

Named Entity Classification

Madita Huvar, Sanaz Safdel, Phillip R.-P.

January 23, 2017

- 1 Einführung
- 2 Daten & Tools
 - Tools
 - Korpus
 - Korpusklassen
- 3 Klassifizierer
 - Features für den Baseline-Klassifizierer
 - Erweitertes Featureset
 - Klassifizierertyp
 - Erfahrungen mit den Korpusklassen
- 4 Evaluation
 - Probleme
- 5 Ausblick
- 6 Referenzen

Named Entity Recognition seit 1990er Jahren aktives Forschungsfeld.
(Überblick: Borthwick, 1999, Tjong Kim Sang 2003, Marrero 2013)

Grundlage für weitere Forschungsfelder im Bereich Information Retrieval,
z.B. Semantic Annotation, Question Answering, Opinion Mining, usw.
(Marrero 2013)

Was sind Named Entities

Named Entities sind Phrasen die Namen von Personen, Organisationen, Währungen, usw enthalten:

Beispiele für Named Entities

[ORG U.N.], [PER Obama], [MONEY Dollar], [LOC Moscow]

- Typischerweise werden Named Entity Recognition und Named Entity Classification (NEC) zusammen betrachtet.

- Typischerweise werden Named Entity Recognition und Named Entity Classification (NEC) zusammen betrachtet.
- Wenige Untersuchungen beschäftigen sich nur mit NEC.
(Primadhanty 2014, He 2016, Spangler 2016)

- Typischerweise werden Named Entity Recognition und Named Entity Classification (NEC) zusammen betrachtet.
- Wenige Untersuchungen beschäftigen sich nur mit NEC. (Primadhanty 2014, He 2016, Spangler 2016)
- Dieses Projekt konzentriert sich auf NEC und stellt die Frage, **welchen Einfluss Feature Selection auf die Klassifikationsergebnisse eines Named Entity Klassifizierers hat.**

- Typischerweise werden Named Entity Recognition und Named Entity Classification (NEC) zusammen betrachtet.
- Wenige Untersuchungen beschäftigen sich nur mit NEC. (Primadhanty 2014, He 2016, Spangler 2016)
- Dieses Projekt konzentriert sich auf NEC und stellt die Frage, **welchen Einfluss Feature Selection auf die Klassifikationsergebnisse eines Named Entity Klassifizierers hat.**
- Nutzung einfacher syntaktischer und lexikalischer Features, die in fast allen Forschungsarbeiten in ähnlicher Form genutzt wurden. (*Toral, Munoz, 2006; Kazama, Torisawa, 2007; Ratinov, Roth 2009*)

- Python 3.4+
- Scikit Learn als Klassifizierer
- liac-arff
- matplotlib
- Weka zur Korpusanalyse

- Für Named Entity Klassifikation wird OntoNotes Korpus 2012 genutzt. (*OntoNotes Release 5.0 2012*)
- Englischen Nachrichtentexte des 'The Wall Street Journal'. Für die Entwicklungsphase bereits vorgefertigtes Developmenttest.
- Für die Klassifikation der Named Entities werden die bereits vorgefertigten Trainings- und Testdatensets genutzt.

Table: Anzahl an atomaren Named Entities

Developmentset	Trainingset	Testset
3325	23686	2996

- Für Extraktion der Named Entities wurde ein Korpusreader erstellt.
- Der Reader extrahiert alle Named Entities, inklusive POS-Tags der einzelnen Token, Phrasenart, Kontextwörtern (ne-1, ne+1), und ordnet ihnen Klassen zu.

Extrahierte Instanz der Korpusreader-Klasse

```
{'PERSON':[['Peter', 'NNP'],['Mokaba', 'NNP'],'NP', ('Says', ',')]}
```

Table: Klassen im OntoNotes Korpus (*OntoNotes Release 5.0 2012*)

Klassen	Trainingset
ORG	5788
DATE	4080
PERSON	3756
GPE	3601
CARDINAL	1852
MONEY	1509
NORP	1484
PERCENT	1061
FAC, LOC, PRODUCT, EVENT, WORK_OF_ART, LAW, LANGUAGE, TIME, QUANTITY, ORDINAL	< 1800

- Zehn Klassen enthalten nur wenige NE-Instanzen. Diese werden aus dem balancierten Korpus entfernt.
- Semantisch ähnliche Klassen NORP und GPE werden zusammengefasst.
- Numerische Klassen MONEY, PERCENT und CARDINAL werden ebenfalls zusammengefasst.

Table: Balancierte Klassen

Klassen	Beschreibung
PERSON	People, including fictional
NORP_GPE	Nationalities or religious or political groups Countries, cities, states
ORGANIZATION	Companies, agencies, institutions, etc.
DATE	Absolute or relative dates or periods
PERCENT_MONEY_CARDINAL	Percentage (including “%”) Monetary values, including unit Numerals that do not fall under another type

Table: Verteilung der Klassen nach Balancierung

Klassen	Developmentset	Trainingset	Testset
ORG	930	5857	859
GPE_NORP	732	5134	588
PERCENT_CARDINAL_MONEY	564	4672	529
DATE	613	4254	601
PERSON	486	3759	413

Beispielinstanz zur Veranschaulichung der Features

```
[ ['North', 'NNP'], ['-', HYPH], ['America', 'NNP'], 'NP', (',', 'and')]
```


Anzahl der Features: 1317

Table: Features für den Baseline-Klassifizierer

Feature	Wert	Beschreibung
Unigram	numerisch	Vorkommen der Unigramme, die mindestens fünfmal im Trainingscorpus vorkommen. (<i>Mayfield 2003</i>)

Erweitertes Featureset I

Anzahl der Features: 1716

Table: Features für den Klassifizierer I

Feature	Wert	Beschreibung
Unigram	numerisch	Vorkommen der Unigramme (lemmatisiert), die mindestens fünfmal im Trainingscorpus vorkommen. (<i>Mayfield 2003</i>) (america : 1, north : 1)
POS	numerisch	Häufigkeit von 36 POS-Tags aus der Penn Treebank (<i>Florian, Chieu 2003</i>) NNP: '2'
isAllCaps	boolean	Wörter nur in Großschreibung (<i>Nadeau 2006</i>) (0)
Context	numerisch	Vorkommen der Kontexttokens. Beinhaltet Vorgänger- und Nachfolgetoken der NE. (<i>Munro 2003</i>) (, and : 1)
containsDigit	boolean	Vorkommen von Nummern. (0)

Table: Features für den Klassifizierer II

Feature	Wert	Beschreibung (Beispielwert)
isInWiki	boolean	Vorkommen der NE in der Wikipedia. (<i>Toral, Munoz 2006</i>) (1)
isTitle	boolean	Prüft, ob Titelbezeichnungen (z.B. Mr. MA) vorkommen. (<i>Ratinov, Roth 2009</i>) (0)
isNP	boolean	Ist NE eine Nominalphrase. (<i>Sánchez, Cuadrado 2009</i>) (1)
isName	boolean	Prüft, ob Vornamen vorkommen. (<i>Ratinov, Roth 2009</i>) (0)
containsDash	boolean	Vorkommen von Viertelgeviertstrichen. (<i>Mayfield 2003</i>) (1)
isComName	boolean	Prüft aus kommerzielle Bezeichner (0)

- Zur Klassifizierung der NE wird eine Support Vector Maschine mit linearem Kernel verwendet.

- Zur Klassifizierung der NE wird eine Support Vector Maschine mit linearem Kernel verwendet.
- SVM (`sklearn.svm.LinearSVC`)
Featurevektoren haben sehr viele Features daher linearer Kernel.
Mapping in höheren Featurespace eines nicht-linearen Kernels bringt kaum Klassifizierungsverbesserungen. (*Chih-Wei Hsu 2003*)

- Zur Klassifizierung der NE wird eine Support Vector Maschine mit linearem Kernel verwendet.
- SVM (sklearn.svm.LinearSVC)
Featurevektoren haben sehr viele Features daher linearer Kernel.
Mapping in höheren Featurespace eines nicht-linearen Kernels bringt kaum Klassifizierungsverbesserungen. (*Chih-Wei Hsu 2003*)
- *Alternativ wurde ein Decisiontree getestet, dieser hatte allerdings mit allen Featurekombinationen tendenziell schlechtere Evaluationsergebnisse. Zudem trainiert der SVM deutlich schneller.*

Wie die ROC-Kurve zeigt, hat der Klassifizierer insbesondere Schwierigkeiten, die Klassen PERSON und ORG und GPE_NORP zu unterscheiden.

ROC Curve

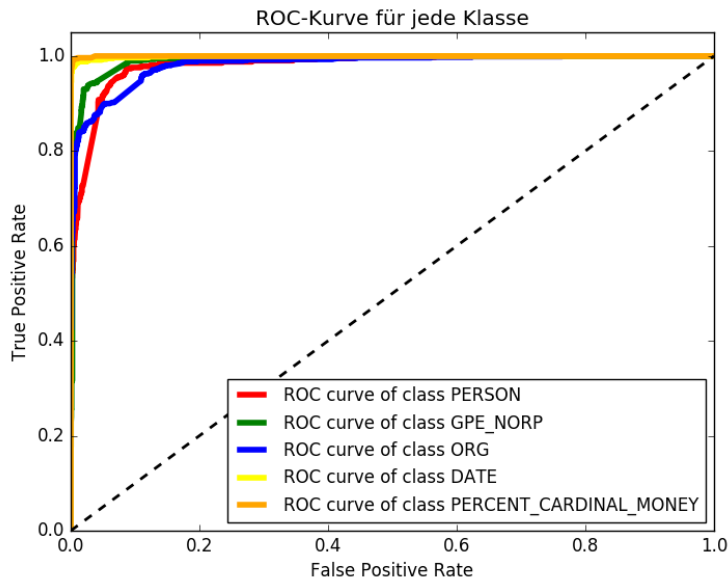


Table: Confusion Matrix

361	28	22	2	0	PERSON
29	549	10	0	0	GPE_NORP
71	45	736	6	1	ORG
0	2	1	591	7	DATE
0	2	0	2	525	PERCENT_CARDINAL_MONEY

- Von 361 PERSON Entities, werden 71 als ORG und 29 als GPE_NORP klassifiziert.
- Numerische Klassen und Datum werden fast zu 100% erkannt.

- Insgesamt wurden elf Features eingesetzt.

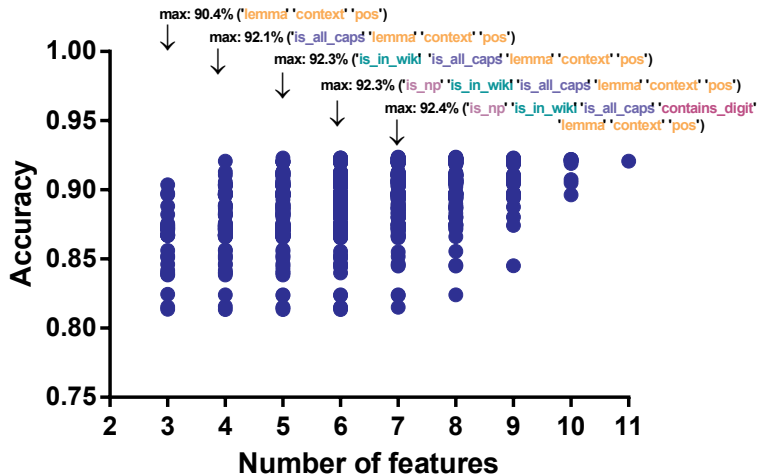
- Insgesamt wurden elf Features eingesetzt.
- Um die Performance der einzelnen Features zu testen, wurde die Potenzmenge des Featuresets gebildet.

- Insgesamt wurden elf Features eingesetzt.
- Um die Performance der einzelnen Features zu testen, wurde die Potenzmenge des Featuresets gebildet.
- Schließlich wurde der Klassifizierer auf allen 1013 Teilmengen durchgeführt.

Für die Evaluation entscheidend waren alle Teilmengen, die die Features 'Unigram' und 'Context' enthalten und mind. drei Features 3 besitzen.

- Accuracy aller Teilmengen ohne diese Features: <69 %.
- Accuracy nur mit Unigram und Context: 87.42%

Featureselektion III



- Beste Features: 'Unigram', 'Context'

- Beste Features: 'Unigram', 'Context'
- Höchste Accuracy: Ab sieben Features. Ab vier Features kaum mehr Verbesserung der Accuracy

- Beste Features: 'Unigram', 'Context'
- Höchste Accuracy: Ab sieben Features. Ab vier Features kaum mehr Verbesserung der Accuracy
- Features, die zur Erhöhung der Accuracy beitragen: 'POS', 'is_all_caps', 'is_in_wiki', 'is_np', 'contains_digit'

- Beste Features: 'Unigram', 'Context'
- Höchste Accuracy: Ab sieben Features. Ab vier Features kaum mehr Verbesserung der Accuracy
- Features, die zur Erhöhung der Accuracy beitragen: 'POS', 'is_all_caps', 'is_in_wiki', 'is_np', 'contains_digit'
- Das Featureset aus vier Features: 'Unigram', 'Context', 'POS', 'is_all_caps' erreicht die beste Accuracy bei möglichst kleinem Featureset.

Evaluationsergebnisse der Baseline im Vergleich mit optimalen Testset

Table: Final Evaluation

Featureset		Accuracy
Baseline	unbalanced	0.7867578489285
	balanced	0.840802675585
Optimales Featureset	unbalanced	0.87286291576
	balanced	0.923745819398

Context bezieht auch Satzzeichen ein (oft ',' oder '.'), dies könnte man auf alphanumerische Strings beschränken.

Klassifikationsfehler im Testset, da nur die automatisch annotierte Testsetversion von OntoNotes v5 zur Verfügung steht.

Beispiel für falsch klassifizierte Instanz

```
{'ORG': [['American', 'JJ'], 'NP', ('to', 'notions')]}  
classified as ['GPE_NORP']
```


- Cho, H.; Okazaki, N. (2013): Named entity recognition with multiple segment representations. In: Information Processing & Management 49
- Derczynski, L.; Maynard, D.; Rizzo, G.; van Erp, M. (2015): Analysis of named entity recognition and linking for tweets. In: Information Processing & Management 51 (2), S. 32–49.
- Konkol, M.; Brychcín, T.; Konopík, M. (2015): Latent semantics in Named Entity Recognition. In: Expert Systems with Applications 42 (7), S. 3470–3479.
- Agerri, R.; Rigau, G. (2016): Robust multilingual Named Entity Recognition with shallow semi-supervised features. In: Artificial Intelligence 238, S. 63–82.
- Tjong Kim Sang, E.; De Meulder, F. (2003): Language-Independent Named Entity Recognition.

- Marrero, M.; Urbano, J. (2013): Named Entity Recognition. Fallacies, challenges and opportunities. In: Computer Standards & Interfaces 35 (5), S. 482–489.
- Mayfield, J.; McNamee, P. (2003): Named entity recognition using hundreds of thousands of features. In: Walter Daelemans und Miles Osborne (Hg.): Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003 -. the seventh conference. Edmonton, Canada. Morristown, NJ, USA: Association for Computational Linguistics, S. 184–187.
- Marrero, M.; Sánchez-Cuadrado, S. (2009): Evaluation of Named Entity Extraction Systems.
- Weischedel, R. (2013): OntoNotes release 5.0. [Philadelphia, Pa.]: Linguistic Data Consortium.
- Nadeau, D.; Turney, P. (2006): Unsupervised Named-Entity Recognition: Generating Gazetteers and Resolving Ambiguity. Berlin.

- Radu F.; Abe It. (2003): Named Entity Recognition through Classifier Combination. In Proceedings of CoNLL-2003.
- Chieu H. (2003): Named Entity Recognition with a Maximum Entropy Approach. In Proceedings of CoNLL-2003.
- Borthwick, A. (1999): A Maximum Entropy Approach to Named Entity Recognition, Diss., New York.
- Primadhanty, A.; Carreras, X. (2014): Low-Rank Regularization for Sparse Conjunctive Feature Spaces: An Application to Named Entity Classification. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics.
- He, Q.; Spangler, S. (2016): Semi-supervised data integration model for named entity classification. Google Patents.

- Munro, R.; Ler, D. (2003): Meta-Learning Orthographic and Contextual Models for Language Independent Named Entity Recognition. In Proceeding CONLL '03 Proceedings of the seventh conference on Natural language learning at HLT-NAACL.
- Ratinov, L.; Roth, D. (2009): Design Challenges and Misconceptions in Named Entity Recognition. In Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL).
Toral, Antonio; Munoz, Rafael (2006): A proposal to automatically build and maintain gazetteers for Named Entity Recognition by using Wikipedia.