

Semantic Argument Classification

Julian Baumann, Kevin Decker, Maximilian Müller-Eberstein

Ruprecht-Karls-Universität Heidelberg

1 Einleitung

Wem ist durch *wen*, *was*, *wo*, *wie*, *wann* widerfahren? Genau diese Fragen beantwortet die semantische Argumentklassifikation. Den Argumenten des Verbs ihre semantischen Rollen zuzuweisen ist für das tiefere Verständnis textueller Daten unumgänglich. Um diese Aufgabe bestmöglich zu lösen, testen wir verschiedene Klassifikationsalgorithmen und orientieren uns dabei an [hier unser einziges relevantes Paper einfügen].

1.1 Organisatorisches

hier Gantt-Diagramm einfügen

2 Grundlagen

2.1 Daten

PropBank in NLTK (Größe)
112917 PropBank-Instanzen/Sätze
292975 Argumente

2.2 Tools

Python 3.4
NLTK 3.0.0
Weka 3.7.11

2.3 Algorithmen

Classifiers
SVM
NaiveBayes
J48

2.4 Vergleichsgrundlagen

Paper: Support Vector Learning for Semantic Argument Classification
PropBanks goldene Annotierung
Trainings- und Testdaten aufsplitten (60% Training, 20% Test, 20% Development)

3 Hauptteil

3.1 Zielsetzung

die automatische Zuordnung mittels Supervised Learning von

ARG0	agent
ARG1	patient
ARG2	instrument, benefactive, attribute
ARG3	starting point, benefactive, attribute
ARG4	ending point
ARGM	modifier

3.2 Umsetzung

Feature-Extraktion mit Python:

```

featureList = [...] # zu extrahierende Features
for pbInstance in pbInstances :
    for pbArg in pbInstance.arguments :
        features = []
        for feature in featureList :
            featureList.append(extFeature(feature , pbArg , pbInstance))
# write features to file in ARFF

```

Features

- predicate : nominal
- path : nominal
- phrase type : nominal
- position(before/after) : boolean
- voice(active/passive) : boolean
- headword : nominal
- subcategorization : nominal

siehe Support Vector Learning for Semantic Argument Classification

3.3 Evaluation

Evaluation gegen Support Vector Learning for Semantic Argument Classification

4 Ausblick