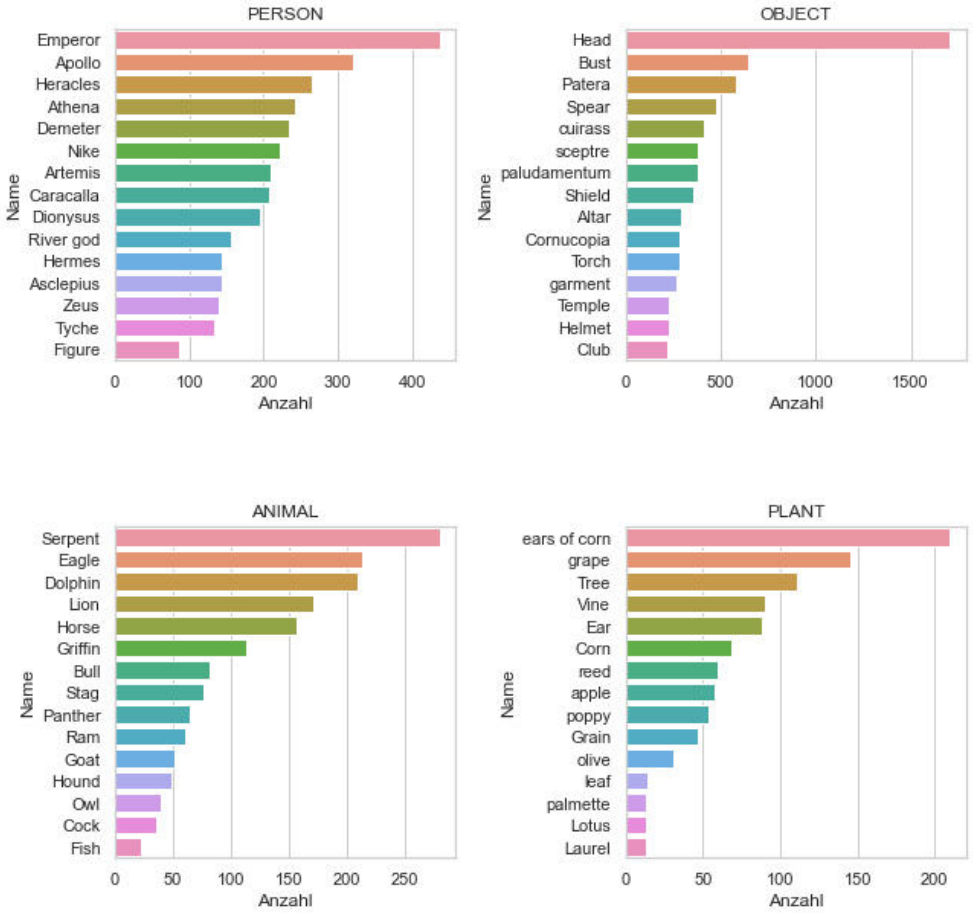

Abbildung 11: Top 15 der jeweiligen Entitätsklassen (englisches Modell)

Zusätzlich zu den vorgestellten vier Entitäten soll ebenfalls eine neue Entität VERB bestimmt werden. Diese dient dazu, das Feature *Path2Str*, zu optimieren als auch generell Verben besser bestimmen zu können bzw. für das RE zu nutzen (siehe **Kapitel 4.4**). Verben stellen dabei die zweitgrößte Menge dar. Es werden insgesamt 9092 Verben, durch das NER Modell, gefunden. Verben stellen nun 30.2% der annotierten Elemente dar.

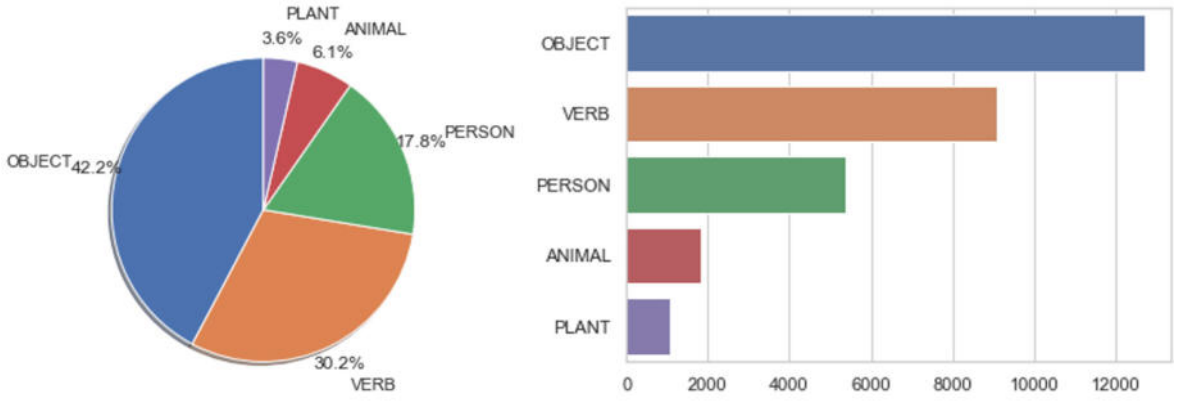

Abbildung 12: Die Verteilung der Entitäten inkl. Verben auf den gesamten englischen Datensatz

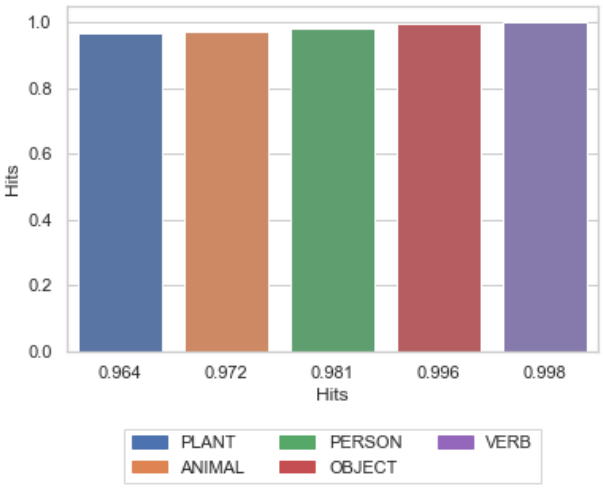

Abbildung 13: Genauigkeit des NER

Als Testsatz wurden 25% der Designs gewählt, nach der Vorhersage hat das Modell 98,1% aller Personen erkannt. Unter den restlichen 1,9% können sich noch unbekannte, aber korrekte, Entitäten befinden, diese müssen manuell überprüft werden, dadurch kann das Modell durch ein erneutes Training verbessert werden. Objekte wurden zu 99,6%, Tiere zu 97,2% und Pflanzen zu 96,4% erkannt. Dass Pflanzen und Tiere eine geringere Erkennungsrate haben als Personen und Objekte, liegt daran, dass diese auch deutlich weniger auftauchen, dennoch ist die Rate im Vergleich zur Anzahl sehr hoch. Beachtet man das F-Maß (**99.1%**, Abbildung 15), der als ein Maß für die Leistung des Modells genommen werden kann, so fällt dieser ebenfalls sehr hoch aus.

	Gesamt	Richtige Vorhersagen	(Falsche) Vorhersagen
Person	1365	1339	17
Object	3201	3187	38
Animal	432	420	6
Plant	280	270	5
Verbs	2327	2322	10

Abbildung 15: Auswertung Testsatz

Entitität	Precision	Recall	F1
Person	98.7	98.1	98.4
Object	98.8	99.6	99.2
Animal	98.6	97.2	97.9
Plant	98.2	96.4	97.3
Verbs	99.6	99.8	99.7
Total	99.0	99.1	99.1

Abbildung 14: Evaluation des Modells – Die Performance jeder Entitätsklasse für sich und Performance total (englisches Modell)

Das Modell erzielt ein F-Maß gemessen an allen Entitäten von 99,20%. Vergleicht werden diese nun mit dem vorherigen Modell. Das vorherige Modell erzielte eine *Precision* bzw. ein *Recall* von 98% bzw. 97% (Klinger 2018, Kap. 6), das neue Modell erzielt eine *Precision* bzw. ein *Recall* von **99%** bzw. **99,1%**. Daraus ist erkennbar, dass die neu hinzugefügten Entitätstypen, ebenfalls sehr gut erkannt werden und die Performance nicht darunter leidet, im Gegenteil sogar besser ist, wobei zu beachten ist, dass in diesem Datensatz 2000 Ikonographen mehr zur Verfügung stehen.

4.5.2 RE Auswertung

Wie bereits erwähnt, werden bei der Extrahierung der Relationen verschiedene Kombinationen aus Klassifikator und Feature getestet, mit dem Ziel die beste Kombination

zu ermitteln. Dafür werden alle verschiedenen Konstellationen mithilfe eines *Gridsearchs* getestet und die besten 20 sind in der Abbildung elf zu sehen.

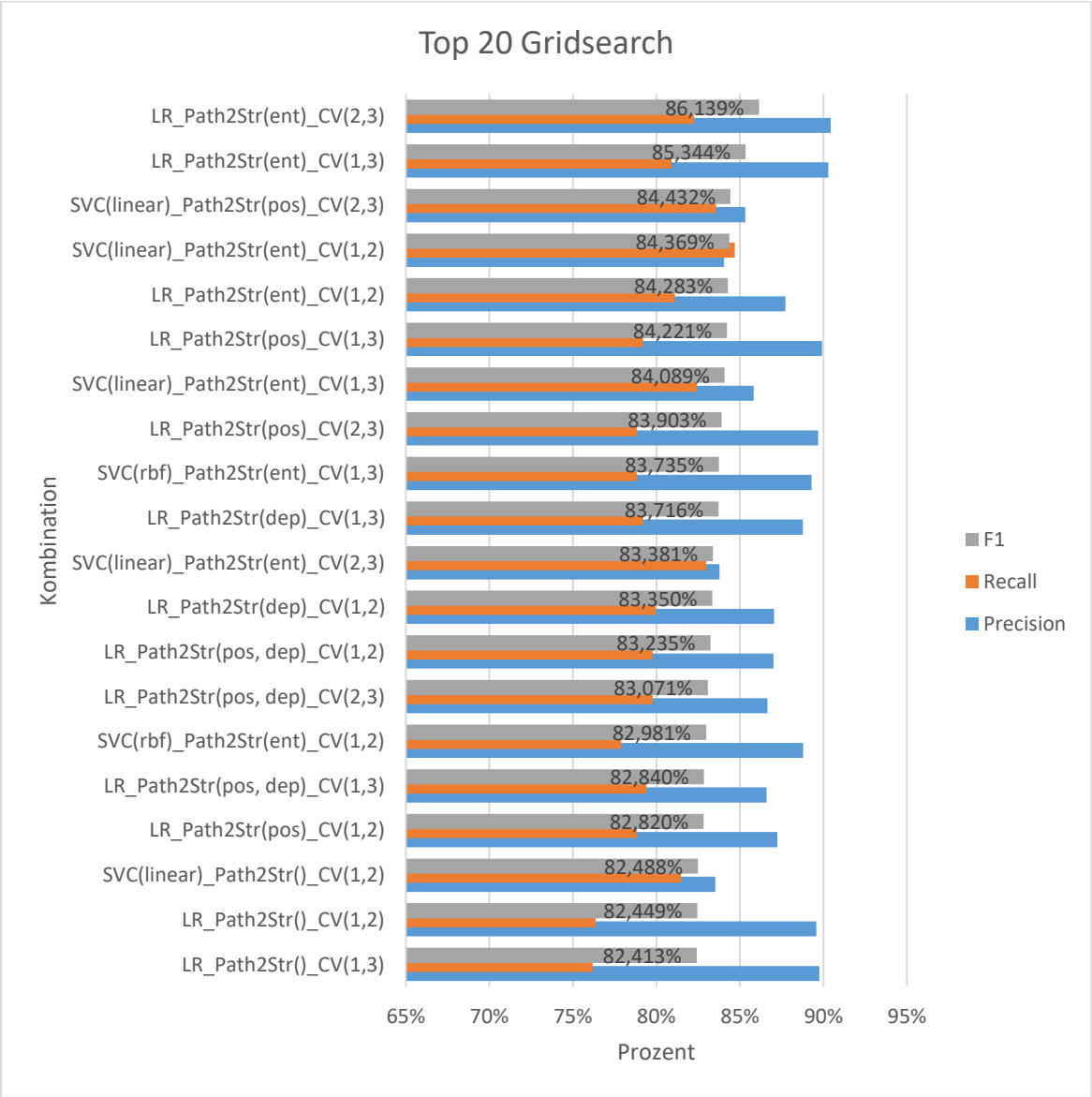

Abbildung 16: Top 20 Kombinationen (F-Maß absteigend, englisch)

Aus der obigen Abbildung ist zu erkennen, dass die besten Kombinationen (im Folgenden nur noch *leader*), bestehend aus Logistische Regression, *Path2Str(ent)* und *Countvectorizer*(*ngram*=2,3), das beste F-Maß erreicht mit einem Wert von **86,139%**. Die *Precision* bzw. *Recall* des *leader* liegt bei **90,04%** bzw. **82,22%**. Zu sehen ist, dass Konstellationen mit dem Klassifikator RF nicht unter den besten 20 auftauchen. Beachtlich ist aber, dass Kombinationen aus RF, *TfidfVectorizer(1,2)* und *Path2Str(dep)* oder *Path2Str(pos)* eine sehr hohe *Precision* erzielen, erstere erreicht 95,2% und letztere 94,9%. Der *Recall* liegt aber nur bei 61% bzw. 63,8% und daraus folgt ein F-Maß von nur 74,4% bzw. 76.3%. Die Konstellation aus SVC(rbf) und *AveragedPath2Vec* erzielt eine *Precision*

von 100%, aber ein *Recall* von nur 1%. Den besten *Recall* erzielt die Kombination aus SVC(linear), *Path2Str(ent)* und *CountVectorizer*(ngram=1,2) mit 84,6%, die *Precision* liegt bei 84% und das F-Maß bei 84,4%.

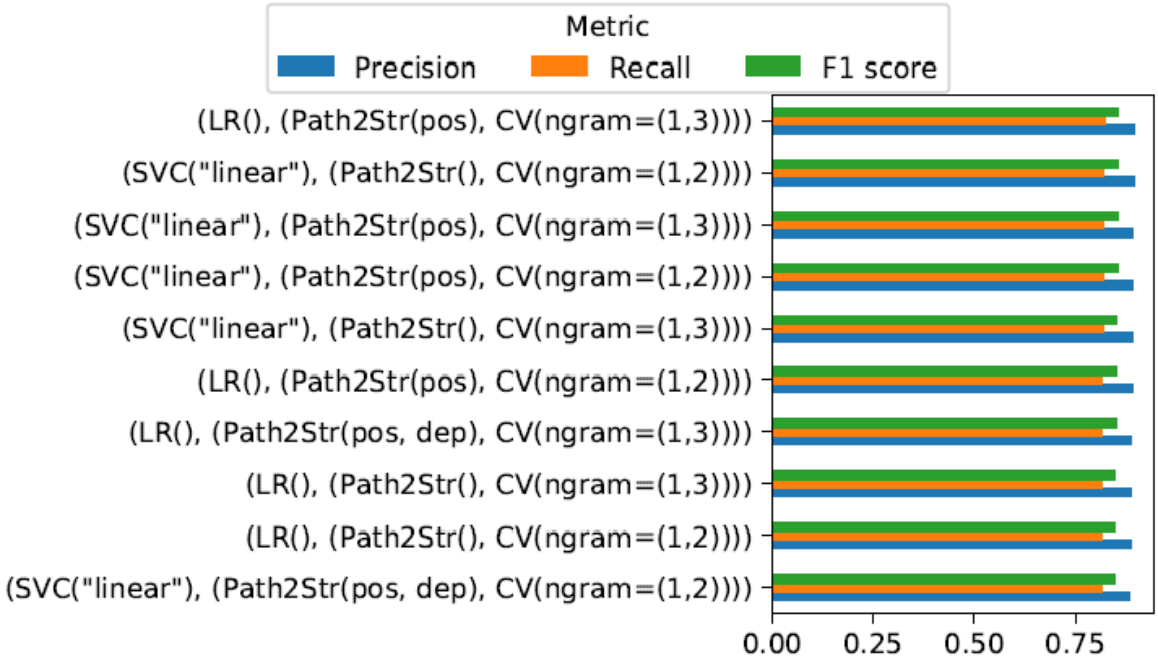

Abbildung 17: Gridseach, Grundlage (PERSON,Verb,OBJECT) – (Klinger 2018, Kap. 6.2)

Aus der Grundlage geht hervor, dass die Kombination mit dem besten Ergebnis aus LR, *Path2Str* und *CountVectorizer,* mit einem ngram von 1 bis 3, besteht. Diese erreicht eine *Precision* bzw. ein *Recall* von 93% bzw. 84% und daraus folgend ein F-Maß von 88% (Klinger 2018, Kap. 6.2). Von diesem F-Maß schwankt das neue Modell um 2%. Vergleicht man die Möglichkeiten der Modelle, so werden nun ebenfalls zu den Klassen PERSON und OBJECT, zusätzlich die Klassen ANIMAL und PLANT erkannt und sowie alle Relationen zwischen diesen Klassen und ausgehend aus PERSON, OBJECT und ANIMAL. In der folgenden Abbildung ist die Performance, aller hier getesteten Kombinationen, zu sehen. Zu erkennen ist, dass sich bei den vier getesteten Klassifikatoren zwei Cluster bilden sowie vereinzelte außerhalb der beiden Cluster. Die beste Performance erreicht die Kombination bestehend aus *Path2Str, CountVectorizer* und einem beliebigen Klassifikator, wobei LR und SVC(linear) die beste Performance im Schnitt erreichen.

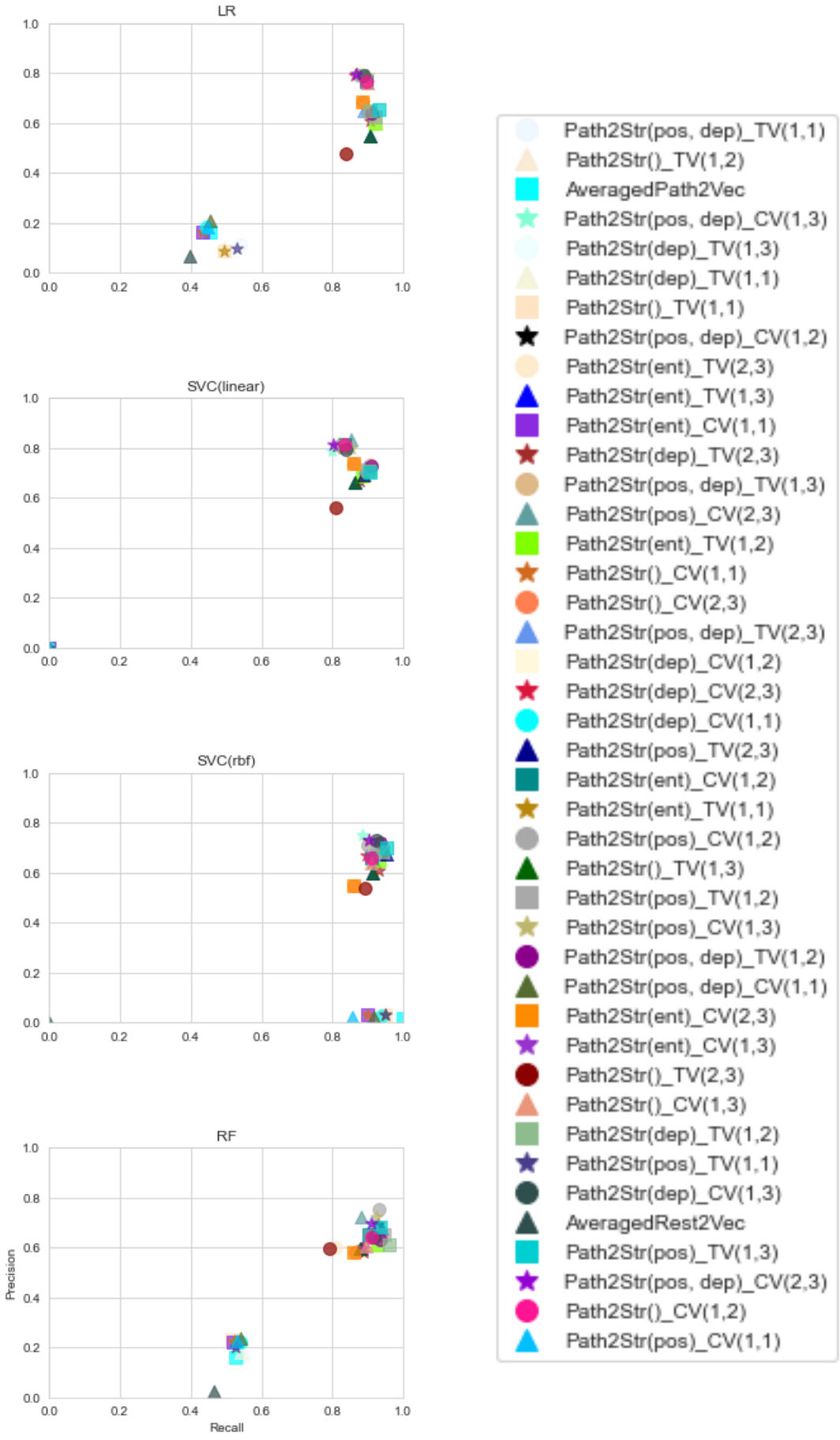

Abbildung 18: Performanceüberblick der verschiedenen Kombinationen (englisches Modell)

4.5.3 Stichprobe

Es wird eine Stichprobe aus dem Trainings- bzw. Testdatensatz entnommen und auf diese das Modell angewendet. Die Stichprobe besteht aus 50 Ikonographen, diese werden nun manuell untersucht. Das Ziel ist es hierbei menschliche Fehler zu finden, die beim manuellen annotieren passieren können, als auch Hindernisse zu entdecken, bei dem das Modell auf Grenzen stößt. Dabei lassen sich die Beobachtungen in vier Punkten aufteilen.

I. Menschliche Fehler beim Erstellen der Annotation
II. Falsche Vorhersagen des Modells
III. Fehlende bzw. Schwache Abhängigkeitserkennung
IV. Leistung des NERs bzw. manuelle Pflege der Datenbank

Beispiel zu I Das Annotieren der *Ground Truth* passiert manuell, heißt es ist nicht auszuschließen, dass menschliche Fehler passieren. Wie bei der folgenden Ikonographie zu sehen, findet das Modell die Relation »Artemis, resting_on, torch«.

> »Artemis standing facing, head left, holding patera in outstretched right hand, left resting on long torch, with quiver over shoulder; hound at side to left; club to right.«[50]

Dank des Modells kann die übersehene Relation zur *Ground Truth* aufgenommen werden.

Beispiel zu II

> »Athena enthroned left; throne decorated with Sphinx, left, front leg ends in lion's paw; holding in right hand patera from which she feeds a serpent entwined around tree in front of her, leaning left arm on throne back; behind her, owl sitting left on a frontal shield.«[51]

[50] DesignID = 112
[51] DesignID = 173

Der obige Ikonograph besitzt folgende Ground Truth:

> (Athena, PERSON, holding, patera, OBJECT), (Athena, PERSON, feeding, serpent, ANIMAL), (Athena, PERSON, resting_on, throne, OBJECT), (owl, ANIMAL, seated_on, shield, OBJECT)

Das Modell erkennt alle korrekt, schlägt jedoch ein weiteres (Athena, PERSON, resting_on, throne, OBJECT). Betrachtet wird der Pfad zwischen Athena und dem ersten Vorkommen von »throne«, der Pfad ist: »Athena, enthroned, decorated, throne«. Betrachtet wird ebenfalls der Pfad des zweiten »*throne*«, dieser ist »Athena, enthroned, decorated, ends, holding, in, patera, feeds, leaning, on, throne«. Zu erkennen ist, dass ersterer Pfad Teil des zweiten ist. Betrachtet man generell die Pfade zwischen »Athena« und »throne«, so ist die Änhlichkeit zum zweiteren, korrekten, »*throne*« gegeben und erklärt die Vermutung des Modells es diesem zuzuordnen.

Beispiel zu III Ein weiteres Problem ist bei der folgenden Ikonographie zu erkennen:

> »The three Charites (or Nymphs?) standing facing, left one head right, middle and right one, head left; wearing long garments, left, holding jar in right hand, right in left hand; the middle holding jar in right hand and ears of corn in left hand.«[52]

Das Modell findet keine Relation in dieser Ikonographie. Betrachtet wird hier ebenfalls der Pfad und es ist zu erkennen, dass der Satz in zwei Teile gespalten wird und somit »*Charites*« bzw. »*Nymphs*« keine Verbindung zu dem Rest des Satzes haben. Das heißt, dass *spaCys dependancy parser* keine Abhängigkeiten bilden konnte. Dies tritt auch in einer weiteren Ikonographie[53] auf, hier wird keine Abhängigkeit zwischen »Demeter« und »*torch*« erkannt.

[52] DesignID = 262
[53] DesignID = 354

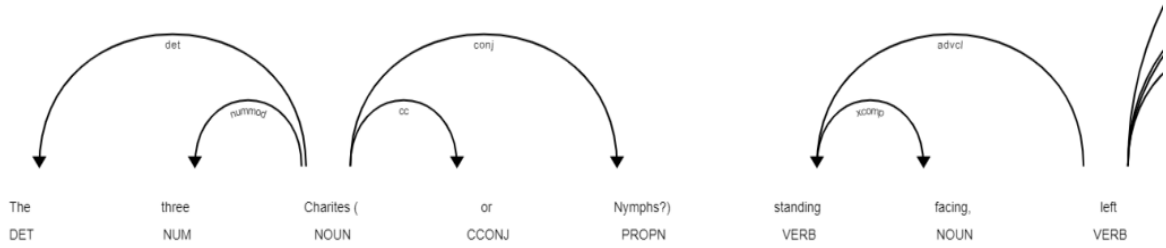

Abbildung 19: Ikonograph wird von spaCy in zwei Teile getrennt

Beispiel zu IV Ein weiteres Problem ist die NER Erkennung bzw. die manuelle Pflege der Tabellen und deren Alternativnamen und Schreibfehlern. In der folgenden Ikonographie wird das Objekt »*calathus*« nicht markiert, heißt der NER Prozess hat diesen nicht erkennen können. Nach Überprüfen der Datenbank, kommt »*calathus*« in dieser Schreibweise nicht vor, sondern nur als »Kalathos«, »Kalathoi« oder »*calathos*«.

> »Cybele enthroned left, wearing calathus, holding patera in outstretched right hand over lion in front of her, seated left, resting left arm on tympanum.«[54]

Dasselbe Problem ist auch bei »grain ear« im Design mit der DesignID 343 aufgetreten.
In der letzten Ikonographie aus der Stichprobe wird eine falsche Entscheidung vom Modell getroffen.

> »Dionysus seated right on panther advancing right, holding long thyrsus in left arm, resting right arm on panther.«[55]

Die *Ground Truth* besteht aus

> [(Dionysus, PERSON, seated_on, panther, ANIMAL), (Dionysus, PERSON, holding, thyrsus, OBJECT), (Dionysus, PERSON, resting_on, panther, ANIMAL)]

[54] DesignID = 325
[55] DesignID = 392

Das Modell erkennt zwei Relationen korrekt und eine Falsch:

> [(Dionysus, PERSON, resting_on, panther, ANIMAL), (Dionysus, PERSON, holding, thyrsus, OBJECT), (Dionysus, PERSON, resting_on, panther, ANIMAL)]

Das Modell schlägt beim ersten Vorkommen von »panther« die Relation »resting_on« vor, im Design tritt aber »seated_on« vor. Analysiert wird nun der Pfad und der Fehler ist auf **Beobachtung II.** zurückzuführen. Die Ähnlichkeit des gefundenen Pfades entspricht eher den Relationen mit »resting_on« als »seated_on«.

Die vollständige Stichprobe kann im Anhang betrachtet werden (»Stichprobe_englisch.xlsx«). Zusammenfassend ist zu erkennen, dass die Pflege der Datenbank bzw. die ständige Erweiterung dieser wichtig ist, um ein gutes NER Modell zu erhalten, denn dieses ist ein wichtiger Faktor in der Erkennung der Relationen bzw. als Voraussetzung für die Erkennung. Ein weiterer wichtiger Faktor ist die Erkennung der Abhängigkeiten im Satz, diese werden von *spaCys dependancy parser*, zwar sehr gut erkannt, zeigen jedoch Schwächen bei der Struktur der Ikonographen. Zuletzt ist die manuelle Arbeit ebenfalls sehr wichtig und für Fehler anfällig.

4.6 (NE, Verb) - Erweiterung

In diesem Abschnitt wird die Idee behandelt, die aus dem Analysieren der Ikonographen entstanden ist. Das Problem, dass durch diese Erweiterung abgedeckt werden soll, ist das viele Ikonographen kein Objekt im Satz besitzen. Als Beispiel können die folgenden zwei Ikonographen betrachtet werden.

»Horse prancing right.«[56]
»Dolphin swimming right.«[57]

[56] DesignID = 741
[57] DesignID = 419

4.6.1 Idee und Implementierung

Die oben gezeigten Ikonographen besitzen jeweils nur ein Subjekt im Satz und ein Verb. Das vorgestellte Modell erkennt hier keine Relation, da es nur auf Relationen zwischen Subjekt und Objekt ausgelegt ist. Diese Informationen sollen aber nicht verloren gehen, daher ist die Idee ein zweites Modell zu entwerfen, das sich mit der Erkennung von Subjekt und Verb Relationen beschäftigt. In den obigen Ikonographen wäre es das Pferd, das tänzelt oder der schwimmende Delfin. Um diese Relationen zu extrahieren, werden diese als Tripel der Form (NE, α, β) interpretiert, mit NE aus {PERSON, OBJECT, ANIMAL, PLANT}, α aus den zu klassifizierenden Verben und β aus den Klassifikationsklassen. In der folgenden Tabelle wird eine neue Klassifikation erstellt, diese unterscheidet sich insofern von der Klassifikation in **Kapitel 4.3**, dass hier alle Verben betrachtet werden, die in den Ikonographen vorkommen, nicht nur Verben zwischen einem Subjekt und Objekt. Die Tabelle spiegelt auch die erstellte Verbtabelle in der Datenbank dar und enthält neben Äquivalenzen auch andere Schreibweisen.

Klasse (β)	Semantisch Äquivalente bzw. in Ikonographen auftauchende (α)
holding	holding. covering, cradling, ploughing, removing, touching, raising, containing, forming, drawing, touching, carrying, brandishing
wearing	wearing
seated_on	seated, riding, galopping, galloping, throning, sitting
resting_on	resting, reclining, leaning, setting, leaned
feeding	feeding
standing	standing
escorted_by	escorted (by)
coiling	coiling, creeping, curling, creeps
lying	lying
advancing	advancing, running, passing, walking
swimming	swimming
extending	extending
receiving	receiving

prancing	prancing
flying	flying
leaping	leaping, jumping
crowning	crowning
grasping	grasping, clasping
kneeling	kneeling, scooping
pushing	pushing
crossing	crossing
sailing	sailing

Tabelle 7: Klassifikation der englischen (NE, Verb) - Erweiterung

Anhand der erstellten Klassifikation werden zum Evaluieren des Modells 1000 Ikonographen annotiert. Die Annotation hat die oben erwähnte Form (NE, α, β) und sieht auf den Ikonographen

> »Nude Apollo advancing right, wearing fluttering chlamys, drawing arrow in right hand from bow in left hand.«[58]

wie folgt aus:

> [(Apollo, advancing, advancing), (Apollo, wearing, wearing), (Apollo, drawing, holding)]

Um die Relation zu extrahieren, wird ein ähnliches Modell genutzt, wie in **Kapitel 4**. Es wird dieselbe Pipeline erstellt, mit dem Unterschied, dass nun statt zwischen Subjekt und Objekt ein Pfad gesucht wird, dieser zwischen Subjekt und der neuen Entität VERB gesucht wird. Hier spielt die neu hinzugefügte Entität VERB eine große Rolle, da wie bereits in **Kapitel 4.4** erwähnt, oft Verben, durch *spaCy*, als Nomen markiert werden. Diese liegt auch an der Struktur der Ikonographen, diese weicht von einem normalen Satz ab. Da dieses Modell sich auf Verben fokussiert und möglichst viele erkannt werden sollen, werden diese als

[58] DesignID = 39

neue Entität eingeführt und selbst markiert. Dies hat sich in **Kapitel 4.5.1** als sehr performant gezeigt und Verben werden sehr gut erkannt. Mit der guten Erkennung der Verben als Grundlage kann nun die Relation zwischen Subjekt und den gefundenen Verben extrahiert werden. Ziel ist es, den Pfad zwischen Subjekt und Verb zu untersuchen und das gefundene Verb seiner zu bestimmenden Klasse zuzuordnen.

4.6.2 Evaluation

Bevor auf die Performance eingegangen wird, wird zunächst erstmal betrachtet was von dem alternativen Modell erwartet werden kann. Die Idee eines alternativen Modells ist aus dem Analysieren des Datensatzes bzw. Annotieren der Ikonographen entstanden, da aufgefallen ist, dass Informationen wie »Eagle flying left.« verloren gehen. Vergleicht man die Klassifikation des Modells aus **Kapitel 4.3** und diesem, so ist zu erkennen, dass zusätzliche Verben erkannt werden.

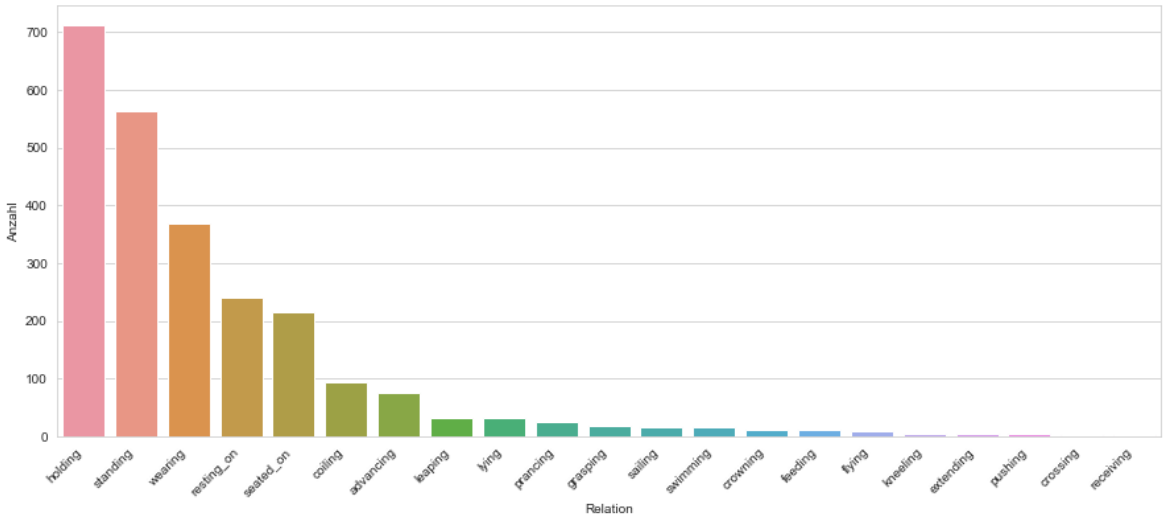

Abbildung 20: Relationsanzahl des annotierten Datensatzes (englisches Modell)

Beispielsweise tauchen »advancing«, »sailing« und »kneeling« nicht im Modell aus **Kapitel 4** auf (siehe **Kapitel 4.3**). Die drei genannten Verben sind nicht vorhanden, mit dem Hintergrund, dass diese nicht in einer Beziehung zwischen einem Subjekt und einem Objekt stehen. Auch zu erkennen ist, dass Klassen wie beispielsweise »standing« deutlich öfter vorkommen, da oft die Beschreibung »Person standing ...«, sprich einer stehenden Person, auftaucht, diese aber nicht in Relation zu Objekt steht. »standing« ist hier die zweitmeist

vertretende Klasse, dasselbe gilt für »advancing«, dass nicht im ersten Modell vorkommt. Zusammenfassend können nun einzelne Handlungen eines Subjektes von dem Modell erkannt werden. Dies ist ein großer Vorteil, um mehr Informationen aus den Ikonographen zu extrahieren.

Damit die bestmögliche Kombination des RE gefunden wird, muss ein *Gridsearch* durchgeführt werden. In der folgenden Abbildung ist zu sehen wie gut der neu eingeführte Parameter für *Path2Str* (siehe **Kapitel 4.4**) abschneidet. Die Kombination aus SVC, *Path2Str*(ent) und *CountVectorizer* mit einem ngram von 1,3 erreicht ein F-Maß von **89.9%**. Überraschenderweise performt hier der RF Klassifizier im Vergleich zum vorherigen Modell sehr gut, eine Kombination aus RF, *Path2Str*(ent) und *Countvectorizer*(1,2) erreicht ein F-Maß von 89.35% und ist dadurch das drittbeste Modell. Der LR Klassifizier mit derselben Kombination, bestehend aus LR, *Path2Str*(ent) und *CountVectorizer*(1,3), ist mit 89.56% das zweitbeste Modell, jedoch ist der LR Klassifikator hier nicht oft Vertreten. Die Klassifikator SVC und RF überwiegen dabei beim alternativen Modell.

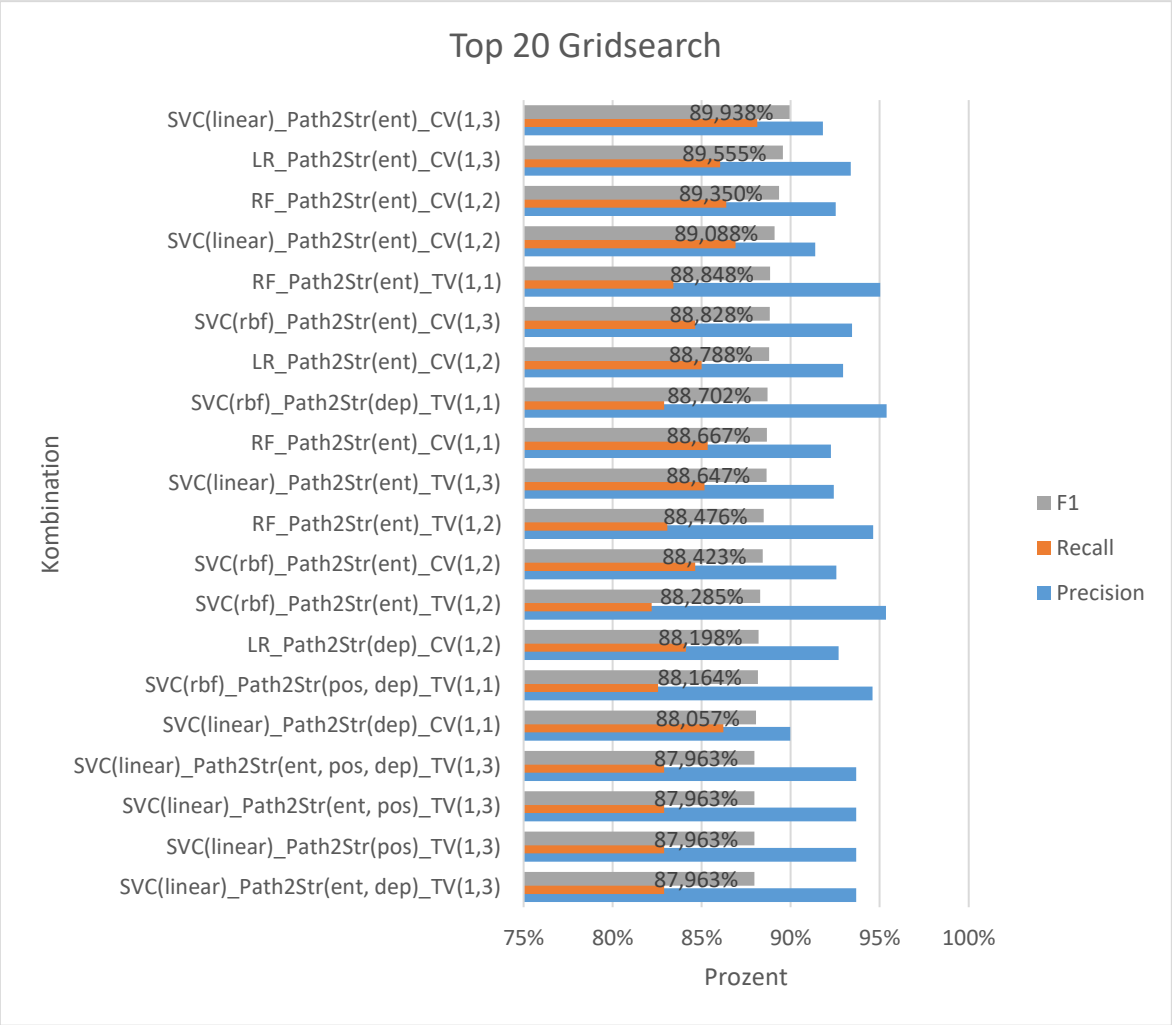

Abbildung 21: Gridsearch (NE, Verb) - Erweiterung

In Abbildung 22 sind alle möglichen Kombinationen zu sehen. Zu erkennen ist, dass sich hierbei ein großes Cluster bildet und die Klassifikatoren sehr gut abschneiden. Lediglich die Kombination aus *Path2Str*(pos) und dem *TfidfVectorizer* mit ngram 1,2 performt bei allen Klassifikatoren nicht sehr gut.

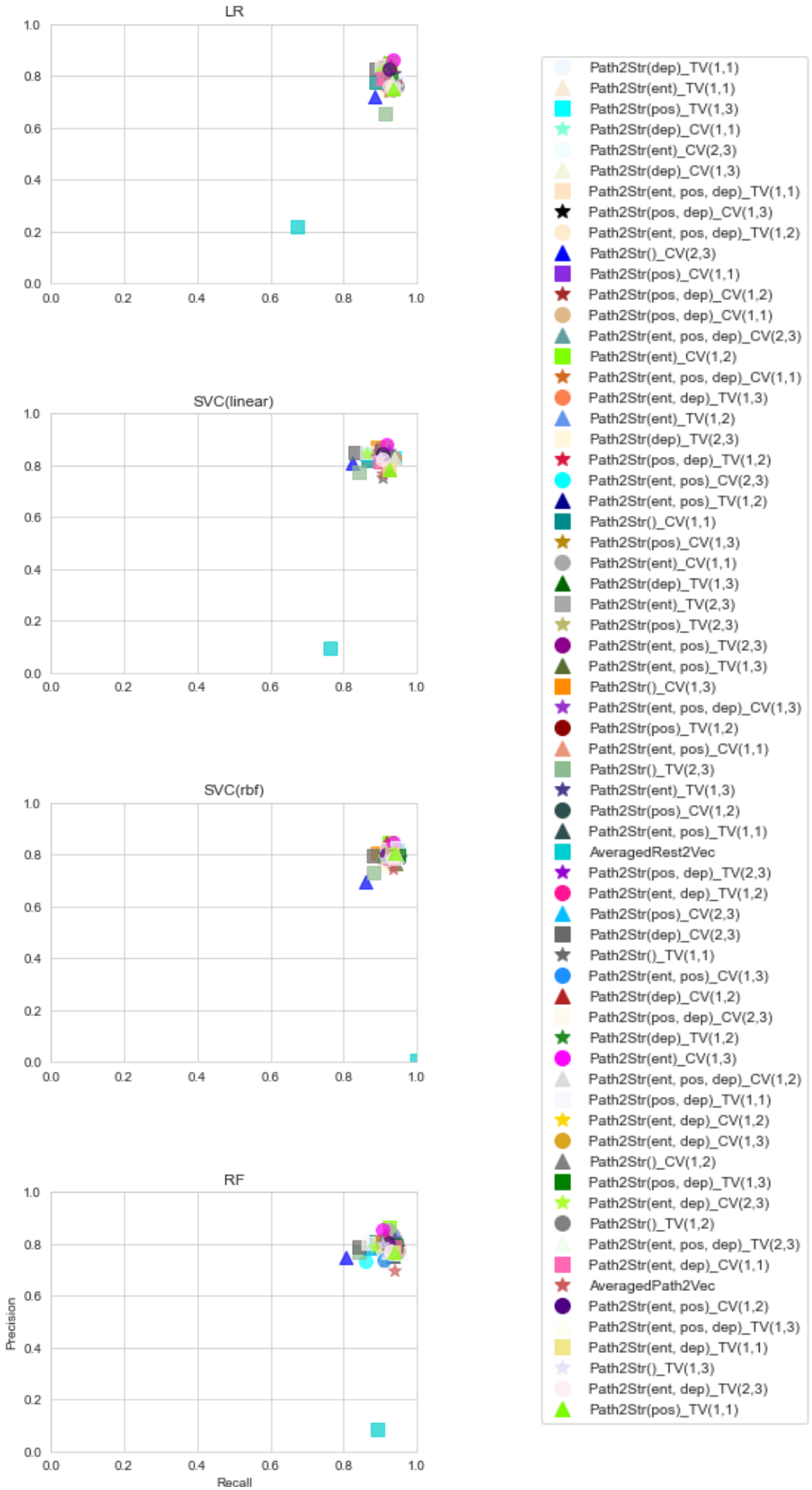

Abbildung 22: Performanceüberblick der verschiedenen Kombinationen (Erweiterung)

5. Das deutsche Modell

Nachdem das englischsprachige Modell erweitert und evaluiert wurde, ist es der zweite Teil dieser Ausarbeitung ein Modell zu implementieren, das die Ikonographen des deutschen Datensatzes verarbeiten kann. CNO verfügt hierbei für jede englische Ikonographie über eine deutsche Übersetzung – das heißt, dass es sich dabei, angenommenerweise, um inhaltlich exakt dieselben Beschreibungen handelt. Zum einen soll das NER realisiert werden, dass im Deutschen wie auch schon im Englischen Personen, Objekte, Tiere und Pflanzen erkennen soll. Zum anderen soll das RE, wenn nötig, angepasst werden, sodass auch Relationen im Deutschen zwischen jeweils zwei Entitäten als Tripelform (NE_1, α, NE_2) mit α als Bindungsglied erkannt werden können. Besonders für den letzteren Teil ist ein grundlegendes Verständnis über die sprachlich-grammatikalischen Unterschiede zwischen den beiden Sprachen von Nöten (siehe **Kapitel 3.5**). Nachdem das Modell vorgestellt wird, werden die Unterschiede in der Theorie daraufhin auf ihre praktische Relevanz untersucht und diskutiert. Daraus folgende Erkenntnisse werden vorgestellt. Abschließend wird das Kapitel mit einer erneuten Evaluation, diesmal für den deutschen Datensatz, abgeschlossen.

5.1 Implementierung

Die Vorgehensweise der Implementierung des deutschen Modells ist im Kern dieselbe wie des originalen englischen Modells. Zu aller Erst wurde das vortrainierte deutsche Modell heruntergeladen und installiert. Gewählt wurde das größere Paket »de_core_news_lg«[59] (Version 2.3.0). Das größere Paket besitzt zusätzlich zu den Trainingsquellen »TIGER Corpus« und »WikiNER« noch »OSCAR (Common Crawl)« und »Wikipedia (20200201)«. Kurz bedeutet dies, dass das Modell anhand einer größeren Grundlage trainiert wurde. Da die Ikonographen alle auch in deutscher Sprache vorliegen, mussten die Beschreibungen nicht vorbearbeitet werden. Bei den Entitätstabellen, bestehend aus PERSON, OBJECT, ANIMAL und PLANT musste für alle Tabellen, außer die für Personen eine manuelle

[59] https://spacy.io/models/de (13.09.20)

Übersetzung gemacht werden und in einer jeweils separaten Tabelle abgespeichert werden. Das nun bereits erweiterte englische Modell sollte nun äquivalent für den deutschen Datensatz realisiert werden. Das bedeutet, eine NER, die in den deutschen Ikonographen Personen, Objekte, Tiere und Pflanzen erkennen und markieren soll. Auf Basis dieser erkannten Entitäten sollen, wie auch schon im englischen Modell, Relationen zwischen Entitäten vorhergesagt bzw. extrahiert werden können. Hierbei bleibt der »Datenflow« der gleiche. Die Ausgabe des NERs sind Tripel, die die betrachtete Ikonographie und ein mögliches Subjekt plus Objekt mitführen. Diese Ausgabe dient auch als direkte Eingabe für die im Anschluss folgende RE- Pipeline. Die möglichen Subjekt-Objekt Paare, zwischen denen Relationen extrahiert werden sollen, sind hierbei (PERSON, OBJECT), (PERSON, PERSON), (PERSON, ANIMAL), (PERSON, PLANT), (ANIMAL, ANIMAL) und (ANIMAL, OBJECT). Die RE- Pipeline extrahiert dann aus der betrachteten Ikonographie und allen möglichen übergebenen Subjekt- Objekt Paaren die vorhandenen Relationstripel in der Form (Subjekt, Relation, Objekt).

Eine Ikonographie, die durch die NER- Pipeline läuft, wird nach den Entitäten die diese enthält markiert. So sind im Anschluss in folgender Ikonographie beispielsweise »Marc Aurel« als Person, »Panzer«, »Paludamentum« und »Standarte« als Objekt und »Pferd« als Tier markiert worden.

> »Kaiser (Marc Aurel **PERSON**) mit Panzer **OBJECT** und Paludamentum **OBJECT** nach rechts auf einem Pferd **ANIMAL** reitend, die Rechte erhoben; davor Soldat nach links stehend, in der Rechten Standarte **OBJECT** schräg haltend.«[60]

Jedes mögliche Subjekt-Objekt Paar dieser Ikonographie wird dann an die RE- Pipeline in folgender Form weitergegeben. Da die deutsche Sprache nicht nur Sätze zulässt, in denen das Subjekt einer Relation dem Objekt vorangeht, sind Konstellationen in alle Richtungen möglich:

[60] DesignID = 4615

[(Kaiser (Marc Aurel) mit Panzer und Paludamentum nach rechts auf einem Pferd reitend, die Rechte erhoben; davor Soldat nach links stehend, in der Rechten Standarte schräg haltend., Marc Aurel, Panzer),
(Kaiser (Marc Aurel) mit Panzer und Paludamentum nach rechts auf einem Pferd reitend, die Rechte erhoben; davor Soldat nach links stehend, in der Rechten Standarte schräg haltend., Marc Aurel, Paludamentum),

Und so weiter, mit den restlichen Kombinationen:

(Marc Aurel, Pferd), (Marc Aurel, Standarte), (Pferd, Marc Aurel),
(Pferd, Panzer), (Pferd, Paludamentum), (Pferd, Standarte)

Aus diesen ganzen potentiellen Relationen sollte die RE- Pipeline dann diese Subjekt-Objekt Paare mit ihrer Relation ausgeben, zwischen denen eine tatsächliche Relation besteht:

[(Marc Aurel, tragen, Panzer),
(Marc Aurel, tragen, Paludamentum),
(Marc Aurel, sitzen, Pferd)].

Hierbei ist »tragen« die Relation, die zwischen »Marc Aurel« und »Panzer« bzw. »Paludamentum« existiert und »sitzen« die Relation, die »Marc Aurel« und »Pferd« mit einander verbindet.

Auf diese beiden Pipelines, die aus dem englischen Teil schon bekannt sind, wird in den folgenden Abschnitten näher eingegangen. Da die beiden Pipelines jedoch weitreichend im **Kapitel 4** vorgestellt wurden, wird der Fokus in diesem Abschnitt eher auf die Neuerungen und Änderungen fallen, die wegen des deutschen Datensatzes gemacht werden mussten.

5.1.1 Named Entity Recognition

Wie im englischen Modell bekommt der NER die Entitätstabellen für Personen, Objekte, Tiere und Pflanzen um die Ikonographen annotieren zu können. Wie schon erwähnt,

mussten Übersetzungen der Tabellen für Objekte, Tiere und Pflanzen vorgenommen werden. Die Tabelle der Personen beinhaltete schon von Beginn an eine Spalte mit den jeweiligen deutschen Namen. Während eine »Artemis« auch im Deutschen »Artemis« heißt, gibt es einige Personen wie »*emperor*« (»Kaiser«), für die eine Übersetzung nötig ist. Die annotierten Ikonographen werden anschließend für das Training verwendet. Die Annotation besitzt dieselbe Form wie schon, die im bereits vorgestelltem Modell, eine Liste von Tripeln, die das Auftreten von Entitäten in der betrachteten Beschreibung darstellen.

> »Kaiser (Marc Aurel) mit Panzer und Paludamentum nach rechts auf einem Pferd reitend, die Rechte erhoben; davor Soldat nach links stehend, in der Rechten Standarte schräg haltend.«

Die Annotation für diese Beschreibung wäre somit:

> [(8, 18, PERSON), (24, 30, OBJECT), (36, 49, OBJECT), (74, 79, ANIMAL), (60, 69, OBJECT)].

Das Training selbst findet äquivalent wie im originalen Modell statt (siehe **Kapitel 4.3**). Der annotierte Datensatz wird in einen Trainings- und Testsatz geteilt (erneut mit *sklearn. model_selection.train_test_split* durchgeführt). Der Testsatz dient auch hier zum anschließenden evaluieren der Performance. Mit dem Trainingssatz wird durch drei Iterationen trainiert und im Anschluss wird das Können des NERs auf den Testsatz angewendet. Dabei werden auf dem nicht annotierten Datensatz Vorhersagen in Form von Enitäts-Tags getroffen. Die Auswertung erfolgt wie gewohnt durch den Vergleich der Vorhersagen mit dem *Ground Truth* (die zuvor erfolgten Annotationen). Auch im deutschen Modell können und sollten die *False Positive*- Vorhersagen manuell untersucht werden, da sich darunter tatsächlich korrekte, aber noch unbekannte, also neue Entitäten verbergen könnten – womit wiederum die Entitätstabellen erweitert werden können. Die Metriken, die zum Evaluieren genutzt wurden, sind im deutschen Modell logischerweise dieselben*: Precision, Recall* und das F-Maß (siehe **Kapitel 5.4**).

Wie man sieht, unterscheidet sich die Realisierung des NERs für das deutsche Modell nicht von der des englischen Modells. Die einzige Vorarbeit die nötig ist, ist eben

das Vorbereiten der Entitätstabellen in Form von Übersetzen in die richtige Sprache, sodass mit Hilfe der Annotation der Ikonographen ein geeigneter Trainings- und Testsatz erstellt wird.

5.1.2 Relation Extraction

Der Ansatz RE als ein Klassifikationsproblem zu betrachten, bleibt sprachunabhängig der gleiche. Wie schon im erweiterten englischen Modell werden hierbei nicht nur Relationen zwischen PERSON und OBJECT betrachtet, sondern auch PERSON zu PERSON, PERSON zu ANIMAL, PERSON zu PLANT, ANIMAL zu ANIMAL und ANIMAL zu OBJECT. Da in diesem Modell mit einer anderen Sprache als englisch gearbeitet wird und die alten Relationsklassen wie beispielsweise »*holding*« nicht mehr genutzt werden können, musste eine erneute Klassifikation gemacht werden. Nachdem 1000 deutsche Ikonographen manuell annotiert wurden, ergaben sich neue Relationsklassen.

Klasse	**Semantisch Äquivalent**
halten	halten, schwingen, schleudern, ausgießen, ziehen (Pfeil), pflücken, herführen (Kerberos), spannen (Bogen), spielen (Leier) , entfernen (Dorn aus Tatze), erheben, schultern, lesen (Ähren), hängen, führen, in, mit
tragen	tragen, bedecken, mit, im, in
stützen	stützen, ruhen, liegen, lehnen
sitzen	sitzen, reiten
bekränzen	bekränzen
stehen	stehen, fahren, in, auf
winden	winden, ringeln
füttern	füttern (Schlange), säugen (Romulus und Remus)
packen	packen, würgen, umfassen, schöpfen (Gold)
drücken	drücken (Hirschkuh zu Boden)
empfangen	empfangen (Apfel)
brechen	brechen (Kiefer)

fliegen	fliegen
no_existing_relation	

Tabelle 8: Klassifikation für das deutsche Modell

Die Relationsklassen entsprechen, wenn sie denn noch im Deutschen vorhanden waren, denen des englischen Modells. Einige der Bindeglieder bzw. Verben, die entweder ganze Relationsklassen bilden oder auch nur als semantische Äquivalente vertreten sind, verschwinden durch die Übersetzung in das Deutsche. So wird im englischen Modell »*holding*« auch durch die Begriffe »*car(r)ying*«, »*cradling*« und »*supporting*« abgedeckt. »*Car(r)ying*« wird in den deutschen Ikonographen als »tragen« übersetzt und ist somit kein semantisches Äquivalent von »halten«. Der Begriff »*cradling*« wird nicht als einzigartiges Verb ins Deutsche übersetzt. Im Deutschen hat es die Übersetzung »stützen« und fällt somit als semantisches Äquivalent in »halten« weg. Ähnlich ist es mit »*supporting*«, auch dieser Begriff wird ins Deutsche mit »stützen« übersetzt. »*Holding*« verfügt außerdem über »*containing*«, welches im Deutschen nicht einmal in ein Verb bzw. Partizip übersetzt wird. Stattdessen wird die Präposition »mit« genutzt. Im nächsten Abschnitt wird dies ausführlicher betrachtet. Die Klasse »*stepping_on*« verhält sich ähnlich und wird als »auf« in »stehen« klassifiziert. Die Relationsklassen »*resting_on*« und »*lying*« werden für das deutsche Modell gemäß ihrer Übersetzung (als »stützen« und »ruhen«) unter »stützen« untergebracht. »*Hurling*« wird als »schwingen«/ »schleudern« in »halten« gesteckt. Zu guter Letzt ist die Klasse »*escorted_by*« zu betrachten. Die Ikonographie die diese Relation beinhaltet, besitzt die Formulierung »[…]*, escorted by Nike on left*, […]«, welches im Deutschen wie folgt formuliert wurde: »[…]; links Nike stehend von vorn […]«. Da diese Relation in CNO nur einmal auftaucht, existiert sie für das deutsche Modell gar nicht. In der oben genannten Ikonographie wären die Relationen folgende:

> [(Marc Aurel, tragen, Panzer), (Marc Aurel, tragen, Paludamentum), (Marc Aurel, sitzen, Pferd)]

Ins Auge fällt, dass die Relationsklasse »tragen« als Wort in der Ikonographie gar nicht auftaucht. In der Arbeit von P. Klinger gibt es einen ähnlichen Fall. So sagt Sie in Ihrer Arbeit in Kapitel 4.2:

»In some cases, it was useful to annotate more relations than only those explicitly represented by verbs. For

"Nike in biga, right."

the corresponding annotation is

[("Nike", "standing", "biga")]

because "standing" is the relation used in all other similar designs depicting Nike in a biga and can thus be annotated although not being explicitly mentioned. «

Diese Argumentation von P. Klinger ist jedoch in diesem Fall nur die halbe Erklärung. Während »auf« und »in« (Biga) auch in diesem Modell zu der Relationsklasse »stehen« klassifiziert werden, gibt es die Präpositionen »mit« und »im« für Relationsklasse »tragen«, die an dieser Stelle völlig neu sind. Die Präpositionen »in« und »mit« bekommen je nach Beschreibung eine Mehrfachzuweisung und sind auch in »tragen« und »halten« vertreten. Der Ursprung für diese Entscheidung liegt zurück in der Formulierungswahl der Übersetzungen. Denn das Verb »*wearing*« wurde in den Ikonographen nicht einfach nur zu »tragen/d« übersetzt. Stattdessen entschied man sich für eine ganze Umformulierung des Satzes, sodass die Entitäten die vorher mit »*wearing*« gebunden wurden nun mit einer Präposition gebunden werden. Um dies vor Augen zu führen, hier eines dieser besagten Ikonographen:

»Head of Dionysus, right, wearing mitra and ivy wreath.«[61]

Während im Englischen die Relationen [(Dionysus, wearing, mitra), (Dionysus, wearing, ivy wreath)] offensichtlich der Relationsklasse »*wearing*« zuzuordnen sind, sieht es in der deutschen Übersetzung anders aus:

[61] DesignID = 389

»Kopf des Dionysos nach rechts mit Efeukranz und Mitra.«

Das »*wearing*« bzw. »tragen« verschwindet aus der Ikonographie, stattdessen wird das »mit« zum Bindeglied zwischen Dionysos und dem Efeukranz oder der Mitra und muss deshalb zu der Relationsklasse »tragen« klassifiziert werden, um die semantischen Information nicht zu verlieren. Davon betroffen ist auch in seltenen Fällen »halten«. Genauer wird auf diese Problematik im nächsten **Kapitel 5.3.1** eingegangen.

Wie schon erwähnt, gibt es die Präpositionen »in« und »mit«, die eine Mehrfachzuweisung aufweisen und abhängig vom Kontext mehr als nur zu einer Relationsklasse zuzuordnen sind. Damit beim Klassifizieren keine Fehler unterlaufen, sind die »in«- Fälle eindeutig geregelt. So haben alle Ikonographen, bei denen »in« zu »halten« klassifiziert wird, dieselbe Form. Es handelt sich hierbei immer um Sätze mit der Formulierung: »in der Rechten/Linken NE_2«. Das heißt bei dieser Art von Formulierung ist davon auszugehen, dass die in der Relation eingeschlossene Entität, in der rechten bzw. linken Hand ist, sprich: gehalten wird. Die CNO- Datenbank enthält 1262 Ikonographen mit solch einer Formulierung. Für die restlichen »mit«- Fälle wurde für jede annotierte Entität eine sinnvolle Zuweisung bestimmt, die für alle zukünftigen Annotationen als Richtwert dient. Dabei wurden die Entitäten wie folgt aufgeteilt:

Relationsklasse	Zugeordnete Entitäten
halten	Bogen, Lorbeerzweig, Kerykeion, Geldbörse, Patera, Ähre, Fackel, Kalathos, Gegenstand, Weinrebe, Thyrsos, Kantharos, Zepter, Keule, Apfel, Weintraube, Tympanon, Hirtenstab, Schilfzweig, Füllhorn, Parazonium, Steuerruder, Pedum, Speer, Schlangenstab, Adlerzepter, Leiter, Blitzbündel, Traube, Standarte, Schriftrolle, Dreizack
tragen	Lorbeerkranz, Gewand, Stiefel, Paludamentum, Mauerkrone, Schleier, Mitra, Panzer, Löwenfell, Strahlenkrone, Efeukranz, Kriegsbekleidung, Diadem, Schuppenpanzer, Petasos, Cestus, Chlamys

Tabelle 9: Klassenzuweisung von Entitäten aus »mit«- Relationen

Das Analysieren des Aufkommens der Relationsklassen im Englischen zeigte, dass »*holding*« und »*wearing*« im Vergleich zu den anderen Klassen stark überwiegen (siehe **Kapitel 4.3**). In Anbetracht der Tatsache, dass im Deutschen »tragen« grundsätzlich unterschiedliche bzw. zusätzliche semantische Äquivalente hat als »*wearing*«, ergibt sich für das deutsche Modell folgende Verteilung:

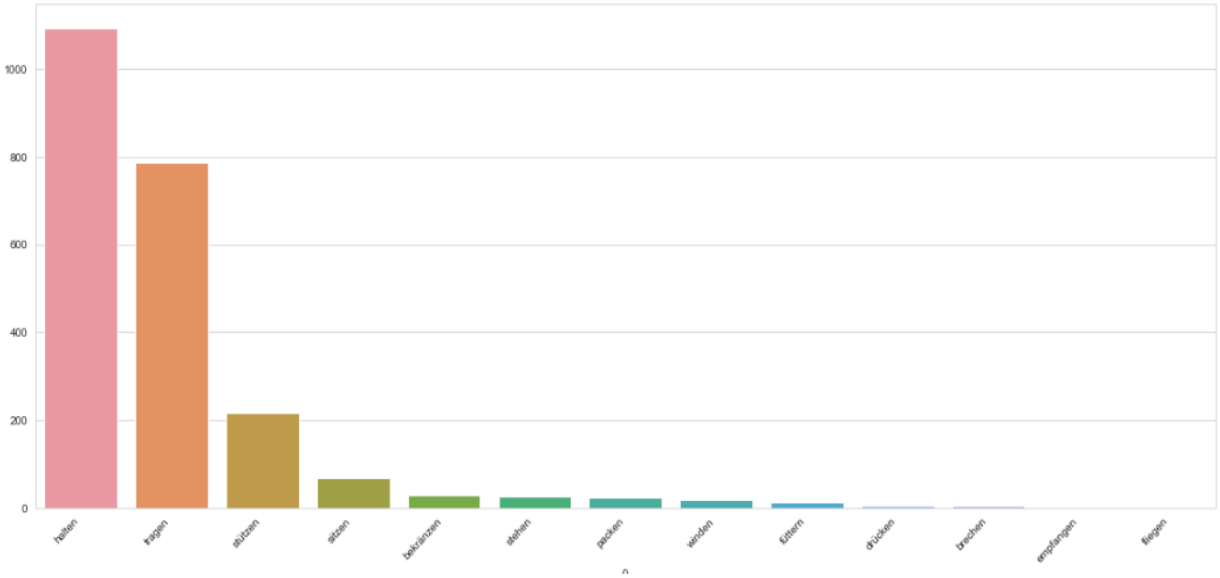

Abbildung 23: Verteilung der Klassen im genutzten Datensatz (deutsches Modell)

Es ist deutlich zu erkennen, dass auch beim deutschen RE die Relationsklassen »halten« und »tragen« stark überwiegen. In den 1000 annotierten Ikonographen kommen 1099 Mal »halten« und 786 Mal »tragen« vor. Die nächst größte Klasse wäre wie schon im englischen das »*resting_on*« (234 Mal im englischen RE) das »stützen« mit 216 Aufkommen. Im Vergleich: Beim englischen RE mit 1000 Ikonographen wurden 1063 Mal »*holding*« und 723 Mal »*wearing*« Relationen gefunden. Während im englischen Modell »*wearing*« aus zwei semantischen Äquivalenten besteht, nämlich »*wearing*« und »*covered_with*«, beinhaltet das »tragen« im Deutschen fünf Äquivalenten. Darunter die drei erwähnten Präpositionen »mit«, »im« und »in«.

5.2 Gewonnene Erkenntnisse

In diesem Abschnitt wird das vermittelte Grundwissen aus **Kapitel 3.5** über die sprachlichen Unterschiede auf ihre Relevanz in dieser Ausarbeitung untersucht. Herausforderungen und

Probleme, auf die während des Umsetzens der Implementierung gestoßen wurde, werden kategorisch vorgestellt. Ihr Wirken auf diese Arbeit wird beleuchtet und, falls vorhanden, eine daraus folgende Erkenntnis formuliert. Dabei bleibt das Augenmerk hauptsächlich auf der Datenqualität, welche es zu analysieren und bewerten gilt.
Des Weiteren werden notwendige Schritte gezeigt, die als Vorbereitung, also *Preprocessing,* gemacht werden mussten, damit das Modell reibungslos in Betrieb genommen werden konnte.

5.2.1 Datenqualität und Herausforderungen

Bevor die Herausforderungen die aufgetreten sind gezeigt werden, sollte man sich zuerst die Art der Formulierung der deutschen Ikonographen ansehen. Insgesamt steht zu allen 5542 englischen Ikonographen jeweils eine deutsche Übersetzung zur Verfügung. Zur Veranschaulichung wird folgende Beschreibung gewählt:

(1)

> »Half-nude Aphrodite standing facing, head left, holding apple in raised right hand.«
> »Halbnackte Aphrodite stehend von vorn, Kopf nach links, Apfel in der erhobenen Rechten haltend.«[62]

Ganz grundlegend ist zu erkennen, dass die Ikonographen des deutschen Datensatzes genauso, wie die des englischen im Präsens formuliert wurden. Die standardisierten Kriterien zum Formulieren der Ikonographen (siehe **Kapitel 4.2**) wurden im Deutschen äquivalent eingehalten. Dazu ist zu sagen, dass bei der Entscheidung der Wortwahl als auch beim Formulieren der Übersetzung eine gewisse Freiheit besteht, welche unmittelbaren Einfluss auf die Performance des Modells hat. Bevor jedoch über diese Problematik gesprochen wird, wird über die Form des Verbes bzw. der Adjektive der Ikonographen diskutiert – denn hier hat man sich im generellen für Partizipien entschieden.

[62] DesignID = 33

Das Partizip Das Wort Partizip kommt vom lateinischen Wort »*particeps*«, was so viel wie »teilhabend« bedeutet (Imo 2016, Kap. 4.1.4). Genau das tun Partizipien auch, sie sind Wörter, die gleich an zwei Wortarten teilhaben – Verben und Adjektive. Partizipien werden aus Verben gebildet und wie Adjektive verwendet. In der oben erwähnten Ikonographie (1) wären die Partizipien die Wörter »stehend« und »haltend«. Konkreter befinden sich diese Worte im Partizip I, auch Partizip Präsens genannt. Das Partizip I wird gebildet, indem man an den Verbstamm das Suffix -end anhängt. Das Partizip I kann verwendet werden, wenn das Verb eine aktive Handlung, die in der Gegenwart stattfindet, beschreibt.[63] Dies ist eben der Fall bei Beschreibungen von Kunstobjekten, worunter auch Münzen zugeordnet werden können, die in der Tat oft eine aktive Handlung abbilden. Das Englische Pendant ist das »*present participle*«. Dieses wird im Englischen ausnahmslos immer mit der Endung -ing gebildet. In der englischen Ikonographie (1) wären dies die Wörter »*standing*« und »*holding*« (Wagner 2009, Kap. 7). Doch das Partizip I ist nicht das einzige Partizip womit wir es bei den Ikonographen zu tun haben, das gilt jedenfalls für die englischen Ikonographen. So wird Beispielsweise immer von »*veiled*« gesprochen, wenn es darum geht eine Person mit Schleier zu beschreiben.

(2)

»*Veiled head of Demeter, right, wearing corn wreath.*«[64]

»*Veiled*«, die »*past participle*« Form des Verbes »*to veil sth*«, gebildet mit der Endung -ed, steht in direkter Verbindung zu einem Substantiv und nimmt nur noch die Rolle eines Adjektivs ein. Sowohl *present participle* als auch das *past participle* können im Englischen verwendet werden, um Sätze, die sich auf ein und dasselbe Subjekt beziehen, zu kürzen (Wagner 2009, Kap. 7). Interessant ist hierbei jedoch die deutsche Übersetzung von (2) zu betrachten. Die wurde wie folgt übersetzt:

»Kopf der Demeter nach rechts mit Ährenkranz und Schleier.«

[63] https://learngerman.dw.com/de/partizip-i-1/gr-39131663 (16.09.2020)
[64] DesignID = 342

Bei der Übersetzung hat man sich dafür entschieden, »*veiled*« nicht mit »verschleiert« zu übersetzen, sondern stattdessen »mit Schleier« zu verwenden. Es existieren 129 Sätze, die das Wort »*veiled*« beinhalten – 126 der deutschen Ikonographen nutzen an der Stelle die »mit Schleier« Übersetzung (die restlichen 3 »*veiled*« gehen dank schlechter Übersetzung verloren und werden gar nicht übersetzt). In den deutschen Ikonographen wird, unabhängig der englischen Beschreibung, oft auf das Partizip II verzichtet, einer der wenigen Ausnahmen bildet das Wort »gestützt« (tritt 477 Mal auf). Über die Veränderungen innerhalb der Übersetzungen wird im Abschnitt »Die Wortwahl der Übersetzung« mehr diskutiert.

Aus diesen Beobachtungen lässt sich die Erkenntnis schließen, dass die englischen und besonders die deutschen Ikonographen in den seltensten Fällen wirklich über ein Verb verfügen. Durch die »Veradjektivierung« der Verben ins Partizip II verfügen die Ikonographen strenggenommen nur noch über beschreibende Adjektive und im seltensten Fall über Verben. So existieren nur 30 Ikonographen, bei denen das Verb »halten« im Präsenz konjugiert als halt/hältst/hält/halten/haltet/halten vorkommt. Im Vergleich: »haltend« kommt in ganzen 1245 Ikonographen vor. Die daraus resultierende Annahme, dass die Leistung des REs unter der Tatsache, dass die meisten Ikonographen nicht über Verben im eigentlichen Sinne verfügen, leidet, ist jedoch falsch. Dies hängt mit der Funktionsweise der für das RE verwendeten Path- Funktion bzw. dem *DependencyParser* zusammen (siehe **Kapitel 5.4**).

Verben Wie schon im **Kapitel 3.5** angesprochen, bestand im Deutschen die Gefahr, dass die Ikonographen »auseinandergerissene Verben« bzw. mehrteilige Verben besitzen. Um diese Gefahr zu bewerten, wurden die Ikonographen durchsucht und analysiert. Dabei ergab sich, dass in keinem der Beschreibungen Modalverben auftauchen. Konkret bedeutet dies, dass wir keine Sätze haben mit einem der Modalverben »dürfen, können, mögen, müssen, sollen und wollen« + einem Vollverb (Imo 2016, Kap. 5). Übrig bleiben die Hilfsverben (Auxiliarverben). Zum einen hätten wir das Auxiliarverb »werden«, welches vier Mal in der Form »wird« + einem Vollverb auftaucht und die Genus-Verbi-Form des Passivs bildet. Eines der Ikonographen wäre Folgende:

»Kaiser (Gordian) in Kriegsbekleidung nach links mit Patera und Zepter; wird von hinter im stehender Nike bekränzt, zu seinen Füßen Altar.«[65]

Ein weiterer Fall, ist das Auxiliarverb »sein« in der Form »ist«, welches mit dem Vollverb zusammen das Tempus Perfekt markiert. Ikonographen in dieser Form kommen insgesamt 7 Mal vor. Ein Beispiel wäre diese Beschreibung:

»Krieger auf einem Schiff stehend nach rechts, Kopf zurückgewandt, in der Linken Speer, die Rechte ist ausgestreckt; hinter ihm, ein Krieger, Brustbild der Athena auf Acrostolium, noch ein dahinter; Krieger mit Helm und Schild auf einem anderen Schiff sitzend nach rechts; im Felde links, Herold mit Trompete auf hohem Turm.«[66]

In keinem der Ikonographen, steht jedoch das Auxiliarverb »ist« + Vollverb als α in einer direkten Relation (NE_1, α, NE_2) und ist für diese Ausarbeitung somit irrelevant. Somit bleiben nur noch die erwähnten 4 Sätze mit der Genus-Verbi-Form des Passivs übrig, die betrachtet werden sollten. Denn der oben aufgeführte Satz, als auch die restlichen drei Beschreibungen, besitzen Relationen, auf die das Auxiliarverb direkten Einfluss hat. So ist es im vorgeführten Beispiel die Relation (Nike, bekränzen, Gordian), wobei das »wird« die Handlungsrichtung des Passivs, in dem Fall Gordian, angibt.

Wie schon im Abschnitt zuvor erkannt, ist die Formulierung und Struktur der Ikonographen des CNO größtenteils einheitlich erfolgt. Die standardisierte auftretende Form des Verbes ist das Partizip I. Das bedeutet, der Datensatz sollte durch und durch einheitlich gestaltet werden – im Sinne der Einheitlichkeit, aber in erster Linie auch im Sinne der Effektivität des Lernens des *Machine Learning* (siehe **Kapitel 6.1**).

Die Wortwahl der Übersetzung Ein Faktor dessen Einfluss nicht unterschätzt werden sollte, ist die Formulierungs- und Wortwahl der Übersetzungen. Wie bereits angeschnitten bildet die größte Gruppe der ikonographischen Unterschiede zwischen dem englischen und deutschen Datensatz, das häufige Wegfallen der Adjektive bzw. des Partizips II. Konkret bedeutet dies, dass im deutschen CNO- Datensatz 430 Mal »*laureate*«, 219 Mal »draped«,

[65] DesignID = 5203
[66] DesignID = 4876

129 Mal »*veiled*«, 105 Mal »*turreted*« und 85 Mal »*diademed*« wegfallen und stattdessen die Substantive, in diesem Fall alles Elemente von OBJECT, »Lorbeerkranz«, »Gewand«, »Schleier«, »Mauerkrone« und »Diadem« jeweils mit dem Bindewort »mit« hinzukommen. Anders ausgedrückt ist anzunehmen, dass in den deutschen Ikonographen somit annährend 900 anerkannte Objekte hinzukommen. Das bedeutet wiederrum auch, dass knapp 900 neue Relationen im Deutschen auftauchen. Die »mit« Relationen und den oben genannten Objekten werden in der Relationsklasse »tragen« untergebracht (NE_1, »tragen«, NE_2). Die dadurch hinzukommenden Relationen sind Ikonographen wie:

> »Veiled and turreted Cybele enthroned left, holding patera in outstretched right hand, left arm resting on tympanon; beside throne, lion lying left, second lion seen in background behind throne, lying left.«[67]

Mit den Relationen:

> [Cybele, holding, patera]
> [Cybele, resting_on, tympanon]

Die deutsche Ikonographie wäre:

> »Kybele mit Mauerkrone und Schleier nach links thronend, in der vorgestreckten Rechten Patera haltend, den linken Arm auf das Tympanon gestützt; auf beiden Seiten des Throns je ein Löwe nach links liegend.«

Zu den Relationen:

> [Kybele, halten, Patera]
> [Kybele, stützen, Tympanon]

Kommen zwei neue, trotz selber Ikonographie, hinzu:

[67] DesignID = 330

[Kybele, tragen, Mauerkrone]

[Kybele, tragen, Schleier]

Beim Übersetzen entschied man sich gegen das 1425 Mal auftretende »*wearing*«, das »mit«, »in« und »im« bilden zum Großteil alleine die Relationsklasse »tragen«. Das Wort »tragend« selbst kommt nur 12 Mal in den Beschreibungen vor.

Es gibt jedoch auch den Fall, dass beim Übersetzen eine neue Entität entstanden ist, welche aber nicht in den Entitätstabellen abgedeckt wird. Als Beispiel hätte man die Übersetzung folgenden Satzes:

»Infant Heracles kneeling right, struggeling with serpents.«[68]

Mit der deutschen Übersetzung:

»Heraklesknabe nach rechts kniend, mit zwei Schlangen kämpfend.«

Das Kind Herakles, im Englischen beschrieben als »*Infant Heracles*« und ins Deutsche übersetzt als »Heraklesknabe«. In der englischen Ikonographie findet das NER die Person »*Heracles*«. Das deutsche NER würde auch die Person »Herakles« jederzeit finden, doch als zusammengesetztes Substantiv- Form als »Heraklesknabe«, welches nicht in der PERSON-Tabelle vorkommt, wird das NER keine korrekte Vorhersage treffen. Das Wort kann und wird nicht als PERSON markiert. Hierrüber hinaus konnte dieses Phänomen außerdem in folgender Beschreibung festgestellt werden:

»Ram's head, left, within dotted square border; within incuse square.«[69]

Im Deutschen als:

»Widderkopf nach links im geperlten quadratum incusum.«

[68] DesignID = 4664
[69] DesignID = 996

»*Ram*« als auch »Widder« sind bereits bekannte Entitäten in den jeweiligen Tiertabellen. Doch durch die Wahl der Übersetzung und dem zusammensetzen der Substantive »Widder« und »Kopf« zu »Widderkopf« geht in der deutschen Ikonographie diese bekannte Entität des »Widders« verloren. »Widderkopf« ist weder als Entität bekannt, noch steht es als ein Alternativname in einer der Tabellen. Hätte man sich für eine der englischen äquivalenten Übersetzung entschieden, nach der Form »Kopf eines Widders […]«, hätte man an dieser Stelle kein Problem. Nach aktuellem Stand findet so also im Deutschen ein Entitätsverlust statt. Für »Widderkopf« und »Heraklesknaben« bedeutet dies alleine 16 verlorengegangene Entitäten.

Ein weiterer Fall der beobachtet werden konnte, ist eine uneinheitliche erfolgte Wahl der Übersetzung. Aufgefallen ist dies bei Ikonographen mit »Forepart of prancing horse«. So wurde beispielsweise die Ikonographie:

»Forepart of prancing horse, right. Border of dots.«[70]

mit

»Protome eines springenden Pferdes nach rechts. Perlkreis.«

übersetzt (18 Mal als »Protome eines springenden Pferdes«). Während man sich bei dieser Beschreibung:

»Two conjoined foreparts of prancing horses; above, monogram.«[71]

für

»Zwei miteinander verbundene Pferdeprotomen; darüber Monogramm.«

entschieden hat (20 Mal als »Pferdeprotom«).

[70] DesignID = 1514
[71] DesignID = 1493

Der Kasus und Plural Es wurde beobachtet, dass in manchen deutschen Ikonographen weniger Entitäten vom NER erkannt wurden, obwohl diese, meist übliche und bereits bekannte Entitäten waren. Bei genauerem Untersuchen konnte der Fehler auf die im Deutschen besonderen Kasusendungen zurückgeführt werden. Betrachtet man diese Ikonographie:

> »Three ears of corn in shape of a trident.«[72]
> »Drei Ähren in Gestalt eines Dreizacks.«

Im deutschen NER wurde der »Dreizack« nicht als OBJECT erkannt und markiert. Durch den Genitiv, der Frage »Wessen«, wird dem Wort der Suffix -s angehängt. Da jedoch das Wort in der Form »Dreizacks« nicht in der OBJECT Tabelle vorliegt und auch »Dreizacks« nicht als Alternativname für »Dreizack« hinterlegt ist, wird diese Entität nicht erkannt. Darüber hinaus wurde dies auch beim Wort »Pferdes« beobachtet. Damit das NER auch solche erkennt, gilt es diese händisch als Alternativnamen zu ergänzen. Wenn man diese Art von Fehlern dauerhaft vermeiden möchte, ist man gezwungen für alle Entitäten auch ihre deklinierten vier Fälle als Alternativnamen zu ergänzen. Das heißt für die Entität »Dreizack« müssten folgende Fälle abgedeckt werden:

	Singular	*Plural*
Nominativ	der Dreizack	die Dreizacke
Genitiv	des Dreizacks	der Dreizacke
Dativ	dem Dreizack	den Dreizacken
Akkusativ	den Dreizack	die Dreizacke

Tabelle 10: Deklinationen des Wortes »Dreizack«[73]

Dieselbe Aussage gilt auch für die Pluralformen aller Entitäten. Es gilt bereits als Standard, die Pluralform einer Entität in Alternativnamen hinzuzufügen. Im Fall der deutschen Sprache ist dies gegebenenfalls mit weiteren Deklinationen, die abzudecken sind,

[72] DesignID = 1436
[73] https://www.crodict.de/nomen/deutsch/Dreizack (16.09.20)

verbunden. So reicht es nicht nur »Dreizacke« als Pluralform zu hinterlegen, da der Dativ des Plurals »Dreizacken« lautet.

Die Wortstellung Wie schon in **Kapitel 3.5** ausführlich erklärt, beschränkt sich die deutsche Sprache nicht nur auf S-P-O Sätze. Der häufigste aufkommende Satz, der diese Wortstellung bricht, sind die Ikonographen, die »ringelnde Schlange« bzw. »windende Schlange« Ausdrücke beinhalten. Diese haben unter anderem folgende Form:

> »Athena nach links mit Speer und Patera; sie füttert eine sich um einen Olivenbaum ringelnde Schlange vor ihr aus der Patera.«[74]

Ikonographen in denen die Schlange als Subjekt erst nach dem Objekt aufgezählt wird, kommen 21 Mal vor. Wie Ikonographen mit Auxiliarverben bilden diese O-P-S Sätze die Ausnahme- bzw. Sonderfälle in CNO und entsprechen nicht dem üblichen Formulierungsmuster (siehe **Kapitel 6.1**).

5.2.2 Preprocessing

Das Übersetzen der Entitätstabellen für Objekte, Tiere und Pflanzen, als auch das Erweitern der Alternativnamen für Personen, mussten im Voraus durchgeführt werden. Diese Vorarbeit für das NER hat zwei Tage Arbeit, circa 16 Stunden, in Anspruch genommen. Es wurden hierbei auch neue Entitäten hinzugefügt. Die restlichen Anpassungen waren hauptsächlich fehlende Alternativnamen. Zu diesen gehörten im Grunde nur korrekte Kasusendungen der Substantive (siehe **Kapitel 5.3.1**), die in Ikonographen vorkamen, aber durch die Tabellen nicht abgedeckt wurden.

Für das RE musste eine ganz neue »*Golddata*«, also eine *Ground Truth*, erstellt werden. Insgesamt wurden 1000 deutsche Ikonographen manuell annotiert. Dies entsprach einem Arbeitsaufwand von vier Tagen, also etwa 32 Stunden.

[74] DesignID = 3357

Gridsearch Auch für das deutsche Modell wird per *Gridsearch* die beste Klassifikator-Feature Kombination ermittelt. Erneut wurde mit *cross-validated Gridsearch* über fünf Iterationen die beste Kombination berechnet. Dies nahm circa fünf Stunden in Anspruch (bei einem i5-8250U CPU @ 1,6GHz).

5.3 Analyse und Evaluation

In diesem Kapitel geht es darum, die Leistung des deutschen Modells zu evaluieren. Dies geschieht mit den in **Kapitel 3.3** vorgestellten Metriken. Hierbei wird in Kapitel 5.3.1 als erstes das NER analysiert und evaluiert. Im Anschluss wird in Kapitel 5.3.1 die Leistung des REs bewertet – die Evaluierung der besten Klassifikator- Feature Kombination für das RE wird im darauffolgendem *Gridsearch* Kapitel 5.3.3 behandelt.

5.3.1 NER Auswertung

Bevor das NER evaluiert wird, sollte man sich die Zahlen anhand einer kurzen Analyse bewusstmachen. Das NER kennt 383 Personen, 264 Objekte, 63 Tiere und 36 Pflanzen, welche alle von den jeweiligen Entitätstabellen gedeckt werden. Zum Evaluieren des NERs wird ein *train_test_split* über alle Designs ausgeführt, dabei wird mit dem *random_state =* 1 getrennt. Die Testgröße entspricht 25% des Datensatzes. Nach dreifacher Iteration des Trainings bzw. Tests, erzielt das NER folgende Ergebnisse:

Entität	Gesamt	Richtige Vorhersagen	(Falsche) Vorhersagen
PERSON	1179	1119	24
OBJECT	3109	3072	27
ANIMAL	390	374	10
PLANT	208	197	11

Tabelle 11: Ergebnisse des NERs auf den deutschen Datensatz - Die Anzahl Entitäten gesamt, wird mit der Anzahl der richtigen und falschen Vorhersagen gezeigt (wobei die falschen Vorhersagen FPs beinhalten)

Häufigster auftretender Entitätstyp in dem Testsatz ist OBJECT mit 3109 Aufkommen. Ein Drittel so oft werden Personen markiert – hier sind es insgesamt 1179 PERSONs. Tiere und Pflanzen kommen beide jeweils unter 500 Mal vor. Das Modell macht zwar beim Vorhersagen der Objekte mit 27 falschen Vorhersagen die meisten Fehler, jedoch ist zum einen zu bedenken, dass die falschen Vorhersagen *False Positives* beinhalten und somit potentiell richtige Entitäten darunter sein könnten und zum anderen bildet OBJECT eben die größte Entitätsgruppe, bei der 27 Fehler auf 3072 richtige Vorhersagen im Verhältnis sehr wenig sind. Von den 3109 vorhandenen Objekten werden 99,2% korrekt vorhergesagt.

Betrachte man den gesamten deutschen Datensatz bestätigt sich die Verteilung der Entitäten des Testsatzes (siehe Abbildung 24). Von 20210 markierten Entitäten sind 61,9% aus OBJECT, 26,2% aus PERSON gefolgt von ANIMAL mit 7,9% und PLANT mit nur 4%.

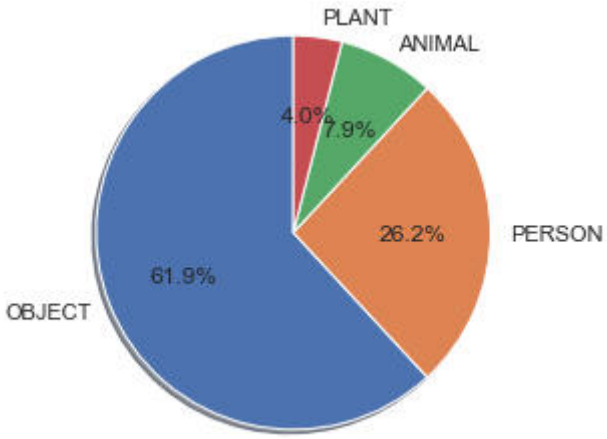

Abbildung 24: Die Verteilung der Entitäten auf den gesamten deutschen Datensatz

Im Fall der PERSON Entitäten ist der »Kaiser« die am häufigsten auftretende Person. Der Kaiser taucht 419 Mal in den CNO Ikonographen auf. »Apollon« wiederrum ist mit 329 Vorkommen auf Platz zwei und gleichzeitig der erste und häufigste Eigenname in den deutschen CNO Ikonographen. Darüber hinaus sind es elf weitere Eigennamen, die alle jeweils über 100 Mal erwähnt werden. Bei den Objekten hebt sich der erste Platz weiter vom Rest ab. »Kopf« kommt 1589 Mal vor und wird von dem Objekt »Brustbild« gefolgt, das gerade einmal 645 Mal auftaucht. Bei den Tieren ist es die »Schlange«, die mit einer Häufigkeit von 246, knapp vor dem »Pferd« mit einer Häufigkeit von 238 gelistet wird. In der kleinsten Entitätsklasse der Pflanzen, gibt es, ähnlich wie bei den Objekten, eine Entität die besonders raussticht. Die »Ähre« kommt 269 Mal vor.

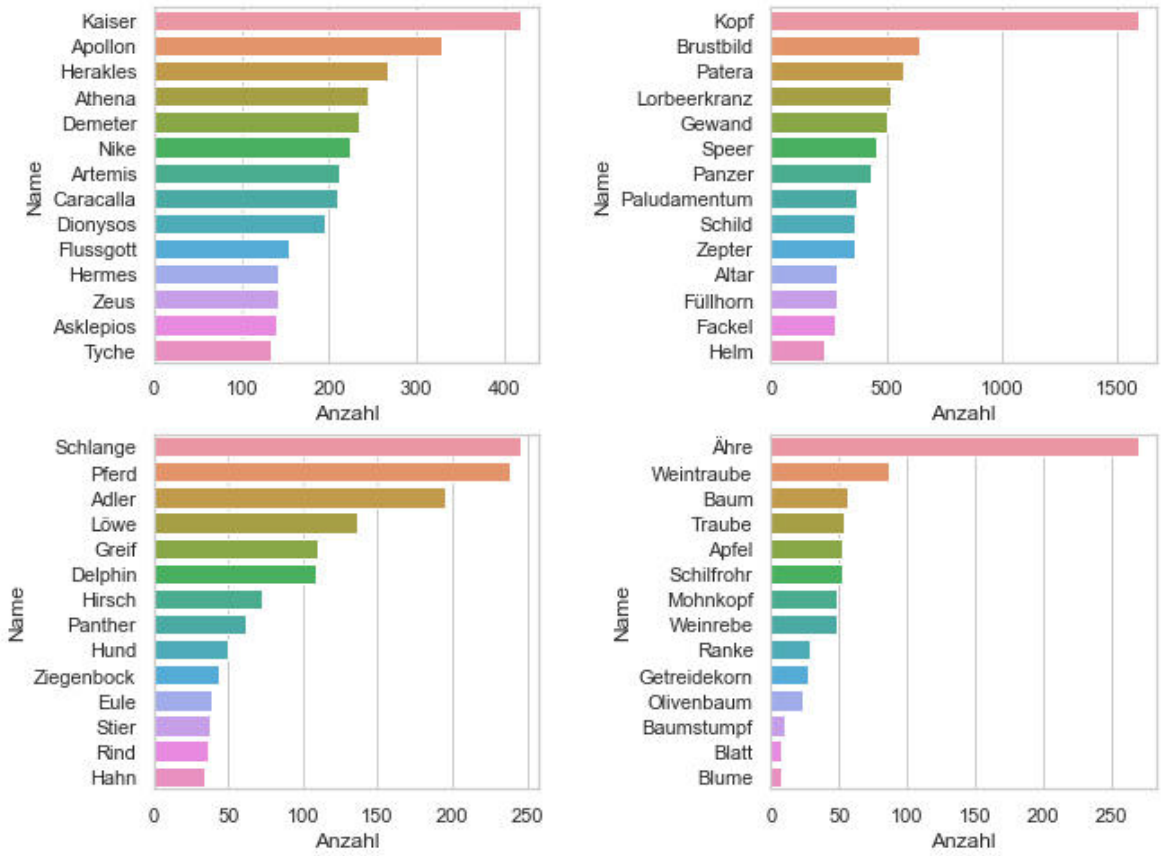

Abbildung 25: Top 15 der jeweiligen Entitätsklassen (deutsches Modell)

Um das NER Modell zu evaluieren, werden die Entitätsklassen mit dem Metriken *Accuracy, Precision, Recall und* F-Maß (F1) für sich selbst betrachtet evaluiert und im Anschluss das gesamte NER inklusive aller möglichen Entitätstypen. Dabei wurden folgende Ergebnisse gemessen:

Entität	Accuracy	Precision	Recall	F1
PERSON	96.6	98.1	94.7	96.4
OBJECT	99.2	98.2	99.2	98.7
ANIMAL	96.4	97.2	96.4	96.8
PLANT	94.7	88.2	96.6	92.2
Gesamt	98.2	97.6	97.8	97.7

Tabelle 12: Evaluation des Modells – Die Performance jeder Entitätsklasse für sich und Performance total (deutsches Modell)

Nach Tabelle 12 beträgt das F-Maß der NER Leistung des Modells aller möglichen Entitäten **97,7%**. Weiterhin erreicht das gesamte Modell eine *Accuracy* von **98,2%**, eine *Precision* von **97,6%**, und ein *Recall* von **97,8%**. Doch trotz dieser guten Leistung, gibt es selbst bei diesem Modell noch fehlende Vorhersagen von neuen Entitäten. Dies sieht man in der Vorhersage folgender Ikonographie:

»Nackter Hermes PERSON nach rechts mit Beutel und Kerykeion OBJECT.«[75]

»Hermes« als PERSON als auch »Kerykeion« als OBJECT wurden korrekt erkannt und markiert. »Beutel« wurde jedoch nicht erkannt. Deshalb gilt es auch im Deutschen die Vorhersagen des Modells händisch zu bearbeiten. Im Falle des »Beutel(s)« handelt es sich hierbei um kein *True Negative*, deshalb wäre es notwendig »Beutel« manuell in die Tabelle OBJECT hinzuzufügen.

Wie in **Kapitel 4.4** schon vorgestellt, wurde die Beobachtung gemacht, dass eine (NE, Verb) - Erweiterung nötig ist, welche Relationen abfängt die kein Objekt besitzen. Stattdessen werden Subjekte als Tupel mit Verben verarbeitet. Während es im Englischen das Problem gab, dass in den Ikonographen Verben fälschlicherweise als Nomen markiert wurden, gibt es im Deutschen das Problem, dass Verben als Partizip, also Adjektiv, markiert werden (siehe **Kapitel 5.2.1**). Um alle Verben also erkennen zu können, sodass die (NE, Verb) - Erweiterung mit diesen arbeiten kann, ist es auch hier nötig diese als eigenständige Entität zu implementieren. Schaut man sich die Entitäten inklusive der Verben an, so steigt die gesamte Anzahl der Entitäten durch die Verben um 5134. Hierbei werden Verben mit einer *Precision* von 98,37%, einem *Recall* von 98,73% und ein F-Maß von 98,56% markiert.

[75] DesignID = 5300

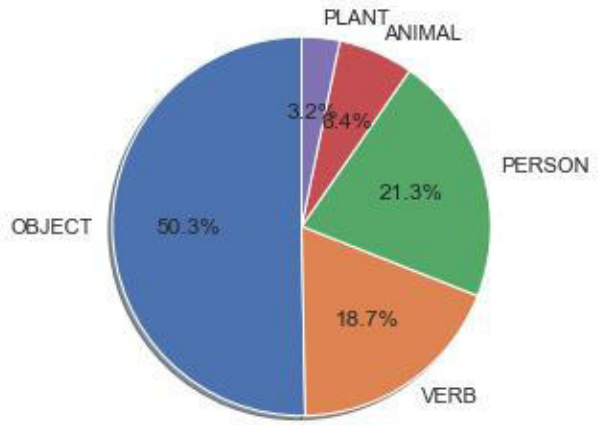

Abbildung 26: Die Verteilung der Entitäten inkl. Verben auf den gesamten deutschen Datensatz

5.3.2 RE Auswertung

Um das RE zu trainieren, wurden die *Ground Truth* Annotationen in einen Trainings- und Testsatz aufgeteilt. Annotiert wurden Relationstripel der Form (NE_1, α, NE_2) mit NE_1 ∈ {PERSON, ANIMAL, OBJECT}, NE_2 ∈ {PERSON, ANIMAL, OBJECT, PLANT} und α ∈ {halten, tragen, stützen, sitzen, bekränzen, stehen, winden, füttern, ausgießen, drücken, hängen, brechen, schöpfen, säugen, non_existing_relation}. Die häufigsten annotierten Relationen sind »halten« und »tragen«, mit einem Aufkommen von 1099 bzw. 786.

Auf 25% der *Ground Truth* Annotationen wurde das RE des Modells letztendlich getestet. Genutzt wurde die durch den *Gridsearch* ermittelte Klassifikator- Feature Kombination. Diese besteht aus *Path2Str* (mit pos=True) als *Stringconverter*, als *Vectorizer* der *CountVectorizer* mit *ngram* = (2,3) und einem *SVM* (der *Support Vector* Classifier) als Klassifikator. Die dabei gemachten Vorhersagen erzielten folgende Ergebnisse: die *Precision* erreicht **89,5%**, der *Recall* beläuft sich auf **84,6%** und das F-Maß beträgt **86,9%**.

Wie bereits vorgestellt, arbeitet das RE auf Basis der *path*- Funktion, welche die Informationen des *DependencyParsers* von spaCy verarbeitet (siehe **Kapitel 4.3**). Die entferntesten Vorfahren von Subjekt und Objekt des Satzes werden bestimmt, um das Bindeglied, in den meisten Fällen ein Verb in Partizip II oder eben einer der akzeptierten Präpositionen (»mit«, »in« und »im«), zu ermitteln. Nehme man erneut die Ikonographie:

»Demeter nach links thronend, in der Rechten zwei Ähren haltend, den linken Arm auf lange Fackel gestützt. Bildleiste.«[76]

So sieht der durch *displaCy* generierte Abhängigkeitsbaum wie folgt aus:

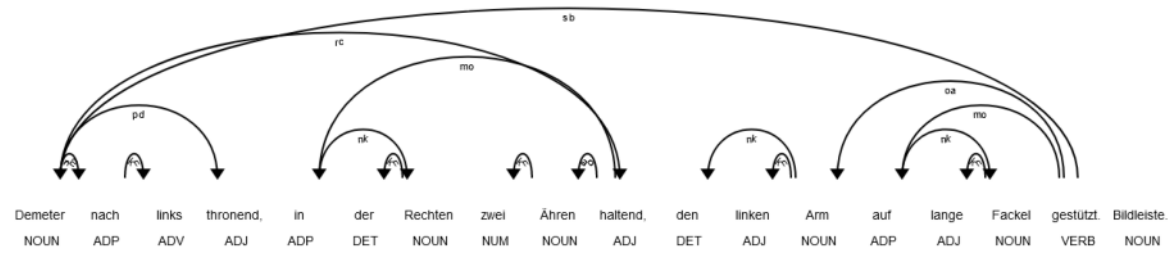

Abbildung 27: Abhängigkeitsbaum

In dieser Ikonographie wären die möglichen Subjekt-Objekt-Kombinationen die betrachtet werden: {Demeter, Ähren, Fackel} x {Demeter, Ähren, Fackel}. Betrachte man nun »Demeter« als Subjekt und »Fackel« als Objekt, so würde die *path*- Funktion folgenderweise vorgehen: Als Erstes wird das Objekt »Fackel« betrachtet, von diesem aus wird Vorfahre für Vorfahre, in Form von eingehenden Kanten, zurückverfolgt. Der entfernteste Vorfahre ist gefunden, wenn der aktuell betrachtete Knoten keine eingehende Kante mehr hat. Im Falle der »Fackel« wäre dies: »Fackel« ← »auf« ← »gestützt«. Im Anschluss wird dasselbe auf das Subjekt »Demeter« angewendet: »Demeter« ← »gestützt«. Der entfernteste Vorfahre vom Subjekt und Objekt ist in diesem Beispiel »gestützt«. Das Rückverfolgungsergebnis wird in einer Liste notiert und die Ausgabe von *path* ist:

[Demeter, gestützt, auf, Fackel]

Dies wird analog wie im Englischen in der Pipeline weitergegeben und verarbeitet, sodass das RE des Modells auf dieser Basis folgende Relationsvorhersage treffen kann:

(Demeter, stützen, Fackel)

[76] DesignID = 1667

Äquivalent würde man die Vorfahren auch bei der Kombination »Demeter« und »Ähren« untersuchen. Dort ist die *path* Ausgabe: [Demeter, haltend, Ähren] zu der die Vorhersage (Demeter, halten, Ähren) getroffen wird.

Um die beste Klassifikator- Feature Kombination zu ermitteln, wird ein *cross-validated Gridsearch* ausgeführt. Getestet wird mit fünf Iterationen über alle möglichen Klassifikatoren und Features. Die Resultate des *Gridsearch* werden im Folgenden vorgestellt.

5.3.3 *Gridsearch*

Die RE Performance wird, wie bereits bekannt, durch die NER Leistung limitiert. Das NER erzielt eine *Accuracy* von 98,2%, *Precision* von 97,6%, *Recall* von 97,8% und ein F-Maß von **97,7%.** Die beste Performance die das RE erzielt sind *Precision* mit **89,55%**, *Recall* mit **83,84%** und das F-Maß mit **86,60%**.

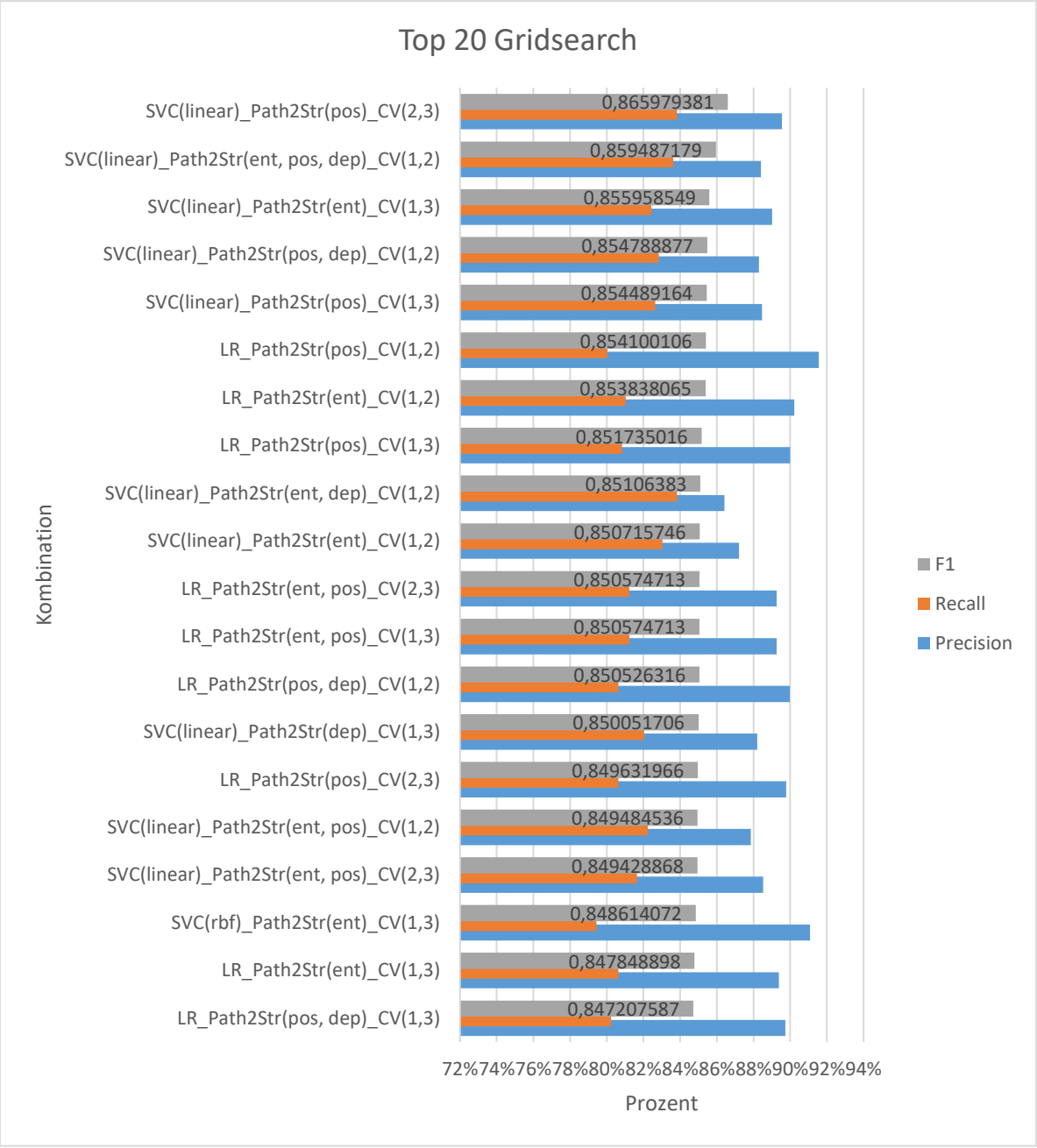

Abbildung 28: Top 20 Kombinationen (F-Maß absteigend, deutsch)

In Abbildung 28 sind die Top 20 Kombinationen gemessen am F-Maß abgebildet. Als beste Kombination bewährte sich als Klassifikator den *Support Vector Classificator* SVC mit linearem Kernel zu nutzen, kombiniert mit den Features *Path2Str* mit PoS-Tags und *CountVectorizer* mit BoW ngram = (2,3). An zweiter Stelle schneidet die selbe Kombination ab, jedoch mit *CV* ngram = (1,2) und den zusätzlichen *Path2Str* Hyperparametern ent und dep. Die Performance beträgt 88,40% *Precision,* 83,63% *Recall* und 85,95% für das F-Maß. Erst an sechster Stelle kommt als Klassifikator die logistische Regression zur Verwendung.

Die Logistische Regression mit den Features *Path2Str* und BoW mit ngram = (1,2) erzielt die höchste *Precision* der Top 20 Kandidaten mit 91,55%, einem *Recall* von 80,04% und einen F-Maß von 85,41%. Auch hier ändert die Größe des ngrams die Performance nur minimal zum Schlechteren. Mit logistischer Regression scheint ngram = (1,2) besser abzuschneiden als ngram = (1,3). Im Allgemeinen lässt sich erkennen, dass das Feature *Path2Str* auch für das deutsche Modell eine gute Leistung erzielt. Dabei scheint der PoS-Tag als alleiniger Hyperparameter besser zu fruchten. Der in dieser Arbeit implementierte ent Hyperparameter, durch welchen alle Entitäten inklusive Verben markiert werden, bewährt sich als zweitbester Hyperparameter für *Path2Str.* Während dep- Tags alleine nicht schlecht abschneiden (max. F-Maß von 85%), erreichen PoS- Tags (86,6% F-Maß) oder ent- Tags (85,6% F-Maß) bessere Leistungen. Als *Vectorizer* wird fast ausschließlich der *CountVectorizer* genutzt. TF-IDF bzw. TV, welcher zu keinem der Top 20 Kombinationen gehört, erzielt im besten Fall mit SVC(‚linear') und *Path2Str* eine Maximalleistung von bis zu 84% für das F-Maß, aber dafür eine *Precision* von bis zu 100% (dann aber ein F-Maß von knapp 2%). *Doc2Str* als Feature ist nicht die Erwähnung wert, ein F-Maß von mehr als 3% wird nicht erreicht.

Bezüglich der Klassifikator-Frage lässt sich erkennen, dass der SVC(‚linear') für das deutsche Modell am besten geeignet ist. Als zweitbester Klassifikator eignet sich die logistische Regression. SVC(‚rbf') an dritter Stelle erreicht einen F-Maß von bis zu 84,9% mit einer guten *Precision* von 91%. Der *Random Forest Classifier* in Kombination mit *Path2Str*(ent, pos) und CV(1,3) bewegt sich im Mittelfeld mit den Leistungen 88,6% *Precision*, 77,6% *Recall* und 82,8% F-Maß.

Alle möglichen Kombinationen sind in folgender Abbildung zu sehen. Dabei wird jeder Klassifikator getrennt. Je Klassifikator sind deutlich je zwei Clusterbildungen zu erkennen. Eine eher effektivere Feature-Gruppe und eine eher schlechtere.

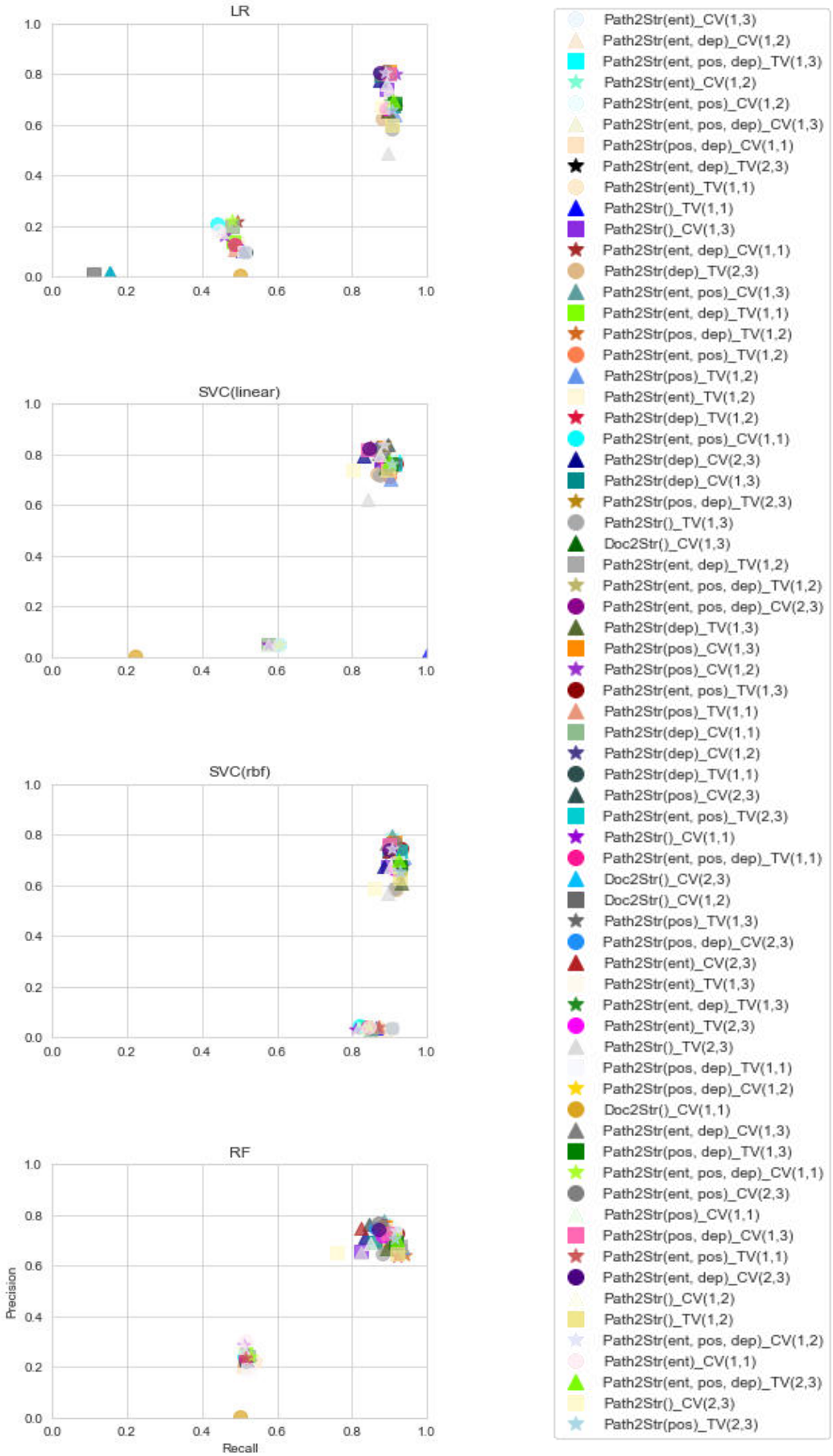

Abbildung 29: Performanceüberblick der verschiedenen Kombinationen (deutsch)

5.3.4 Stichprobe

In diesem Teil wird per Stichprobe das Modell bzw. die Vorhersagen des Modells im konkreten betrachtet. Man nehme 50 Ikonographen die äquivalent denen aus der Stichprobe des englischen Teils entsprechen (siehe **Kapitel 4.5.3**). Auf die Strichprobe angewendet erzielt das Modell eine *Precision* von 96,6%, einen *Recall* von 92,7% und ein F-Maß von 94,6%. Im Folgenden werden die »interessanten« Fälle, also diese, dessen Vorhersage nicht dem *Ground Truth* entspricht, vorgestellt. Insgesamt entsprachen **sechs Vorhersagen** nicht der *Ground Truth* Annotation. Diese Fälle lassen sich in drei Beobachtungen zusammenfassen:

I. Fehlende bzw. schwache Abhängigkeitserkennung
II. Falsche Vorhersagen des Modells
III. Menschliche Fehler beim Erstellen der Annotation

In der englischen Stichprobe wurde als IV. Beobachtung (siehe **Kapitel 4.5.3**) ein Fall erläutert, in dem eine fehlende Entität gefunden wurde. Konkreter geht es dabei um »*grain ear*«. In der deutschen Ikonographie ist aber die Rede von der »Ähre«, dementsprechend konnte im Deutschen nicht eine ähnliche Beobachtung festgestellt werden.

Beispiel zu I Eine Ikonographie, zu der keine Vorhersage bezüglich ihrer Relationen gemacht wird, ist folgende:

> »Athena nach links thronend; Thron mit Sphinx nach links und Löwenfüßen verziert; in der vorgestreckten Rechten Patera haltend, aus der sie eine sich um einen Baum ringelnde Schlange vor ihr füttert, und die Linke am Thronsitz lehnend; hinter ihr frontaler Schild, darauf Eule nach links, Kopf von vorn.«[77]

Zu erkennende Relationen ist (Bemerkung: »Thronsitz« ist kein Element der OBJECTs, weshalb die Relation mit »Athena« und »lehnen« nicht annotiert wurde):

[77] DesignID = 173

[(Athena, PERSON, halten, Patera, OBJECT), (Athena, PERSON, füttern, Schlange, ANIMAL)]

Das Modell sagt keine der beiden Relationen vorher. Betrachtet man den relevanten ersten Teil des Abhängigkeitsbaums der Ikonographie, lässt sich erkennen, dass zwischen »Athena« und »Patera« mit »halten« als Bindeglied keine Verbindung geschaffen und gefunden wird.

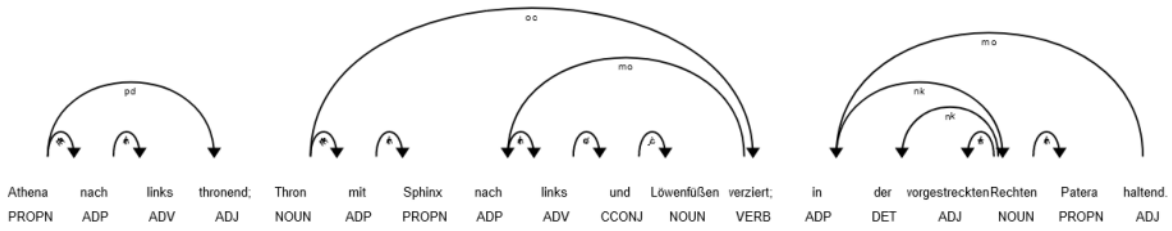

Abbildung 30: Abhängigkeitsbaum bei Ikonographie mit Semikolons

Es wird vermutet, dass dies an der Nutzung von Semikolons liegt. Dies ist auch daran zu beobachten, dass an den Stellen des Auftretens der Semikolons isolierte Segmente geschaffen werden. Das erste Segment endend mit »[…] thronend;[…]« und das zweite Segment endet bei »[…] verziert; […]«. Entfernt man die Semikolons und ersetzt sie durch Kommata, so ist zu erkennen, dass die Relation zwischen »Athena« und »Patera« nun auf Basis der bestehenden Abhängigkeit erkannt wird.

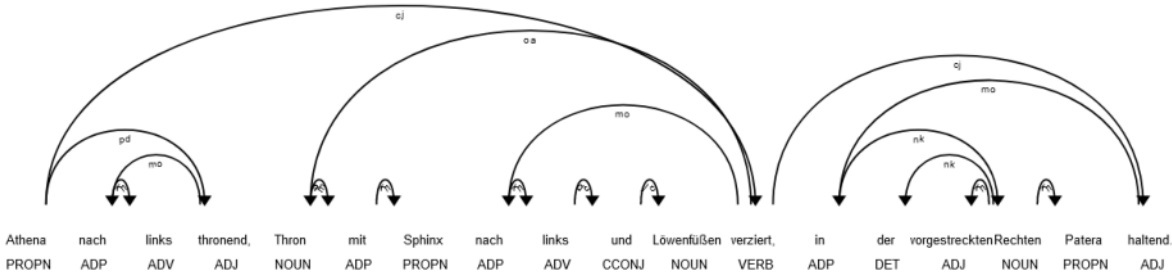

Abbildung 31: Abhängigkeitsbaum nach Ersetzen von Semikolons mit Kommata

Auf diese Art und Weise gehen durch das Vorhandensein eines Semikolons Relationen in insgesamt vier Ikonographen der Strichprobe verloren. Auf Nachfrage bei Frau Ulrike

Peter[78], deutsche Numismatikerin und Mitleiterin des Projektes CNO, besteht die Grundidee des Verwendens von Semikolons darin, die zuerst genannte Hauptfigur einer Münze, vom Rest, dem Nebenbild, abzuheben bzw. abzutrennen. Dass in diesem Fall die Relation zur »Patera« nicht gefunden wird ist ein ungünstiger Fall und einer der eher selteneren »verschachtelten Beschreibungen«. Eine Trennung mit Komma stattdessen empfinde sie eher als verwirrend und sieht als möglichen Lösungsweg eventuell ein Anhängen des Einschubes als eigenständigen Satz am Ende der Ikonographie. Mit dieser Änderung ist die Erkennung der »Patera« - Relation möglich (siehe Abbildung 32).

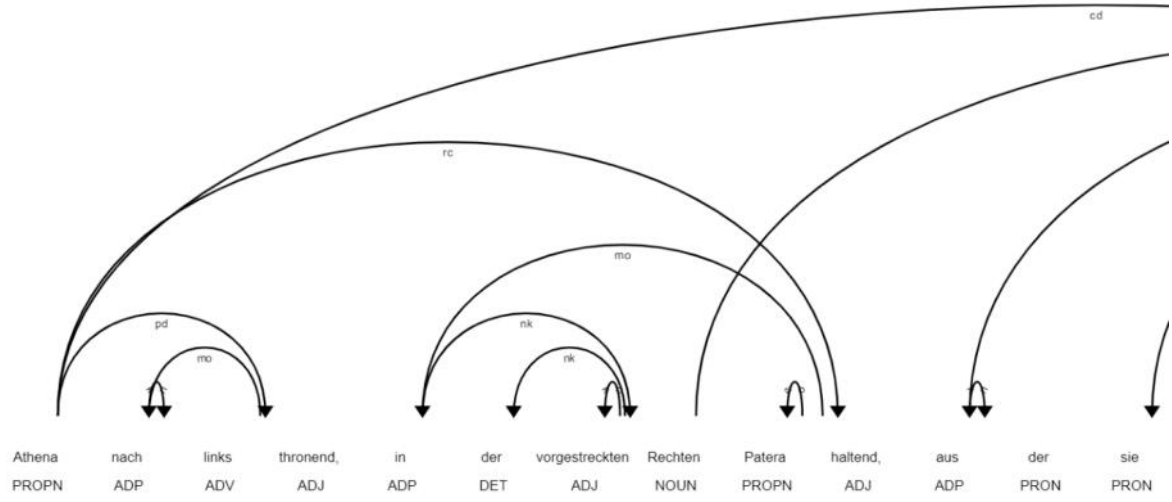

Abbildung 32: Abhängigkeitsbaum nach Verschiebung des Einschubs

Die zweite, nicht vorhergesagte Relation ist die zwischen »Athena« und der »Schlange«, die von ihr »gefüttert« wird. An erster Stelle verhindern auch in diesem Fall die Semikolons das Erkennen einer Abhängigkeit zwischen den Entitäten – doch selbst nach dem Ersetzen der Semikolons wird diese Relation nicht vorhergesagt, obwohl der Pfad zwischen den Entitäten korrekt erkannt wird. Dies ist auf deutsche Sonderfälle bezüglich der Wortstellung zurück zu führen (S-O-P statt S-P-O) welche das Modell nicht korrekt verarbeiten kann. Eine weitere Ikonographie die betrachtet wird ist:

[78] E-Mail-Verkehr vom 08.12.20

»Dionysos auf einem nach rechts laufenden Panther nach rechts sitzend, langen Thyrsos in der Linken haltend und den rechten Arm auf den Panther gestützt.«[79]

Mit den Relationen:

[(Dionysos, PERSON, sitzen, Panther, ANIMAL), (Dionysos, PERSON, halten, Thyrsos, OBJECT), (Dionysos, PERSON, stützen, Panther, ANIMAL)]

Alle Relationen werden korrekt vorhergesagt, bis auf die erste Relation. Das Modell sagt hier die Relationen (Dionysos, PERSON, stützen, Panther, ANIMAL) erneut vorher. In Abbildung 33 befindet sich der Abhängigkeitsbaum dieser Ikonographie.

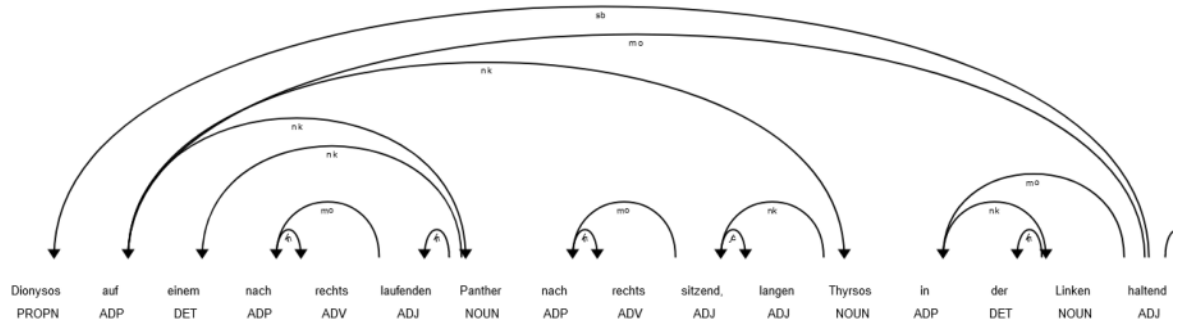

Abbildung 33: Abhängigkeitsbaum mit schwierigem Pfad

Eine Abhängigkeit zwischen »Dionysos« und dem »Panther« wird zwar erkannt, doch betrachtet man die Vorfahren der beiden Entitäten, erhält man folgenden *path*:

[Dionysos, haltend, auf, Panther]

Der *path* wird in diesem Fall, für das Szenario mit »Dionysos« als Subjekt und »Panther« als Objekt, nicht korrekt aufgebaut, sodass das Bindeglied, nämlich »sitzen/d«, nicht mit aufgenommen wird, da es in keinem der Vorfahren-Pfade der beiden Entitäten durch eine Abhängigkeit vorkommt. Dies ist eines der Beispiele bei dem der *DependencyParser* versagt (natürlich in Anbetracht der Tatsache, dass auf beschreibende, an Form Kriterien

[79] DesignID = 343

gehaltene Ikonographen angewendet wird). Dass jedoch dennoch (Dionysos, PERSON, stützen, Panther, ANIMAL) vorhergesagt wird, ist darauf zurückzuführen, dass das Modell bereits die Relation (Dionysos, PERSON, stützen, Panther, ANIMAL) kennt (wie auch im zweiten Teil der Ikonographie selbst) und die *paths* in dieser Art von Relation die Gemeinsamkeit haben, die Worte »Dionysos«, »Panther«, als auch »auf«, zu beinhalten. Besonders »auf« ist in der Kombination mit »stützen« üblich und verbreitet (»sich auf […] stützend.«).

Beispiel zu II Eine weitere Ikonographie für die nicht alle Relationen vorhergesagt werden ist folgende:

> »Kybele mit Mauerkrone nach links auf einem nach rechts springenden Löwen sitzend, den rechten Arm auf dem Tympanon, im linken Arm Zepter haltend.«[80]

Hierbei wird von den Relationen

> [(Kybele, PERSON, tragen, Mauerkrone, OBJECT), (Kybele, PERSON, sitzen, Löwe, ANIMAL), (Kybele, PERSON, stützen, Tympanon, OBJECT), (Kybele, PERSON, halten, Zepter, OBJECT)]

alle, außer der vom auf dem »Löwen« »sitzenden« »Kybele« erkannt. Eine Abhängigkeit zwischen beiden Entitäten konnte hergestellt und erkannt werden, dennoch wurde sie nicht vorhergesagt. Relationen der Form (NE_1, »sitzen«, »Löwe«) mit $NE_1 \in$ Entität kommen im Trainings- und Testsatz nur insgesamt drei Mal vor, wodurch auf diese Relationen nur schwach trainiert werden kann.

Beispiel zu III Darüber hinaus wurden beim manuellen Überprüfen und Analysieren dieser Strichprobe sechs menschliche Fehler im *Ground Truth* erkannt und behoben. Insgesamt sind es 126 annotierte Relationen auf 50 Ikonographen. Das heißt bei knapp 5% dieser

[80] DesignID = 333

annotierten Relationen gab es Fehler von menschlicher Seite. Diese waren simple Fehler, wie doppelt annotierte Relationen oder übersehene nicht-annotierte Relationen.

Die gesamte Strichprobe wird im Anhang dieser Arbeit als »Strichprobe_deutsch.xlsx« angehängt.

5.4 (NE, Verb) - Erweiterung

Beim Evaluieren des englischen Modells wurde die Erkenntnis gemacht, dass viele Ikonographen über Subjekt-Verb-Relationen verfügen, ohne dass dazu ein Objekt existiert. Dies hatte zur Folge, dass diese »halben« Relationen nicht beachtet und annotiert wurden, sodass einiges an Informationen der Ikonographen verloren ging. Dasselbe Problem ist auch in den deutschen Ikonographen anzutreffen.

»Adler nach links fliegend, im Linienquadrat; im quadratum incusum.«[81]

Der »Adler fliegend« wäre in diesem Fall die Information, die das RE des behandelten Modells weder erkennen, noch behandeln würde. Um diese Paare auch abfangen zu können, wird in diesem Kapitel die (NE, Verb) – Erweiterung, welches ein eigenständiges Modell ist, aus **Kapitel 4.6** für den deutschen Datensatz implementiert, angewandt und evaluiert.

5.4.1 Die Implementierung

Zum Implementieren gilt es das Modell, das im Unterschied zum RE zuvor nicht etwa Relationen im Tripel der Form (NE_1, α, NE_2) erkennt, sondern stattdessen sich auf eine einzige Entität plus Verb fokussiert. Betrachtet und erkannt werden sollen alle NE_1 aus [PERSON, OBJECT, ANIMAL, PLANT] und Verben aus den möglichen Relationsklassen in der Form (NE_1, Verb). Genauer arbeitet man mit Tripel der Form (NE, α, β), mit NE aus {PERSON, OBJECT, ANIMAL, PLANT}, α aus den zu klassifizierenden Verben und β aus den Klassifikationsklassen (welche auch Verben sind).

[81] DesignID = 5943

Als *Preprocessing* Schritt muss auch für dieses Modell eine *Ground Truth* Notation gemacht werden. Annotiert werden diesmal alle Tripel der Form (NE, α, β). In der Praxis würde für folgende Ikonographie die Annotation wie folgt aussehen:

»Pferd nach rechts springend.«[82]

mit der *Ground Truth* Annotation:

[(Pferd, springend, springen)]

Das Erstellen der *Ground Truth* kostete 24 Stunden Arbeitsaufwand. Es wurden 1000 deutsche Ikonographen hierfür annotiert. Auch die (NE, Verb) - Erweiterung betrachtet das Anliegen als ein Multiklassenproblem. Auch für dieses Modell ist es also von Nöten, mit den festgestellten Tripel eine erneute Klassifikation zu machen.

Klasse	**Semantisch Äquivalent**
halten	halten, spannen, ausgießen, herführen, entfernen, spielen, drücken, stemmen, hängen, schwingen, ausholen, schleudern, erheben, ziehen
tragen	tragen, schultern
stützen	stützen, lehnen, lagern
sitzen	sitzen, thronen, reiten, galoppieren
bekränzen	bekränzen
stehen	stehen, treten, fahren
winden	winden, ringeln
füttern	füttern, säugen
packen	packen, würgen, fassen
empfangen	empfangen
fliegen	fliegen
schreiten	schreiten, laufen

[82] DesignID = 741

springen	springen
strecken	stecken, vorstrecken, hervorstrecken
drehen	drehen
umschlingen	umschlingen
befreien	befreien
schwimmen	schwimmen
knien	knien, ducken
liegen	liegen
brechen	brechen
drücken	drücken

Tabelle 13: Klassifikation der deutschen (NE, Verb) – Erweiterung

Die Relationsklassen werden, soweit wie möglich, äquivalent zum vorherigen Modell gebaut. Einige bekannte Klassen verfügen hier über mehr semantische Äquivalente, da durch dieses Modell neue Verben, die zuvor nicht beachtet wurden, hinzukommen. Wie bereits diskutiert, kommt das Wort »tragen/d« in den deutschen CNO Ikonographen kaum vor. Da dieses Mal die für »tragen« stellvertretenden Präpositionen nicht als semantische Äquivalente angesehen werden, weil in diesem Modell nur »echte« Verben betrachtet werden, fällt »tragen« deutlich kürzer aus. Neue Relationsklassen, die erst durch dieses Modell erkennbar wurden sind die Klassen »schreiten«, »springen«, »strecken«, »drehen«, »schwimmen« und »knien«. Besonders »springen« und »schreiten«, aus dem englischen »*advancing*«, kommen besonders häufig vor und sind somit die »effektivsten« neuen Klassen für diese (NE, Verb) - Erweiterung. Die Relationsklasse »stehen« ist zwar keine neue Klasse, sie wird in diesem Modell jedoch viel häufiger erkannt. Während es im klassischen Modell 26 Relationen gibt, die als Bindeglied α = »stehen« besitzen (im Tripel (NE_1, α, NE_2)), gibt es im diesem Modell 572 Relationen mit β = »stehen« (im Tripel (NE, α, β)).

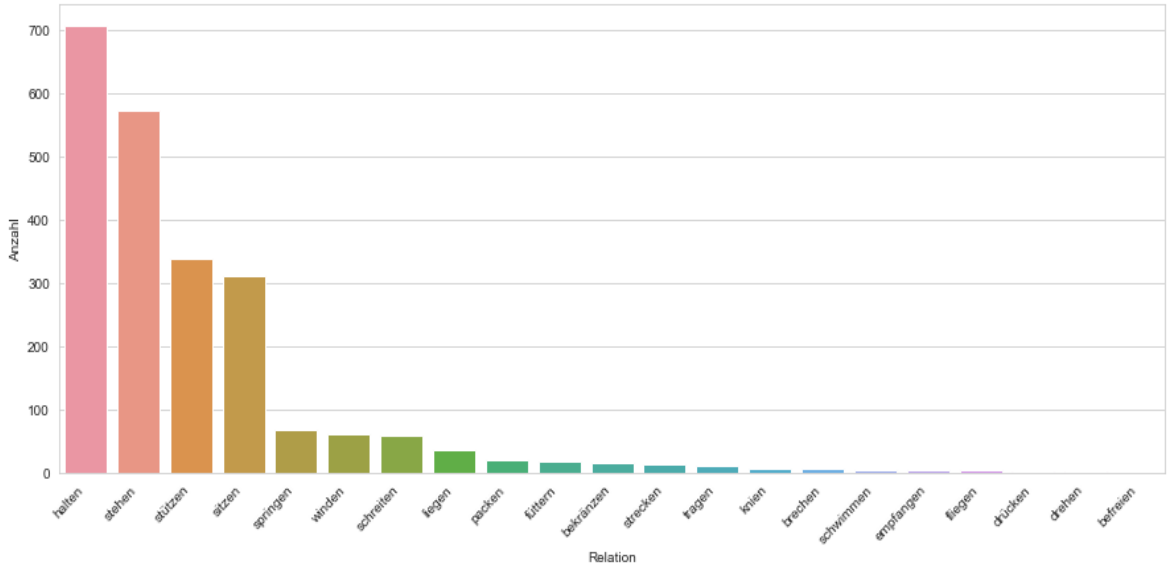

Abbildung 34: Verteilung der Klassen im genutzten Datensatz der deutschen (NE, Verb) – Erweiterung

5.4.2 Evaluation

Diese (NE, Verb) - Erweiterung erzielt nach Wahl der besten Kombinationen von Klassifikator und *Feature* ein F-Maß von **85,59%**. Um erneut die beste Klassifikator und *Feature* Kombination zu ermitteln, wird ein *Gridsearch* durchgeführt. Dieser dauerte fünf Stunden. Als beste Kombination bewährt sich der *Random Forest Classifier* mit dem *Feature Path2Str* mit ent- Tags und einem ngram(1,1). Diese Kombination erreicht neben dem F-Maß von 85,59%, eine *Precision* von 93,32% und einen *Recall* von 79,04%. Auffällig ist, dass alle Kombinationen als *Feature Path2Str* mit dem neu implementierten ent- Tags nutzen. Dies ist naheliegend, da wie schon in **Kapitel 5.3.1** erwähnt, das Erkennen und Arbeiten mit den Verben der Ikonographen für dieses Modell essentiell ist. Da der *DependencyParser* die Verben häufig, durch das in **Kapitel 5.2.1** erläuterte »Partizip- Problem«, als Adjektive markiert, ist das Nutzen der Verben- Tags unerlässlich. Die ent- Tags sind innerhalb des *Path2Str* nötig, sodass die Abhängigkeitspfade korrekt gebaut werden (nämlich mit Partizipien als Verben markiert). Erst an sechster Stelle nutzt man zu den ent- Tags zusätzlich die PoS- Tags, die sich im klassischen Modell bewährt hatten. Hier eben erst an sechster Stelle, da die PoS- Tags bezüglich der Verben Findung wenig beitragen. Als zweites *Feature* wird auch in diesem Modell der *CountVectorizer* benutzt, TF-IDF wird in keiner der Top 20 Kombinationen erwähnt. In der besten Kombination wird ein ngram(1,1) genutzt.

Im Gegensatz zum klassischen Modell, kann festgestellt werden, dass der *Random Forest Classifier* knapp die beste Leistung erreicht. Die Auswahl des Klassifikators hat innerhalb dieses Modells nur eine sehr geringe Auswirkung auf die Klassifikationsergebnisse. Diese liegen unabhängig vom verwendeten Klassifikator auf einem ähnlichen Level.

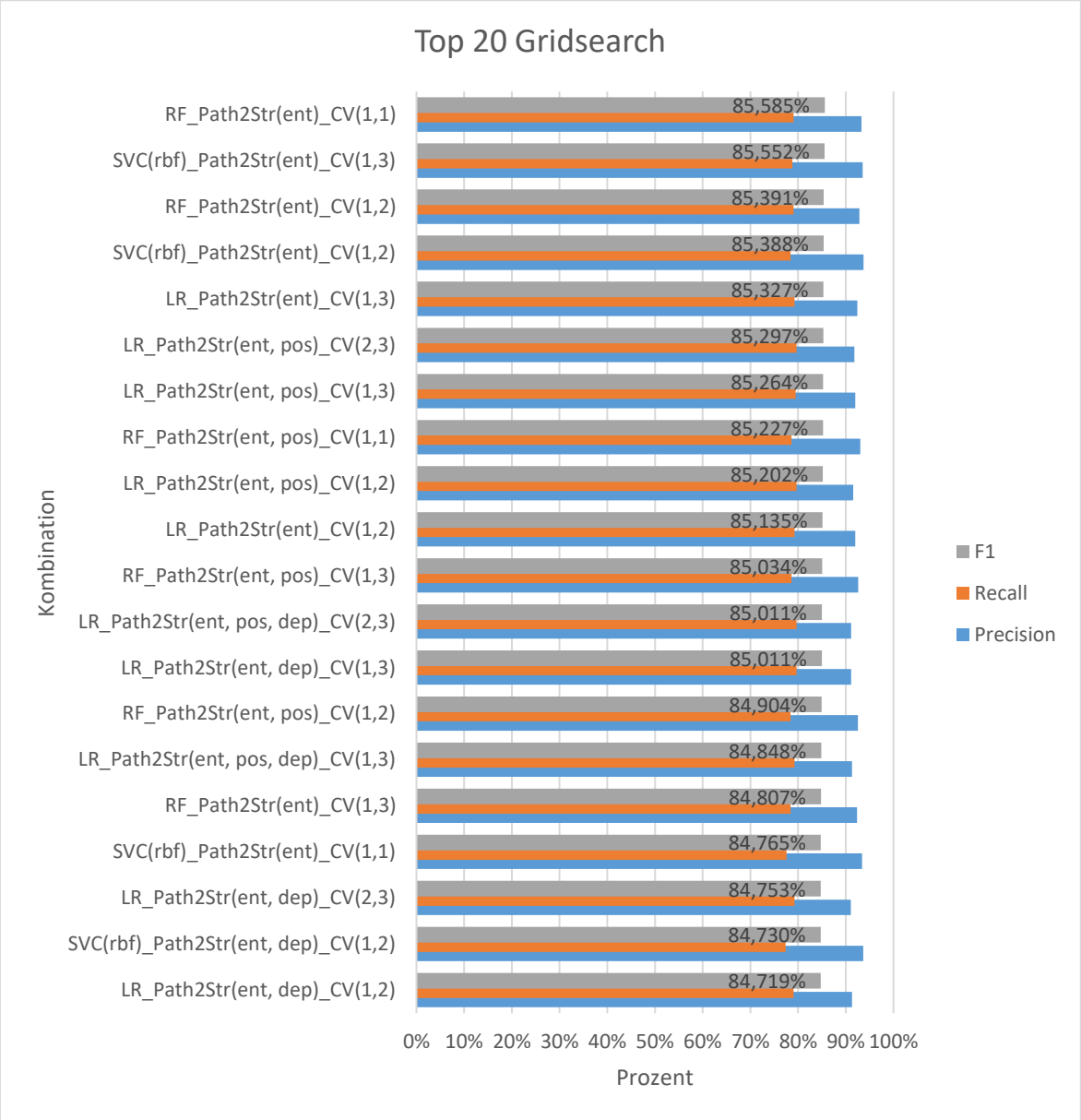

Abbildung 35: Top 20 Kombinationen (F-Maß absteigend, deutsch, (NE, Verb) – Erweiterung)

Die folgende Abbildung zeigt alle getesteten Klassifikatoren und Features. Es ist deutlich zu erkennen, dass pro Klassifikator genau ein großes Feature Cluster entsteht. Das einzige Feature, das deutlich schlechter abschneidet und nicht in die jeweiligen Cluster fällt, ist *AveragedRest2Vec.*

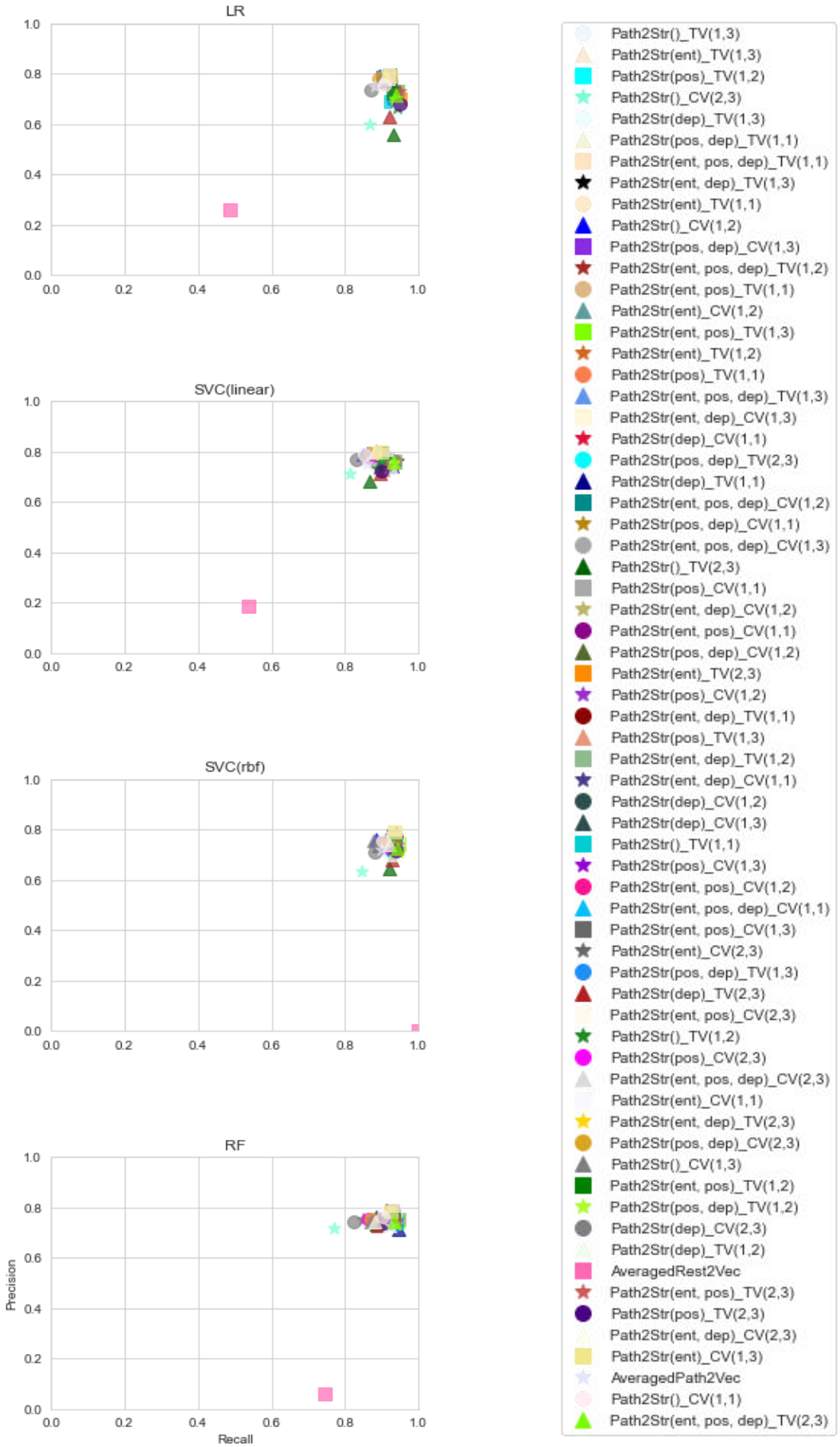

Abbildung 36: Performanceüberblick der verschiedenen Kombinationen (deutsch, (NE, Verb) – Erweiterung)

6. Die Übertragbarkeit

In diesem Kapitel gilt es, die Übertragbarkeit des Modells zu beobachten und analysieren. Dabei wird die Übertragbarkeit aus zwei unterschiedlichen Aspekten betrachtet, zum einen die Übertragbarkeit innerhalb verschiedener Sprachen, zum anderen die Übertragbarkeit des Modells auf fremde Datensätze.

Im ersten Abschnitt 6.1 werden auf Basis der Ergebnisse aus Kapitel 4, dem ursprünglichen englischen Modell und Kapitel 5, dem Modell angewendet auf den deutschen Datensatz, Konklusionen gezogen. Die Herausforderungen und Probleme, die beim Implementieren einer zweiten Sprache aufkamen, werden diskutiert. Der zweite Aspekt beschäftigt sich mit der Übertragbarkeit auf einen anderen Datensatz. Hierbei wird das Modell auf den OCRE (*Online Coins of the Roman Empire*)[83] Datensatz angewendet.

6.1 Übertragbarkeit innerhalb verschiedener Sprachen

Im Folgenden werden die Modelle für die englische als auch deutsche Sprache miteinander verglichen. Betrachtet wird das NER und RE des klassischen Modells in den jeweiligen beiden Sprachen. Die Analyse erfolgt dabei im direkten Vergleich zwischen den beiden umgesetzten Sprachen. Eingeleitet wird mit der Analyse innerhalb der *Named Entity Recognition.* Gefolgt von einer Analyse der *Relation Extraction* und anschließender Betrachtung der (NE, Verb) - Erweiterung.

6.1.1 Named Entity Recognition

Bevor die Leistungen des NERs der Modelle in den jeweiligen Sprachen mit einander verglichen werden, sollte der englisch-deutsch verfügbare CNO- Datensatz analysiert werden. Angefangen wird mit einer Auswertung der gefundenen Entitäten in den jeweiligen Datensätzen.

[83] http://numismatics.org/ocre/ (10.11.20)

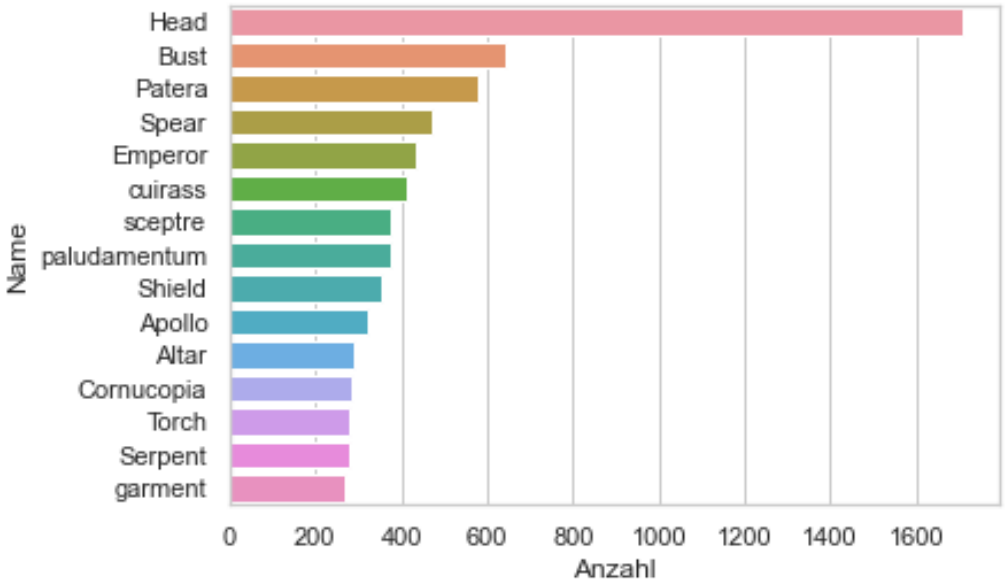

Abbildung 37: Die 15 häufigsten Entitäten im englischen Datensatz

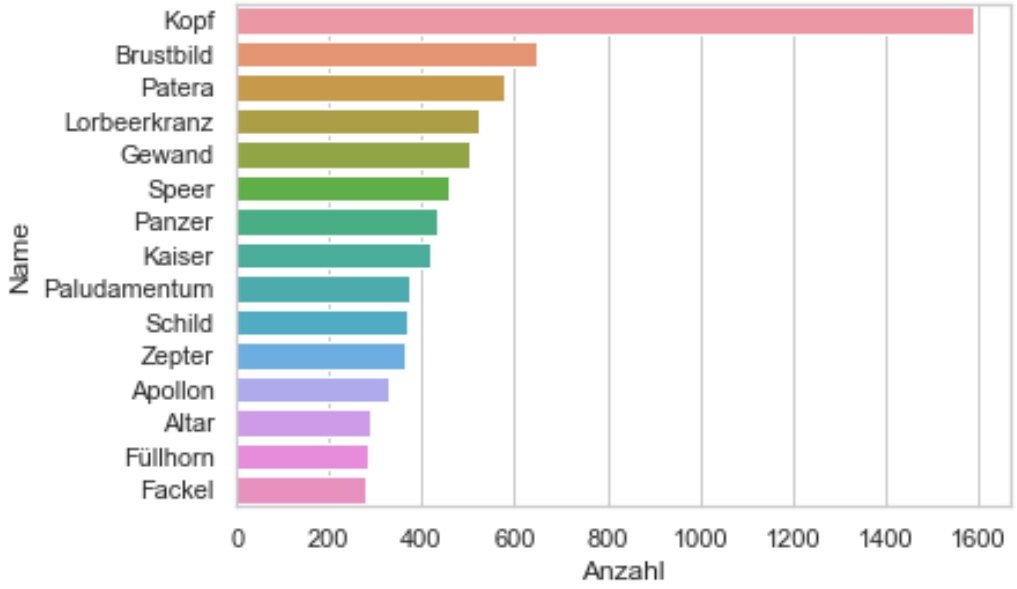

Abbildung 38: Die 15 häufigsten Entitäten im deutschen Datensatz

Betrachtet man im direkten Vergleich das Aufkommen von Entitäten im Englischen und Deutschen, erkennt man wie sich einige der Erkenntnisse aus **Kapitel 5.4.3** darin widerspiegeln. In beiden Datensätzen wird im Modell mit Abstand am häufigsten das Objekt »*head*« bzw. »Kopf« markiert. Addiert man alle gefundenen Variationen, von »*head*« bis »*heads*«, sind dies im Englischen 1713 »*head(s)*«. Im Deutschen kommen wir auf 1572 Mal »Kopf«. Bei beiden Datensätzen dominiert das Wort zwar eindeutig die Liste, doch eine Differenz von annähernd 150 »*head(s)*« muss zunächst erklärt werden. Dies

wurde unter anderem bereits in Kapitel 5.4.3 erkannt und ist zurückzuführen auf den Schritt der Übersetzung. Uneinheitlich erfolgte Übersetzungen, wie beispielsweise »*bare head*« in der Ikonographie:

»Bare head of Antoninus Pius, right, with traces of aegis on left shoulder.«[84]

das unüblicherweise mit »Brustbild von […]« übersetzt wird und zusammengesetzte neue Substantive, möglich Dank der deutschen Sprache, wie schon das bereits diskutierte »Ram's head« zu »Widderkopf«, tragen dazu bei, dass Entitäten, in diesem Fall »*head*«, verloren gehen. Gefolgt wird die Liste mit »*bust*« (641 Mal) bzw. »Brustbild« (645 Mal) und »*patera*« (577 Mal) bzw. »Patera« (579 Mal), die im englischen als auch deutschen Datensatz ähnlich in ihrer Anzahl sind.

Interessant sind auch die Unterschiede bei den am viert- und fünfthäufigsten vorkommenden Entitäten im deutschen Datensatz. Denn an diesen Stellen befinden sich »Lorbeerkranz« mit einer Häufigkeit von 523 und »Gewand« mit einer Häufigkeit von 504 Vorkommen. Wie bereits im Kapitel zuvor erfasst, kommt dies daher, dass viele *past participle* Worte aus den englischen Ikonographen nicht ins Partizip II im deutschen übersetzt werden, sondern stattdessen umformuliert werden. Zur Wiederholung: in den englischen Ikonographen kamen 430 Mal »*laureate*« und 219 Mal »*draped*« vor. Diese sind nun beim Umformulieren in das Deutsche, den Entitäten der OBJECT in Form von »mit Lorbeerkranz« und »mit Gewand« zu Gute gekommen.

Zu erkennen ist, dass die existierenden und erkannten Entitäten innerhalb der Sprachen variieren können. Dies macht die Erstellung von Entitätstabellen für jede Sprache unerlässlich. Ein reines Übersetzen der Entitätstabellen aus dem Englischen in das Deutsche ist nicht ausreichend. Durch die Wahl der Übersetzung, bzw. Variation der Formulierung entstehen neue Entitäten, die vorher in der originalen Sprache nicht immer abgedeckt sein müssen. Nutze man jetzt den F-Maß als Vergleichsmittel, so ist zu erkennen, dass, das erweiterte englische Modell ein F-Maß von **99,2%** erreicht. Das deutsche Modell erreicht **97,7%**. Daraus lässt sich schlussfolgern, dass ein äquivalent effektives Modell für die deutsche Sprache übertragen und umgesetzt werden konnte.

[84] DesignID = 23

6.1.2 Relation Extraction

Um die Leistung des REs innerhalb zwei unterschiedlicher Sprachen, nämlich englisch und deutsch zu bewerten, lässt man die Modelle auf 100 Ikonographen laufen. Diese Ikonographen entstammen der CNO- Datenbank und liegen in englischer als auch deutscher Sprache vor (Im Anhang als »Strichprobe_Übertragbarkeit.xlsx«).

Die Untersuchungen zeigen, dass durch die Implementierung zweier, sprachunterschiedlicher Modelle, es möglich ist Ikonographen effektiver zu analysieren. Durch den Vergleich beider Strichproben in den jeweiligen Sprachen und durch das Erkennen einer ungleichen Anzahl an gefunden Relationen, können folgende Optimierungen vorgenommen werden, die sich in drei Kategorien unterteilen:

I. **Echte Fehler**
   Es können echte Fehler in Form von bspw. Schreibfehlern gefunden und behoben werden.

II. **Erkennung neuer Entitäten**
   Die Modelle können sich gegenseitig verbessern, indem neue Entitäten gefunden und ergänzt werden können.

III. **Unterschiede in den Beschreibungen der Münzen**
   Durch Ungereimtheiten in den Ikonographen zwischen den jeweiligen Sprachen, ist es möglich diese zu Verbessern. Echte Fehler als auch ungenaue Beschreibungen können so identifiziert werden.

Die am häufigsten vorhergesagte Relation ist »*holding*« bzw. »halten«, welche als erstes verglichen wird. In folgender Ikonographie ist beim Vergleichen der Vorhersagen folgender Unterschied aufgefallen. Während im englischen Modell die Vorhersage wie folgt aussieht:

> [(Poseidon, PERSON, holding, dolphin, ANIMAL), (Poseidon, PERSON, holding, trident, OBJECT)]

Fällt auf, dass im Deutschen der »Dreizack« nicht mehr von »Poseidon« »(ge)halten« wird, sondern er sich drauf »stützt«.

> [(Poseidon, PERSON, halten, Delphin, ANIMAL), (Poseidon, PERSON, stützen, Dreizack, OBJECT)]

Um die Quelle dieses Unterschiedes zu ermitteln, werden die Ikonographen betrachtet.

> »Nude bearded Poseidon standing facing, head left, holding dolphin in right hand and trident in left arm.«[85]

> »Nackter Poseidon stehend von vorn, Kopf nach links, in der vorgestreckten Rechten Delphin, die Linke auf den Dreizack gestützt.«

Dabei fällt auf, dass die Vorhersagen beider Modelle zwar korrekt sind, sich die Ikonographen jedoch untereinander unterscheiden. Durch das Vorhandensein der Ikonographen in englischer als auch deutscher Sprache, gibt es Fälle, wie der oben gezeigte, mit semantischen Unterschieden. Das selbe Phänomen, in dem »*holding(s)*« zu »stützen« umgewandelt werden, welches der **III. Kategorie** zuzuordnen ist, kommt in der betrachteten Stichprobe von 100 Ikonographen in weiteren vier Beschreibungen vor (DesignID = 2523, 1505, 1961, 672, 1561, 1580). Ein weiterer Fall wäre folgende Ikonographie:

> »Nude Heracles standing facing, head left, resting right hand on club, left hand on hip, holding lion skin in left arm.«[86]

> »Nackter Herakles stehend von vorn, Kopf nach links, mit der Rechten sich auf die Keule stützend, die Linke in die Hüfte gestemmt; Löwenfell über dem linken Arm.«

[85] DesignID = 964
[86] DesignID = 676

Hier ist zu beobachten, dass das »*holding lion skin*« im Deutschen zu »Löwenfell über dem linken Arm« wird. Wie bereits bekannt, wurden im deutschen Modell einige Präpositionen zu Relationsklassen klassifiziert. Die »über Schulter (Löwenfell)« Formulierung wird zu »tragen« zugeordnet. Das heißt selbst wenn das Modell diese Klassifizierung korrekt vorhersagen würde, hätte man nun ein »tragen« anstelle des »*holding*«.

Abbildung 39: Die Münzabbildung zur Münze mit der DesignID = 676 [87]

Dadurch, dass diese Ikonographie in zwei Sprachen vorliegt, erkennt man, wenn man die Ikonographie betrachtet, dass das »Löwenfell über dem linken Arm« aus dem deutschen eventuell passender ist als das »*holding lion skin in left arm*« der englischen Ikonographie. Dies tritt in zwei weiteren Beschreibung auf (DesignID = 2421, 681). Auch hier ist es eine Beobachtung der **Kategorie III**. Ein weiteres Beispiel, wo ein »*holding*« in Frage steht, ist diese Ikonographie:

> »Veiled Demeter seated left on basket, wearing stephane and long garment, holding three ears of corn and poppy in lowered right hand and long torch in left.«[88]

[87] https://www.corpus-nummorum.eu/coins?id=7248 (11.11.20)
[88] DesignID = 1797

»Demeter mit Schleier, Ährenkranz und im langen Gewand, nach links auf Korb sitzend, in der gesenkten Rechten drei Ähren mit Mohn, die links an der Fackel.«

Das eindeutige »[…] *holding* […] *long torch in left.*« wird im Deutschen zu »[…], die links an der Fackel.«. Eine weitere Ikonographie ist interessant, da hier gleich zwei Unterschiede auftreten.

»Turreted Cybele seated left, on lion jumping right, holding tympanum in right hand, left resting on long sceptre.«[89]

»Kybele mit Mauerkrone nach links auf einem nach rechts springenden Löwen sitzend, den rechten Arm auf dem Tympanon, im linken Arm Zepter haltend.«

Hier ist erneut zu sehen, dass »*holding tympanum*« zu »den rechten Arm auf dem Tympanon« wird. Darüber hinaus ist noch zu erkennen, dass »*resting on long sceptre*«, also eigentlich ein »stützen«, im Deutschen aber zu »Zepter haltend« wird (**Kategorie III**). Letzterer Unterschied wird auch in einer weiteren Ikonographie festgestellt (DesignID = 359).

Als nächste Relationsklasse wird »*wearing*« bzw. »tragen« betrachtet. Der erste signifikante Unterschied innerhalb der Vorhersagen für diese Relationsklasse, ist auf das mehrfach diskutierte Adjektiv-zu-Objekt-Phänomen. Zum Veranschaulichen:

»Veiled Demeter standing facing, head left, holding two ears of corn and poppy in raised right hand and short lighted torch in left arm. Ground line. Border of dots.«[90]

»Demeter mit Schleier stehend von vorn, Kopf nach links, in der erhobenen Rechten zwei Ähren und Mohnkopf und in der Linken kurze brennende Fackel haltend. Bildleiste. Perlkreis.«

[89] DesignID = 333
[90] DesignID = 2443

Das Adjektiv »*veiled*« wird zum Objekt »Schleier«. Dadurch existiert im Deutschen eine weitere »tragen« Relation. Dies passiert, wie schon in **Kapitel 5.2** erwähnt, bei den Worten »*turreted*«, »*draped*«, »*laureate*« und »*diademed*« vor. In dieser Stichprobe tritt das auf weitere 13 Ikonographen zu (DesignID = 1361, 355, 358, 1961, 1839, 351, 1797, 333, 6725, 339, 1622, 340, 343) (**Kategorie III**). Beim Vergleichen der »tragen« Relationen wurde eine weitere Unterschiedsursache festgestellt. In der Ikonographie:

> »Prow with naval ram in shape of animal's head, left; on top, emperor (Marcus Aurelius) standing left, extending right hand, holding parazonium in left arm.«[91]

> »Prora mit Rammsporn in Tierkopfgestalt nach links, darauf Kaiser (Marc Aurel) mit Lorbeerkranz und Panzer nach links stehend, den rechten Arm erhoben, im linken Arm Parazonium haltend.«

Das deutsche Modell sagt hier die Relationen »tragen Panzer« und »tragen Lorbeerkranz« vorher. Im englischen Pendant hingegen sind die Objekte »Lorbeerkranz« und »Panzer« nicht einmal aufzufinden. Ein weiteres Beispiel, das näher betrachtet werden sollte, ist diese Ikonographie:

> »Head of (youthful) Dionysus, right, wearing taenia and ivy wreath with five leaves and two fruits. Hair dress rolled in the back and with two curls falling on his back.«[92]

> »Kopf des (jugendlichen) Dionysos nach rechts, mit Efeukranz von fünf Blättern und zwei Früchten, Haarband, seitlich eingerollter Frisur, die einen Knoten am Nacken bildet, mit zwei Locken auf dem Rücken.«

Während das englische Modell »*wearing Taenia*« vorhersagt, fällt diese Relation zunächst im Deutschen weg bzw. wird nicht vorhergesagt. Beim genaueren Analysieren, fällt auf, dass das Wort »*taenia*« in dieser Ikonographie nicht zu »Taenie«, sondern »Haarband« übersetzt wurde. Dieses Wort wird nicht in der Entitätstabelle der OBJECT aufgeführt und

[91] DesignID = 978
[92] DesignID = 2131

konnte erst dank dem Vergleich der Vorhersagen erkannt und aufgenommen werden. Dies wäre ein Fall der **II. Kategorie**. Die nächste Relation wäre »*resting_on*« bzw. »stützen«. Durch den Vergleich konnte in diesem Szenario ein echter Fehler in der ursprünglichen englischen Ikonographie gefunden werden.

> »Athena standing facing, head right, wearing a helmet and long garment, holding inverted spear in right hand and left resting onshield.«[93]

> »Athena stehend von vorn, Kopf nach rechts, im langen Gewand und mit Helm, in der Rechten den nach unten gerichteten Speer haltend und in der Linken sich auf einen am Boden abgestellten Schild stützend.«

Beim Betrachten der Vorhersagen auf die oberen Ikonographen, ist aufgefallen, dass im deutschen eine »stützen« Relation vorhergesagt wird, die im englischen fehlt. Beim Überprüfen der englischen Ikonographie fällt dabei auf, dass sich ein Schreibfehler eingeschlichen hat. Die »*resting_on*« Relation mit dem »*shield*« konnte auf Grund des Schreibfehlers »[…] *onshield*.« nicht vorhergesagt werden. Dies wäre ein echter Fehler und somit zuzuordnen in **Kategorie I**. Beim Analysieren der »*holdings*« fiel bereits auf, dass einige »*holding*« im Deutschen zu »stützen« wurden. Andersrum ist dies einige Male auch in die Richtung »*resting_on*« zu »halten« der Fall.

> »Turreted Cybele seated left, on lion jumping right, holding tympanum in right hand, left resting on long sceptre.«[94]

> »Kybele mit Mauerkrone nach links auf einem nach rechts springenden Löwen sitzend, den rechten Arm auf dem Tympanon, im linken Arm Zepter haltend.«

Das ursprüngliche »[…] *resting on long screptre*« wird zu »[…] Zepter haltend.«. Dies passiert bei einer weiteren Ikonographie (DesignID = 359) (**Kategorie III**). Weitere Beispiele,

[93] DesignID = 2532
[94] DesignID = 333

in den ein Verschwinden von »*resting on*« zu beobachten ist, ist unteranderem folgende Ikonographie:

> »Nude Heracles seated left on rock covered with lion skin, resting right arm on club, resting left arm on rock.«[95]

> »Nackter Herakles nach links auf Fels sitzend, auf dem Fels Löwenfell, den rechten Arm auf die Keule gestützt, die Linke auf dem Fels.«

Die eindeutige Relation »[…] *resting left arm on rock*.« geht im Deutschen durch die nicht eindeutige Formulierung »[…] die Linke auf dem Fels.« verloren. Dies passiert ein weiteres Mal (DesignID = 2285) (**Kategorie III**). Abschließend erneut ein Beispiel, bei dem eine neue Entität durch den Vergleich gefunden werden konnte.

> »Athena enthroned left; throne decorated with Sphinx, left, front leg ends in lion's paw; holding in right hand patera from which she feeds a serpent entwined around tree in front of her, leaning left arm on throne back; behind her, owl sitting left on a frontal shield.«[96]

> »Athena nach links thronend; Thron mit Sphinx nach links und Löwenfüßen verziert; in der vorgestreckten Rechten Patera haltend, aus der sie eine sich um einen Baum ringelnde Schlange vor ihr füttert, und die Linke am Thronsitz lehnend; hinter ihr frontaler Schild, darauf Eule nach links, Kopf von vorn.«

Während im englischen Modell [(Athena, resting_on, throne)] vorhergesagt wird, fehlt diese im Deutschen. Es wird festgestellt, dass »Thronsitz« nicht durch die Entitätstabelle der OBJECT abgedeckt wurde (**Kategorie II**). Im Falle der Klasse »*seated_on*« gibt es eine Ikonographie, bei der durch das deutsche Modell eine weitere Relation gefunden werden konnte.

[95] DesignID = 674
[96] DesignID = 173

»Emperor (Severus Alexander) in military attire riding right on horseback, covered with panther fur, wearing cuirass, fluttering paludamentum and boots, holding transverse spear in right hand.«[97]

»Kaiser (Severus Alexander) in Kriegsbekleidung mit Panzer, wehendem Paludamentum und in Stiefeln nach rechts auf einem mit einem Pantherfell bedeckten Pferd reitend, in der Rechten Speer den schräg nach unten gerichteten Speer haltend.«

Somit konnte »*horseback*« gefunden werden und als Alternativname für »*horse*« ergänzt werden, da beim Überprüfen der CNO- Datenbank festgestellt wurde, dass »*riding horseback*« ein üblicher Ausdruck ist, der vorher sonst immer verloren ging (**Kategorie II**).

Insgesamt wurden so auf 100 Ikonographen 34 Optimierungsmöglichkeiten gefunden. Dabei sind 30 der **Kategorie III** zuzusprechen, drei Fälle der **Kategorie II** und ein **Kategorie I** Fall. Das heißt auf 100 Ikonographen sind Unstimmigkeiten bei 34 Beschreibungen zu verzeichnen. Nehme man diesen Maßstab als Grundlage, so könnte man theoretisch annehmen, dass bei ca. 30% der Ikonographen durch vorhandene Zweisprachigkeit eine Optimierung möglich ist.

6.2 Übertragbarkeit auf andere numismatische Datensätze

Dieses Kapitel befasst sich mit der Übertragbarkeit des Modells bzw. der Modelle auf andere, fremde Datensätze. Besonders für das NER ist es wichtig, dass es sich bei dem gewählten Datensatz um einen numismatischen Datensatz handelt. Vorzugsweise einer, der äquivalent zu CNO, Münzen aus dem alten Moesien, Thrakien, Mysien und Troas behandelt. Dies ist eine essentielle Bedingung um überhaupt ein annehmbares Resultat erzielen zu können, da die manuell angefertigten Entitätstabellen, Entitäten besitzen, die aus dem oben genannten historischem Raum entspringen. Aus diesem Grund fiel die Entscheidung dabei auf den OCRE Datensatz, welcher Münzprägungen aus dem römischen

[97] DesignID = 1762

Reich beinhaltet. Die Übertragbarkeit wird nur für das englische Modell getestet, da der zur Verfügung stehende OCRE Datensatz nur in englischer Sprache vorliegt.

Angefangen wird die Analyse mit dem Ausführen des NERs auf den OCRE Datensatz. Dabei wird das *Ground Truth* durch den automatischen Annotierprozess erstellt, hierbei stehen den Entitätstabellen schon einige Entitäten aus OCRE bereit. Zu beachten ist dennoch, dass hier vorweg schon mit Einbußen zu rechnen ist, da OCRE Personen der römischen Mythologie behandelt. Während im CNO- Datensatz bspw. die Rede von der Göttin der Jagd ist, so wird von »Artemis« gesprochen – in der römischen Mythologie wäre dies »Diana«. Ausgeführt wird das NER auf 14.467 einzigartige Ikonographen. Es erzielt dabei eine Leistung von **87,6%** für die *Precision,* **85,1%** *Recall* und **86,3%** für das F-Maß.

		Total(TP+FN)	Hits(TP)	Wrongs(FP)	Hits
0	Person	16255	9103	1191	0.560
1	Object	31862	29858	2592	0.937
2	Animal	2034	1881	2165	0.925
3	Plant	2367	2187	1121	0.924

Tabelle 14: NER Vorhersagen auf den OCRE Datensatz

Diese Leistung wird bereits vollbracht, ohne dass hierfür ein wiederholtes manuelles Optimieren für den OCRE Datensatz durchgeführt wird. Das Aussetzen einer manuellen Optimierung wird als Hauptgrund für die schlechte Klassifizierungswerte angesehen. Für die *Precision* als auch den *Recall* sind die *False Positives* bzw. *False Negatives* von Bedeutung. Da es sich beim OCRE Datensatz um 14.467 Ikonographen mit neuen unbekannten Entitäten handelt, fallen diese FPs und FNs dementsprechend hoch aus. Doch nicht alle unter den FPs sind falsch. Um die tatsächliche Effektivität des Modells, neue Entitäten erkennen zu können, zu demonstrieren, wird das Modell nun auf 100 OCRE Ikonographen angewendet. Dabei werden die FPs manuell betrachtet und ausgewertet (siehe Anhang).

	Entität	Total(TP+FN)	Hits(TP)	Wrongs(FP)	Hits
0	PERSON	98	84	5	0.857
1	OBJECT	169	154	10	0.911
2	ANIMAL	5	3	9	0.600
3	PLANT	10	9	12	0.900

Tabelle 15: NER Vorhersagen auf 100 OCRE Ikonographen

Die Klasse der Personen wird dabei als erstes geprüft. Es wurden insgesamt fünf Vorhersagen getroffen, die vom Modell als FPs gewertet wurden. Beim Betrachten dieser scheinbaren FPs, ist zu erkennen, dass es sich bei vier Vorhersagen um echte Personen handelt. Konkreter wären dies: zweimal »Galba«[98], »Uberitas«[99] und »Manlia Scantilla«[100]. Diese können jetzt manuell in die Entitätstabellen der Datenbank aufgenommen werden. Zuvor müsste nur geprüft werden, ob es sich um Alternativnamen handelt. In diesem Fall wären das sogar drei neue Personen. Als nächstes gilt es die zehn FPs von OBJECT zu betrachten. Dabei sind es diesmal vier FPs, die tatsächliche Objekte repräsentieren. Zum einen hätte man dreimal »labarum« und einmal »camp gate«. Bei den Tieren ist nur eins der neun FPs ein echtes Tier und zwar wurde »crocodile« gefunden. Bei den Pflanzen wurde keine neue Entität gefunden. Zurückzuführen ist diese Beobachtung wohl darauf, dass Tiere und insbesondre Pflanzen generell eher selten auftauchen. Im Falle der Pflanze ist es gleichzeitig auch die geringe vorkommende Diversität, sprich: Hat man eine gewisse Anzahl an Pflanzen durch die Entitätstabelle abgedeckt, so wird es im Rahmen von Ikonographen immer unwahrscheinlicher auf etwas Neues zu stoßen. Darüber hinaus ist aufgefallen, dass hin und wieder zwar neue Entitäten der Pflanzen und Tiere gefunden werden, diese jedoch unter den Vorhersagen der PERSONs auftauchen. Das heißt, auf dieser Strichprobe von 100 OCRE Ikonographen mit 36 *False Positives* haben sich neun als echte Entitäten herausgestellt. Das bedeutet 25% der FPs entsprechen tatsächlichen neuen Entitäten. Zu beachten ist, dass der Wert relativ gering ausfällt, weil die Klassen Tiere und Pflanzen seltener vorkommen, dementsprechend auch eher schlechter erkannt werden und somit den Klassen Personen und Objekte unterliegen. Zusammenfassend kann man sagen, dass

[98] http://nomisma.org/id/galba (12.11.20)
[99] http://numismatics.org/ocre/results?q=deity_facet%3A%22Uberitas%22 (12.11.20)
[100] http://nomisma.org/id/manlia_scantilla (12.11.20)

das Modell auf einem fremden Datensatz zuverlässig aus dem CNO bekannte Entitäten erkennt und darüber hinaus, besonders für die Entitätsklassen PERSON und OBJECT, neue Entitäten erkennt.

Nachdem die Übertragbarkeit des NERs auf den OCRE Datensatz gezeigt wurde, gilt es nun die Übertragbarkeit des REs zu überprüfen. Da für die Ikonographen des OCRE Datensatzes keine *Ground Truth* zu Grunde liegt, wird die Leistung des REs auf eine Stichprobe von 100 Ikonographen manuell ausgewertet. Das RE wurde auf alle 14.467 Ikonographen angewendet, doch nur für 1587 Ikonographen wurden Vorhersagen gemacht. Dies liegt daran, dass das RE vom NER limitiert wird. Das NER findet, wie schon erwähnt, zwar bekannte Entitäten, doch bei neu vorhergesagten Entitäten, steht noch der manuelle Ergänzungsprozess aus. Das heißt die meisten Relationen aus dem OCRE Datensatz können ohne jegliche Anpassungen bzw. Ergänzungen nicht gefunden werden. Aus der Menge der 1587 Ikonographen werden nun 100 zufällige Ikonographen analysiert und bewertet. Die Stichprobe von 100 Ikonographen wurde manuell annotiert und es ergeben sich **220 Relationen**, die gefunden werden können. Das Modell schlägt **175 Relationen** vor, von denen sich nach manuellem auswerten **87 Relationen** als richtig erweisen. Das entspricht **39,5%** der möglichen Relationen. Zwei große Faktoren, die diese Leistung erklären, sind – die bereits erwähnte NER Limitierung (bspw. das Objekt *»(holding) Victory*« oder »*reaping hook*«) und Unterschiede in den Formulierungsrichtlinien. Während das Modell CNO- Richtlinien konform (siehe **Kapitel 4.4**) ist, hat OCRE keine Richtlinien. Der häufigste damit verbundene Fehler ist das Bindestrichproblem. So wurde das CNO Modell bspw. nur mit »*laurel branch*« trainiert, während in der OCRE Strichprobe allein 17 Mal »*laurel-branch*« auftaucht, dass zum aktuellen Zeitpunkt nicht korrekt erkannt bzw. verarbeitet wird.

Daraus lässt sich schlussfolgern, dass die Übertragbarkeit des REs zwar gegeben ist, jedoch ohne jegliche Anpassungen nicht die gewünschten Ergebnisse liefert. Um eine angemessene Leistung zu erbringen, muss ein ausführlicher NER Prozess durchgeführt werden, bei dem die vom neuen Datensatz spezifischen Entitäten ergänzt werden müssen. Weiterhin ist eine Entscheidung darüber nötig, wie die Richtlinien zwischen unterschiedlichen Datensätzen geregelt werden sollen. So könnte man bspw. das Problem mit den Bindestrichen damit lösen, dass man sie nicht im generellen verbietet, sondern als Alternativnamen aufnimmt oder den Datensatz den bestehenden Richtlinien anpasst.

7. Fazit und Ausblick

Das Ziel dieser Masterarbeit war es, das englischsprachige NLP- Modell aus der Bachelorarbeit »*Natural Language Processing to enable semantic search on numismatic descriptions*« von Frau P. Klinger zu erweitern und auf die deutsche Sprache zu übertragen. Die erste Erweiterung des Modells besteht aus Hinzufügen zweier weiterer Entitätstypen – den Tieren (ANIMAL) und Pflanzen (PLANT). Die zweite Erweiterung umfasst ein separates Modell, das sich auf das Erkennen von Entität-Verb-Beziehungen fokussiert. Der zweite Aspekt dieser Arbeit war es, das erweiterte Modell auf eine weitere Sprache zu übertragen und anzuwenden. Abschließend wurde die Übertragbarkeit des Modells auf einen weiteren numismatischen Datensatz überprüft.

Alle folgenden Leistungsauswertungen kommen durch den Vergleich zu den Leistungen des Modells aus der Arbeit von P. Klinger zustande und sind am Ende dieses Abschnittes in Form einer Vergleichstabelle angehängt. Das NER des englischen Modells erkennt nun, zusätzlich zu PERSON und OBJECT, die beiden hinzugefügten Entitätstypen ANIMAL und PLANT (siehe **Tabelle 16**, NER) und das RE die Relationen ausgehend aus den Entitäten {PERSON, OBJECT, ANIMAL} zu {PERSON, OBJECT, ANIMAL, PLANT}. Die Leistung, gemessen an den Metriken *Precision*, *Recall* und F-Maß, entspricht für das NER 99,2% für die *Precision*, 99,3% für den *Recall* und 99,2% für das F-Maß (siehe **Kapitel 4.5.1**). Im RE erreicht das englische Modell 90% *Precision*, 82,2% *Recall* und 86,1% F-Maß (siehe **Kapitel 4.5.2**). Damit wurde gezeigt, dass es möglich ist neue Entitätstypen hinzuzufügen und die Relationen zwischen diesen, als auch den alten Entitätstypen zu erkennen (siehe **Tabelle 16,** RE).

Zusätzlich zu der eigentlichen Aufgabenstellung, wurde ein separates Modell implementiert, das besonderes den Fokus auf das Erkennen von (NE, Verb) – Relationen setzt. Das neue Modell (NE, Verb) – Erweiterung ermöglicht es nun auch Relationen aus Ikonographen, bestehend aus nur Subjekt und Verb, zu erkennen. Dadurch wurde es ermöglicht, zusätzliche semantische Informationen aus den Ikonographen zu extrahieren, bei denen das klassische Modell nicht in der Lage war diese zu verarbeiten. Die Erkennung dieser funktioniert mit einer *Precision* von 91,8%, einem *Recall* von 88,1% und einem F-Maß von 89,9% (siehe **Kapitel 4.6**).

Das Modell wurde auf eine weitere Sprache übertragen. Angewendet wurde es auf die deutschsprachigen Ikonographen des CNO- Datensatzes. Das NER erzielt eine zum englischen annährend äquivalente Leistung mit einer *Precision* von 97,6%, *Recall* von 97,8% und F-Maß von 97,7%. Beim RE kann man von nahezu gleichen Werten sprechen. Die *Precision* erreicht 89,5%, der *Recall* 83,8% und das F-Maß 86,6% (siehe **Kapitel 5.3**). Selbst die (NE, Verb) - Erweiterung weicht der Leistung des englischen Modells kaum ab (*Precision* 93,3%, *Recall* 79%, F-Maß 85,6%). Es wurde gezeigt, dass der NLP- Ansatz in eine weitere Sprache übertragen werden kann. Um diese Werte zu erreichen, ist ein passendes, von *spaCy* angebotenes, Sprachmodell als auch ein manuelles *Preprocessing* unerlässlich (siehe **Kapitel 5.2.2**).

Es wurde die Erkenntnis gewonnen, dass es möglich ist, durch das Vorhandensein zweier Modelle in unterschiedlichen Sprachen, Fehler und Unstimmigkeiten in den Ikonographen ausfindig zu machen und somit die zugrundeliegende Datenqualität zu optimieren. So konnte alleine in einer Stichprobe von 100 Ikonographen, durch den zweisprachigen Vergleich bei circa 30% der Ikonographen eine Optimierung vorgenommen werden (siehe **Kapitel 6.1**).

Bei der Übertragbarkeit auf einen weiteren numismatischen Datensatz lassen sich zwei Erkenntnisse gewinnen. Das NER lässt sich auf den OCRE- Datensatz ohne weitere Anpassungen oder *Preprocessing* anwenden. Dabei wird ein F-Maß von 86,3% erreicht. Das Ergebnis weicht vom Ergebnis auf den CNO- Datensatz ab, da es neue unbekannte Entitäten gibt. Es wurde aber gezeigt, dass das Modell diese neuen Entitäten erkennt und vorschlägt (**Kapitel 6.2**). Damit wurde gezeigt, dass eine direkte Übertragbarkeit des Modells besonders innerhalb desselben Sachgebiets, der altgriechisch-römischen Numismatik, gegeben ist.

Für das RE jedoch, geht hervor, dass es ohne jegliche Anpassungen nicht direkt übertragbar ist (siehe **Kapitel 6.2**). Nur CNO- spezifische Relationen werden gefunden, da die neuen Entitäten und damit verbundenen Relationen aus OCRE ohne ein *Preprocessing* nicht erkennbar sind.

	Modelle dieser Arbeit						Modell der Arbeit		
	English			Deutsch			von P.Klinger		
	Precision	*Recall*	F1	*Precision*	*Recall*	F1	*Precision*	*Recall*	F1
NER	99,0%	99,1%	99,1%	97,6%	97,8%	97,7%	98,0%	97,0%	
RE	90,0%	82,2%	86,1%	89,5%	84,6%	86,9%	92,0%	84,0%	88,0%
NER (OCRE)	87,6%	85,1%	86,3%				99,0%	96,0%	
RE(NE, Verb)	93,7%	82,9%	88,0%	93,3%	79,0%	85,6%			

Tabelle 16: Leistungsvergleich

Ausblick Durch die Erkenntnisse dieser Arbeit, ergeben sich unteranderem folgende Punkte, die über diese Arbeit hinaus behandelt werden könnten:

I. *Ground Truth*- Generierung

Da das Vorbereiten eines solchen Modells mit viel und besonders zeitintensiver manueller Arbeit verbunden ist, wäre es eine Idee, eine automatische Generierung der *Ground Truth* zu realisieren. Beim NER Prozess wäre es die automatische Auswertung der neu gefundenen Entitäten, die zurzeit manuell ausgewertet werden, sei es durch automatisierte Überprüfungen in Wörterbüchern o.ä. Um das RE umzusetzen ist eine manuelle Annotation des Datensatzes nötig. Der Trainingssatz könnte durch eine Generierung von passenden randomisierten Ikonographen beschleunigt bzw. weiterhin verbessert werden. Andernfalls wäre es auch möglich, Vorhersagen des Modells durch eine extra angefertigte UI in kürzester Zeit anpassen, ergänzen und der *Ground Truth* hinzufügen zu können.

II. Mehrsprachiges Modell

Es könnte von Interesse sein, ein einziges Modell zu implementieren, welches die Anwendbarkeit auf mehrere Sprachen ermöglicht.

III. Verbesserte Suche durch Einbeziehen zusätzlicher Sprache

Durch eine Suche auf einen mehrsprachigen Datensatz, sollte es möglich sein, mehr Informationen bzw. Münzen ermitteln zu können, da Unterschiede in den bereitgestellten Ikonographen durch ihr sprachliches Pendant wettgemacht werden können.

Danksagung

Wir möchten uns bei Prof. Dott. -Ing. Roberto V. Zicari für das Betreuen unserer Masterarbeit bedanken. Einen besonderen Dank möchten wir an Dr. Karsten Tolle aussprechen, der uns als unser direkter Betreuer stets zur Seite stand und uns bei jeglichen Fragen behilflich war. Außerdem möchten wir Sebastian Gampe danken, der uns bei Fragen zur Münzdatenbank helfen konnte. Unser Dank gilt auch Frau Ulrike Peter. Vielen Dank für das nette Zusammenarbeiten, trotz den aktuellen Umständen, die uns das gemeinsame Arbeiten erschwert haben.

Anhang

Im Anhang befindet sich der in der Masterarbeit diskutierte Code. Zum Ausführen bitte die angehängten und dokumentierten Jupyter Notebooks verwenden. Auch ein Dump der Datenbank und die erstellten *Ground Truth* Daten sind angehängt. Außerdem befinden sich im Anhang alle Stichproben, die in dieser Arbeit analysiert wurden.

Literaturverzeichnis

Bird, Steven und Ewan Klein, Edward Loper (2009). *Natural Language Processing with Python.* ISBN-13: 978-0-596-516499

Bishop, M. Christopher (2006). *Pattern Recognition and Machine Learning.* ISBN-13: 978-0387-31073-2

Brownlee, Jason (2016). *Master Machine Learning Algorithms: Discover How They Work and Implement Them From Scratch.*

Göbl, Robert (1987). *Numismatik: Grundriss und wissenschaftliches System.* ISBN-13: 978-3894412319

Haymann, Florian (2016). *Antike Münzen sammeln: Einführung in die griechische und römische Numismatik, Exkurse zu Kelten und Byzantinern.* ISBN-13: 978-3866461321

Hänsch, Ronny und Olaf Hellwich (2015). *Random Forests.* ISBN: 978-3-662-46900-2

Imo, Wolfgang (2016), *Grammatik: Eine Einführung.* ISBN 13: 978-3-476-05431-9

Klinger, Patricia (2018). *Natural Language Processing to enable semantic search on numismatic descriptions.*

Kroha, Tyll (1986). *Münzen Sammeln: ein Handbuch für Sammler und Liebhaber.* ISBN: 9783781402492

Lane Hobson, Cole Howard und Hannes Max Hapke (2019). *Natural Language Processing in Action.* ISBN: 9781617294631

Marquez, Lluis und Xavier Carreras, Kenneth C.Litkowski und Suzanne Stevenson (2008). *Semantic Role Labeling: An Introduction to the Special Issue.*

Sun, Aixin und Ee Peng Lim, Ying Liu *(2009). On strategies for imbalanced text classification using SVM: A comparative study.*

Twain, Mark (1880), *The Awful German Language.*

Wagner, Birgit (2009), *Pons Praxis-Grammatik Englisch.* ISBN: 978-3-12-562792-5

Wilhelm Schmidt (2013), *Geschichte der deutschen Sprache. Ein Lehrbuch für das germanistische Studium.* ISBN: 978-3-7776-2272-9

Vasiliev, Yuli (2020). *Natural Language Processing with Python and Spacy.* ISBN-13: 978-1-7185-00525

Abbildungsverzeichnis

Tabellenverzeichnis