# Automated Feature Generation from Structured Knowledge

TECHNISCHE UNIVERSITÄT DARMSTADT

**Seminar aus maschinellem Lernen WS 11/12**

**Dr. Heiko Paulheim, Frederik Janssen**

# Inhalt

TECHNISCHE UNIVERSITÄT DARMSTADT

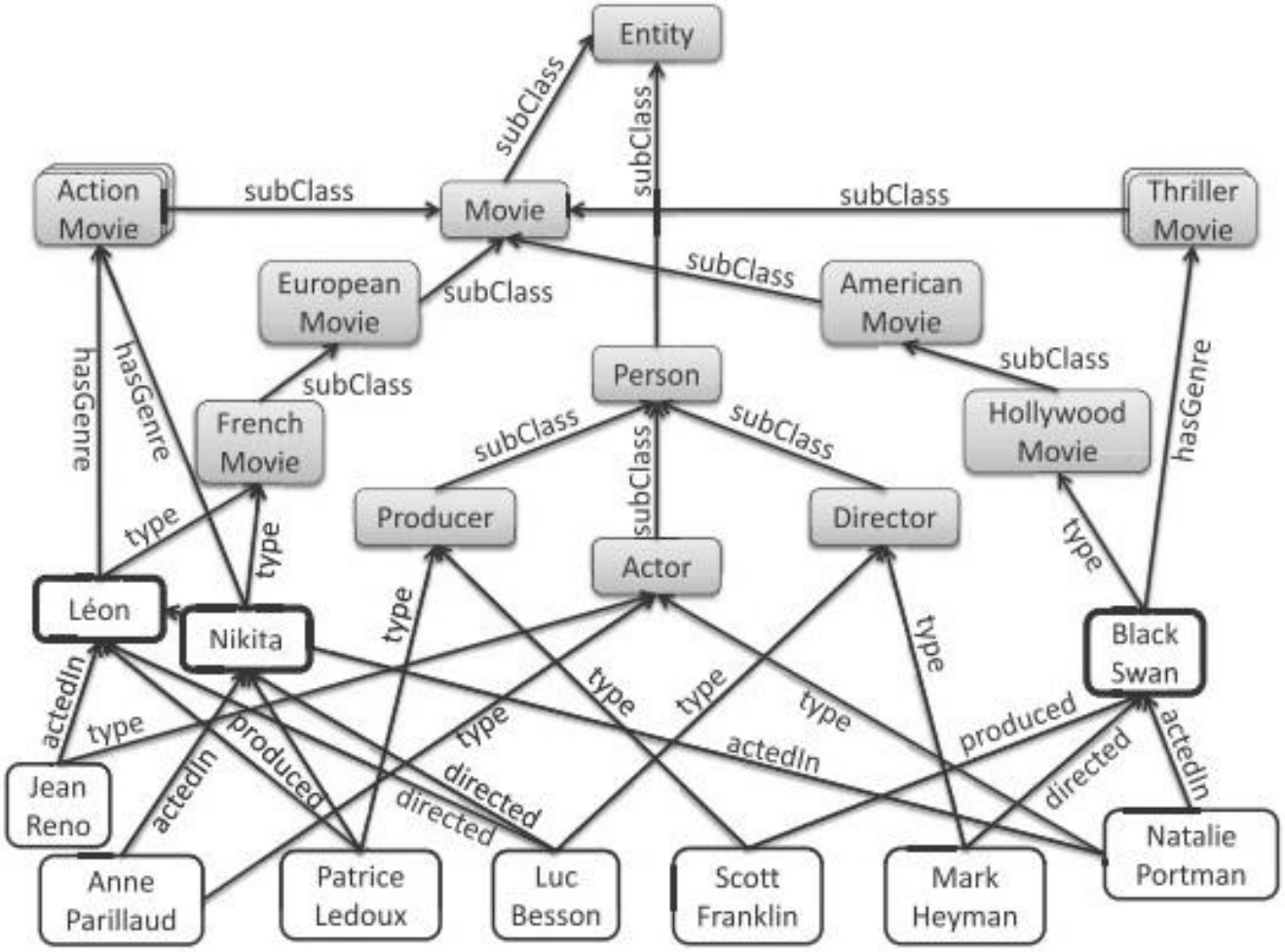
- Motivation
- Semantische Feature
  - Abfrage semantischer Feature
  - Konstruktion von Feature Vektoren
  - Hyponymie-Basierte Feature Vektoren
  - Allgemeine Semantische Feature Vektoren
- Evaluierung
  - Learning Model
  - Test Case: Vorhersage Filmfeedback
  - Test Case: Textklassifikation
- Fazit

# Motivation

TECHNISCHE UNIVERSITÄT DARMSTADT

- Accurancy jeglicher Lerner abhängig von gewählten Features
- In den letzten Jahren immer mehr Knowledge Bases aufgebaut (z.B. YAGO, DBpedia, Freebase)
- Idee
  - Automatische Extraktion der Merkmalsvektoren aus Knowledge Base
  - Knowledge Base wäre wiederverwendbarer „Feature Store“
  - Ausdrucksstarke Abfragesprache benötigt

# Semantische Feature




Quelle: [1]

# Semantische Feature

TECHNISCHE UNIVERSITÄT DARMSTADT

› Annahmen:

| | |
|---|---|
| $\mathcal{C}$ | Menge über die Vorhersage gemacht werden soll |
| $\mathcal{M}$ | Lernmethode |
| $\mathcal{K} = (\mathcal{G}, \mathcal{E}, \mathcal{R}, \mathcal{Q})$ | Knowledge Base |
| $\mathcal{E}$ | Menge Entitäten in K |
| $\mathcal{R} \subset \mathcal{E}$ | Menge Relationen in K |
| $\mathcal{G} \subseteq \mathcal{E} \times \mathcal{R} \times \mathcal{E}$ | Knowledge Graph |
| $\mathcal{Q}$ | Abfragesprache |

# Semantische Feature

TECHNISCHE UNIVERSITÄT DARMSTADT

› Zwei Schritte zur Konstruktion semantischer Merkmalsvektoren

1. Extrahiere semantische Feature aus Knowledge Base zu einer Entität x
2. Automatischer Mechanismus zur Generierung des Feature Vektors $\phi(x)$

# Abfrage semantischer Feature

TECHNISCHE UNIVERSITÄT DARMSTADT

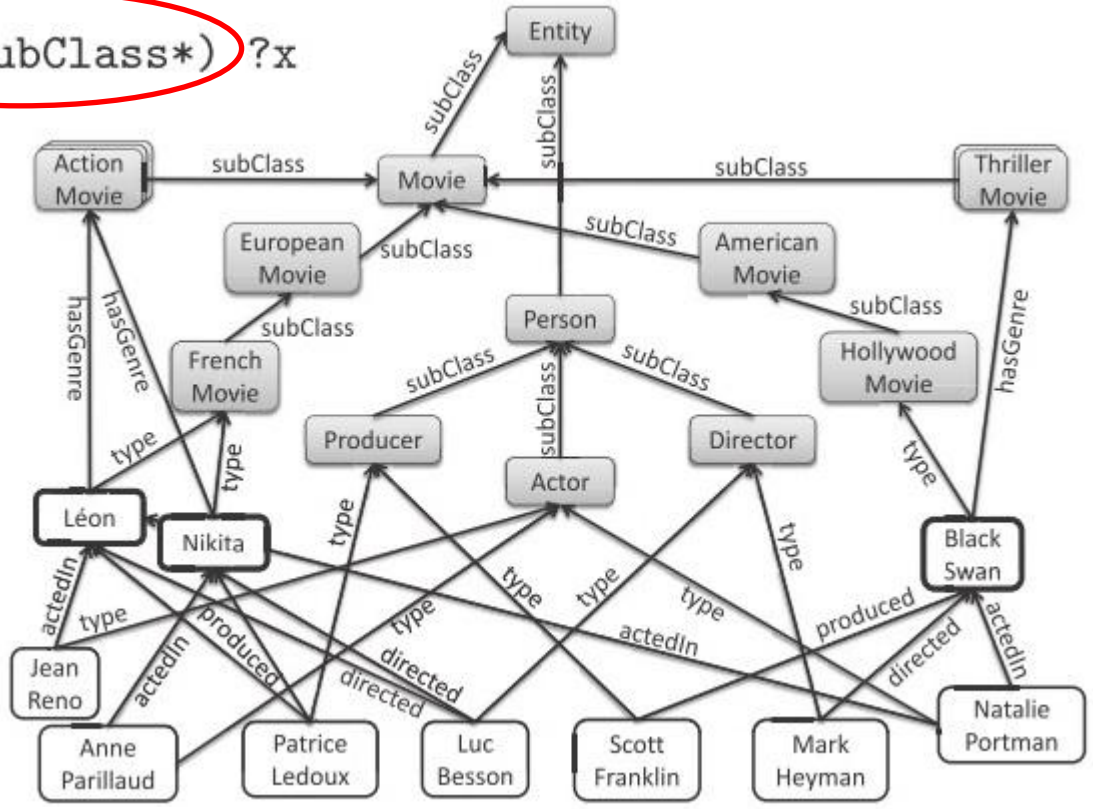
- Abfrage als semantischer Graph aus Entitäten, Relationen und Variablen
- Bekannte Abfragesprachen SPARQL[1] und NADA[2]
  - SPARQL: erlaubt Projektionen (SELECT) und Aggregationen (z.B. COUNT, SUM, MAX)
  - NADA: erlaubt reguläre Ausdrücke über Relationen in den Abfragekanten, dadurch können ganze Pfade aus G mit Abfrage abgeglichen werden

[1] vgl. [3]
[2] Details, siehe [4]

# Beispiel NAGA Abfrage

TECHNISCHE UNIVERSITÄT DARMSTADT

```
Black_Swan hasGenre (type subClass*) ?x
```




```
Black_Swan  type  Hollywood_Movie
Hollywood_Movie  subClass  American_Movie
American_Movie  subClass  Movie
```

# ESPARQL

TECHNISCHE UNIVERSITÄT DARMSTADT

› Ausdrucksstarke Abfragesprache zur Extraktion semantischer Feature nötig:
  – Kombination aus SPARQL und NADA
  – **E**xtended **SPARQL** Query Language

› Beispielabfragen:

```
SELECT DISTINCT ?x
WHERE { e  type subClass*  ?x }
```

Alle Superklassen von e

```
SELECT COUNT (DISTINCT ?x)
WHERE { ?x  starredIn|directed  m.
        ?x  hasWon  Oscar_Award  }
```

Anzahl Darsteller/Regisseure die Oscar gewonnen und an m mitgewirkt haben

# Automatische Konstruktion von Feature Vektoren

TECHNISCHE UNIVERSITÄT DARMSTADT

- bekannte Methoden zur Lösung überwachter Lernaufgaben (supervised learning) nutzen Feature Vektor-Darstellung der Inputdaten (z.B. k-nearest neighbors)

- Ziel der Autoren:
  Generische Methode zur Konstruktion dieser Feature Vektoren aus semantischen Informationen aus einer Knowledge Base für eine gegebene Entität

# Automatische Konstruktion von Feature Vektoren

TECHNISCHE UNIVERSITÄT DARMSTADT

| | |
|---|---|
| $\mathcal{Y}$ | Menge der Kategorien |
| $S \subseteq \mathcal{E} \times \mathcal{Y}$ | Trainingsmenge |
| $\phi : \mathcal{E} \rightarrow \mathcal{F}$ | Abbildung von Entitäten zu Feature Vektoren |
| $e \in \mathcal{E}$ | Entitäten |
| $q(e)$ | ESPARQL Query zur Extraktion sem. Feature von e |

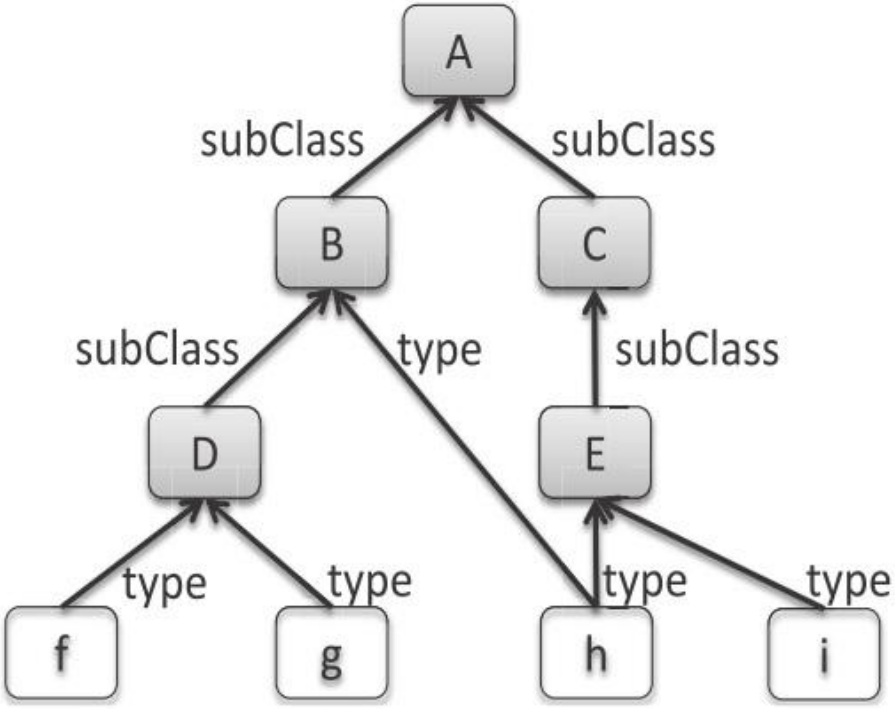
- q(e) enthält k Variablen
- Jede Substitution einer Variable wird ein sem. Feature von e repräsentieren
- Alle „Antworten“ auf Trainingsmenge werden als Feature verwendet

# Automatische Konstruktion von Feature Vektoren - Schritte

TECHNISCHE UNIVERSITÄT DARMSTADT

- Wähle angemessene ESPARQL-Abfragen
- Vereinige Antwort-Mengen der Abfragen für alle Entitäten aus Trainingsmenge zu den Dimensionen des Feature-Raums $F_s$
- Setze die Dimensionen, die mit Abfrageantworten korrespondieren im Merkmalsvektor auf 1

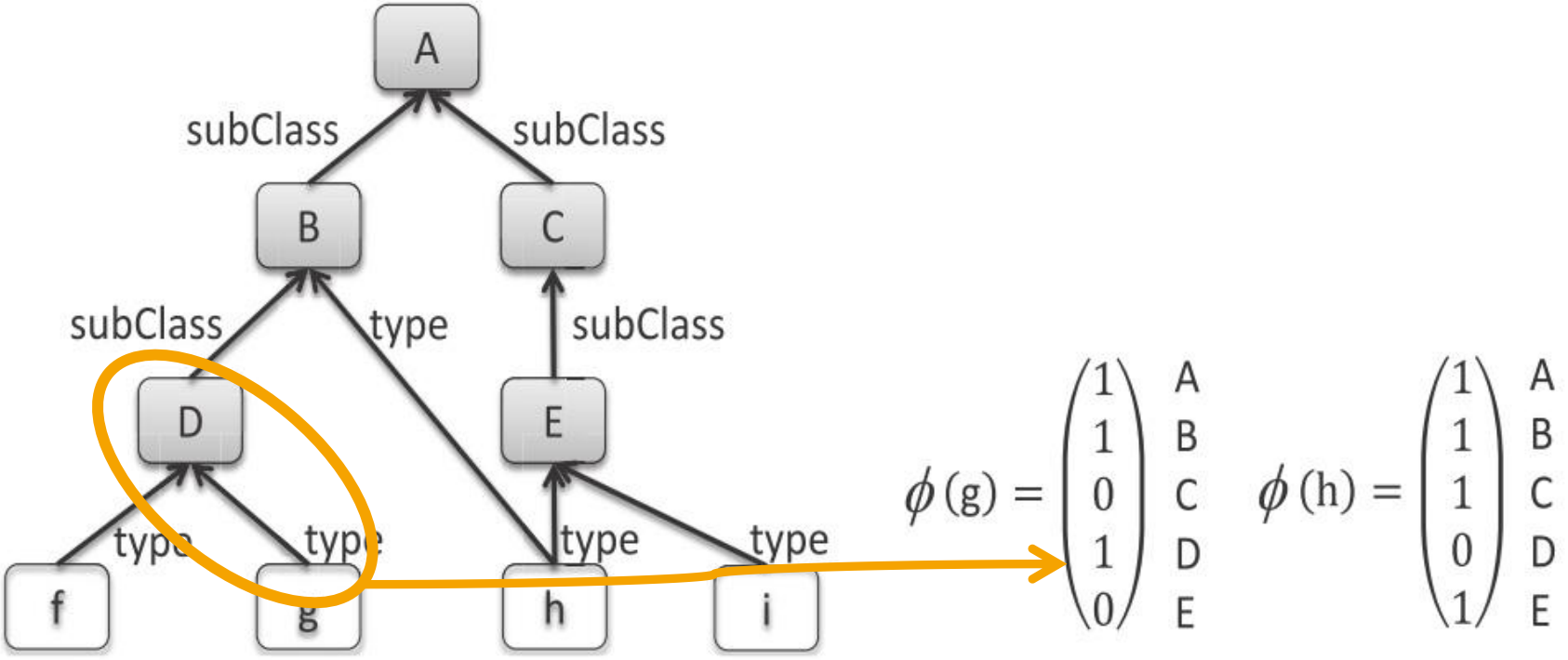
# Automatische Konstruktion von Feature Vektoren - Beispiel




$$\phi(g) = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} \begin{matrix} A \\ B \\ C \\ D \\ E \end{matrix} \qquad \phi(h) = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 1 \end{pmatrix} \begin{matrix} A \\ B \\ C \\ D \\ E \end{matrix}$$

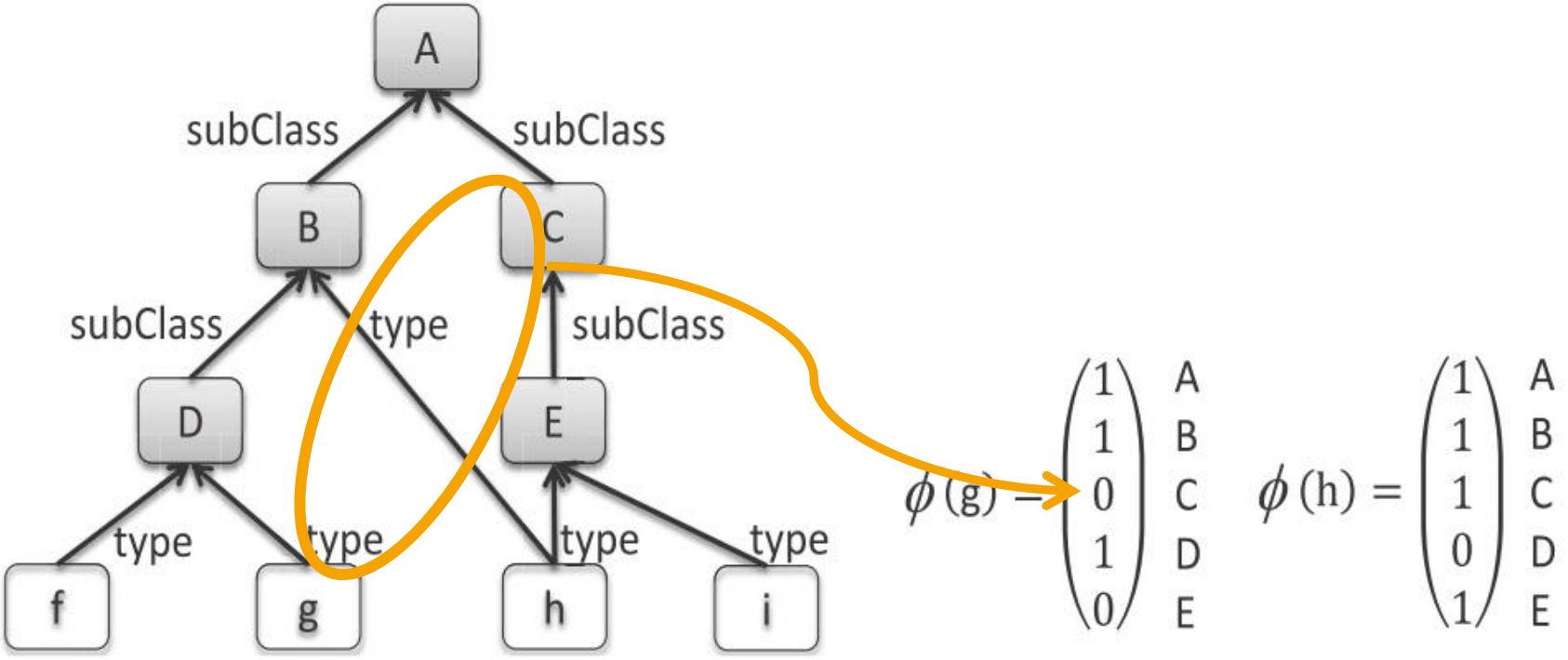
Trainingsmenge: {g, h}
Feature Menge $F_s$ = {A, B, C, D, E}

# Automatische Konstruktion von Feature Vektoren - Beispiel




Trainingsmenge: {g, h}
Feature Menge $F_s$ = {A, B, C, D, E}

# Automatische Konstruktion von Feature Vektoren - Beispiel




Trainingsmenge: {g, h}
Feature Menge $F_s$ = {A, B, C, D, E}

# Hyponymie-Basierte Feature Vektoren

TECHNISCHE UNIVERSITÄT DARMSTADT

- Können von allen überwachten Lernern benutzt werden
- Jedoch optimal für solche die Abhängigkeiten beachten
  - Z.B. Präsident ist immer auch Person
  - Naive Bayes unterstellt Unabhängigkeit
  - Führt bei Hyponymie-Basierten Feature zu schlechterer Performance

- q(e):

```
SELECT ?c
WHERE { h  type subClass*  ?c }
```

# Allgemeine Semantische Feature Vektoren

TECHNISCHE UNIVERSITÄT DARMSTADT

- Knowledge Bases in Praxis oft sehr groß
- Feature Space kann sehr groß werden

- Lösung:
  - **Restricted Entity Domain** $\mathcal{D}_T$
  - Nur Typen von Entitäten betrachten, die für vorliegende Aufgabe relevant sind
  - D.h. nur Unterklassen und Individuen einer Teilmenge T von $\mathcal{E}$

- Allgemeine Query zur semantischen Nachbarschaft

```
SELECT ?e ?r
WHERE { (m   ?r  ?e) UNION (?e  ?r  m) }
```

# Evaluierung

TECHNISCHE UNIVERSITÄT DARMSTADT

- Bewertung der Leistungsfähigkeit der generierten semantischen Feature
- Daten aus YAGO, Movielens, Twitternachrichten
- Zwei verschiedene Aufgaben
  - Filmempfehlung
  - Textklassifikation

# Evaluierung: Learning Model

TECHNISCHE UNIVERSITÄT DARMSTADT

- Generalisiertes lineares Bayes‘sches Probitmodell
- Eingabewerte x als Merkmalsvektor

$$\phi(x) = (\phi_1(x), \ldots, \phi_n(x))^T \in \mathcal{F}_S$$

- Merkmale gewichtet, $w_i$ normalverteilt

$$\mathbf{w} = (w_1, \ldots, w_n)$$

- Vorteile des Modells
  - Abhängigkeiten zwischen Features werden erfasst
  - Kann große Feature Vektoren verarbeiten

# Test Case 1: Vorhersage Filmfeedback

TECHNISCHE UNIVERSITÄT DARMSTADT

- Aufgabe
  - Vorhersage des Feedbacks eines Users für einen Film
  - Leichter als aktuelle state-of-the-art Empfehlungen
  - Hier Fokus auf die Vorteile der verschiedenen Feature Typen
- Aufgabenstellung interessant, wenn
  - Wenige Nutzer und Filme in Trainingsmenge
    - **Cold Start Problem**
- Daten
  - MovieLens: 3.952 Filme, 1.000.206 Bewertungen von 6.040 Nutzern
  - YAGO: Konstruktion der semantischen Feature (Budget, Veröffentlichung,..)

# Vorhersage Filmfeedback: Resultate

TECHNISCHE UNIVERSITÄT DARMSTADT

- Zwei verschiedene Szenarien
  - Keine semantischen Feature benutzt
  - Das Modell benutzt zusätzlich semantische Feature zu den Filmen

- Methode zur Messung der Accurancy
  - Sortiere Filme nach Veröffentlichung in zwei Mengen (Trainings- und Testmenge)
  - Lerne auf Trainingsmenge
  - Dann wird Lerner für jeden Film in Testmenge auf zufälliger Teilmenge L der Bewertungen (L = 5% - 50%) weiter trainiert
  - Experimente 10x wiederholt, dann Mean Absolute Error (MAE) bestimmt

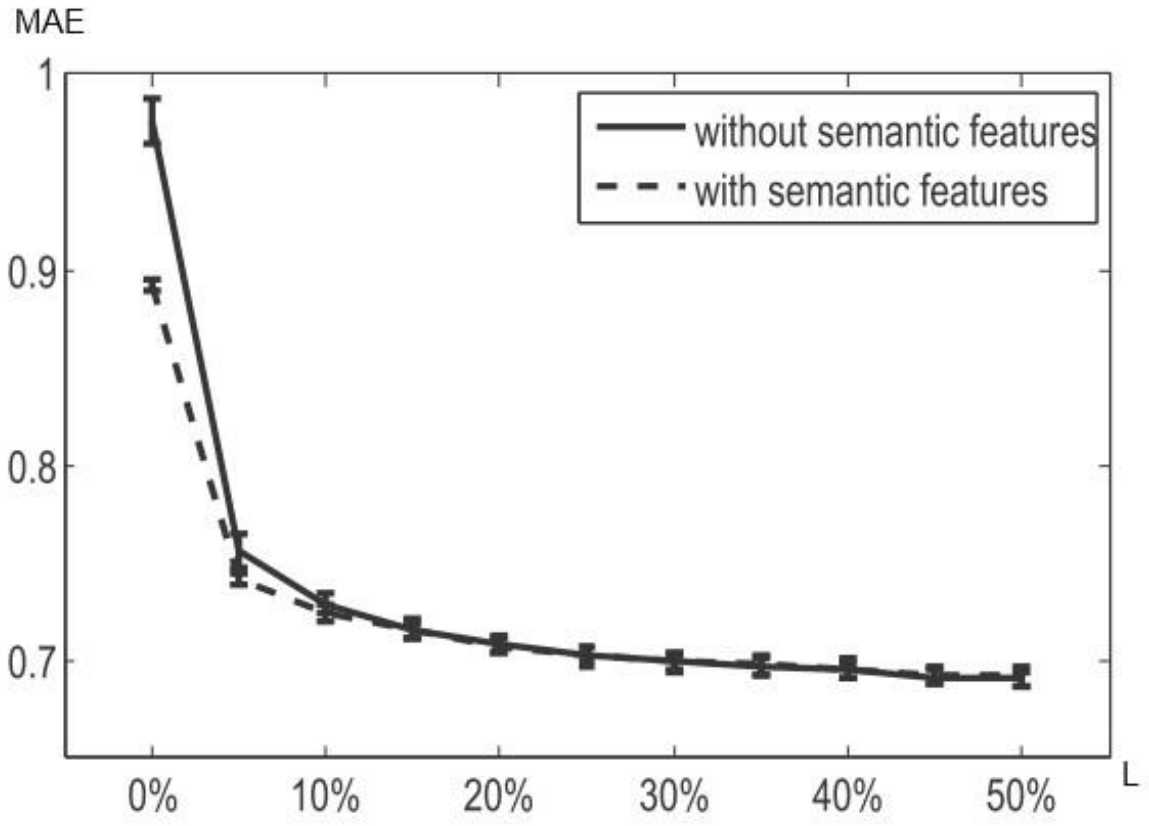
# Vorhersage Filmfeedback: Resultate

TECHNISCHE UNIVERSITÄT DARMSTADT

$|\hat{r}_i - r_i|$ MAE

$r_i$ Echte Bewertung

$\hat{r}_i$ Geschätzte Bewertung




Besonders bei wenigen gegebenen Bewertungen kann mit semantischen Merkmalen Verbesserung erreicht werden (wird vernachlässigbar, wenn Anzahl Feedback steigt)

# Vorhersage Filmfeedback: Resultate

TECHNISCHE UNIVERSITÄT DARMSTADT

- Nicht das volle Potenzial ausgeschöpft
  - Es könnte weitere informative semantische Feature geben
  - Semantische Feature über Nutzer könnten weitere Verbesserung bringen
  - YAGO ist eher allgemein – bessere Performance eventuelle durch themenspezifische Wissensbasis möglich

# Test Case 2: Tweet Klassifikation

TECHNISCHE UNIVERSITÄT DARMSTADT

- Ziel
  - Tweets einer oder mehreren von 9 Klassen zuordnen
- Problem
  - Wenige Worte (Tweet auf 140 Zeichen beschränkt)
  - Oft wenige informative Begriffe (inbes. bei Statusmeldungen)
  - Oft unbekannte Begriffe (neue Firmen, Produkte, o.ä.)
  - Verteilung auf Klassen sehr unausgeglichen
- Datenmenge
  - 22.816 Tweets mit Klasseninformation

# Tweet Klassifikation: Daten

TECHNISCHE UNIVERSITÄT DARMSTADT

| class | #instance | proportion |
|---|---|---|
| Business/Finance (BF) | 2182 | 9.56% |
| Entertainment (E) | 4235 | 18.56% |
| Lifestyle (LS) | 4085 | 17.90% |
| Politics (P) | 1199 | 5.25% |
| Science/Environment (SE) | 789 | 3.45% |
| Sport (S) | 1145 | 5.01% |
| Technology (T) | 1880 | 8.23% |
| World Events (WE) | 2122 | 9.30% |
| Other/Miscellaneous (OM) | 12838 | 56.26% |

| tweet | BF | E | LS | P | SE | S | T | WE | OM |
|---|---|---|---|---|---|---|---|---|---|
| Obama blames Bush for all of his misdeeds and then takes credit for the successful war in Iraq http://is.gd/e0iVM (via @PennyStarrDC) | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| A Modern Approach to an Ancient Game: The story behind The Path of Go http://bit.ly/g04KtA | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| @alex_ Good morning! How was the trip? | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

# Tweet Klassifikation: Feature Generation

TECHNISCHE UNIVERSITÄT DARMSTADT

- Konventionelle Feature in Textklassifikation: Bag-of-Word-Feature
  - Können direkt aus dem Tweet extrahiert werden
- Semantische Feature
  - Zunächst Entitäten im Tweet bestimmen
  - Passendste Entität in YAGO suchen ( YAGO means Relation)
  - Zu diesen Entitäten können semantische Feature extrahiert werden
- Beispiel:
  - *Bush* im ersten Tweet wird als Entität erkannt und auf YAGO Entität *George W. Bush* abgebildet

# Tweet Klassifikation: Evaluierte Modelle

TECHNISCHE UNIVERSITÄT DARMSTADT

- 2 Modelle: Probitmodell und Naive Bayes Modell
- Feature-Mengen:
  - Bag-of-words
  - Semantische Feature inklusive/exklusive Hyponymie-basierter Feature
- 5 Szenarien
  - **SH:** Probitmodell, Bag-of-words ,semantische Merkmale inkl. Hyperonymen
  - **SnH**: Probitmodell, Bag-of-words, semantische Merkmale ohne Hyperonyme
  - **BOW**: Probitmodell, nur Bag-of-words-Mermale
  - **NBSH**: Naive Bayes, Bag-of-words, semantische Merkmale inkl. Hyperonymen
  - **NBBOW**: Naive Bayes, nur Bag-of-words-Merkmale

# Tweet Klassifikation: Resultate

TECHNISCHE UNIVERSITÄT DARMSTADT

- Jedes Modell wird unabhängig trainiert und getestet
- Bewertung der Vorhersagen  mit
  - negative log-likelihood (NLL)
  - Fläche unter ROC-Kurve (AUC)

# Tweet Klassifikation: Resultate

TECHNISCHE UNIVERSITÄT DARMSTADT

› Vergleich basierend auf NLL

| class | SH | SnH | BOW | NBSH | NBBOW |
|---|---|---|---|---|---|
| Business/Finance | 0.1663±.0083(3) | **0.1569±.0073**(1) | 0.1637±.0076(2) | 2.3281±.0575(4) | 2.3511±.0350(5) |
| Entertainment | 0.2993±.0057(2) | **0.2816±.0134**(1) | 0.3000±.0117(3) | 2.4119±.0795(5) | 1.1625±.0358(4) |
| Lifestyle | 0.3499±.0145(2) | **0.3402±.0058**(1) | 0.3609±.0154(3) | 2.3222±.0790(5) | 1.2885±.0314(4) |
| Politics | 0.1074±.0047(3) | 0.1018±.0053(2) | **0.0990±.0042**(1) | 5.1388±.0453(5) | 3.7856±.0799(4) |
| Science/Environment | 0.0935±.0076(2) | **0.0920±.0090(1)** | 0.0938±.0069(3) | 6.5148±.0654(5) | 5.1472±.0691(4) |
| Sport | **0.0839±.0109**(1) | 0.0874±.0087(3) | 0.0862±.0029(2) | 5.7259±.0407(5) | 3.9350±.0608(4) |
| Technology | 0.1311±.0068(3) | 0.1272±.0084(2) | **0.1255±.0116**(1) | 3.0432±.0728(5) | 2.3635±.0588(4) |
| World Events | 0.1752±.0056(2) | **0.1693±.0111**(1) | 0.1783±.0078(3) | 3.2537±.1058(5) | 2.4399±.0555(4) |
| Other/Miscellaneous | 0.4101±.0106(2) | **0.4054±.0073**(1) | 0.4316±.0083(3) | 1.0783±.0560(5) | 0.8987±.0333(4) |
| average rank | 2.22 | 1.44 | 2.33 | 4.89 | 4.11 |

› Vergleich basierend auf AUC

| class | SH | SnH | BOW | NBSH | NBBOW |
|---|---|---|---|---|---|
| Business/Finance | .9238±.0101(3) | **.9395±.0090**(1) | .9335±.0094(2) | .8289±.0115(5) | .8947±.0133(4) |
| Entertainment | .8987±.0047(2) | **.9124±.0077**(1) | .8973±.0085(3) | .7984±.0150(5) | .8607±.0114(4) |
| Lifestyle | .8393±.0147(2) | **.8534±.0068**(1) | .8370±.0112(3) | .7554±.0140(5) | .7927±.0107(4) |
| Politics | .9464±.0129(3) | .9567±.0056(2) | **.9592±.0085**(1) | .8114±.0105(5) | .9004±.0133(4) |
| Science/Environment | .9051±.0266(3) | **.9251±.0136**(1) | .9153±.0137(2) | .7372±.0156(5) | .8239±.0178(4) |
| Sport | .9595±.0154(2) | **.9616±.0091**(1) | .9592±.0115(3) | .7869±.0214(5) | .8784±.0160(4) |
| Technology | .9484±.0122(3) | .9560±.0043(2) | **.9571±.0092**(1) | .8385±.0143(5) | .9138±.0107(4) |
| World Events | .9167±.0037(3) | **.9263±.0061**(1) | .9170±.0102(2) | .8226±.0138(5) | .8762±.0161(4) |
| Other/Miscellaneous | .8909±.0072(2) | **.8955±.0055**(1) | .8815±.0064(3) | .8458±.0110(5) | .8798±.0064(4) |
| average rank | 2.56 | 1.22 | 2.22 | 5.00 | 4.00 |

# Tweet Klassifikation: Resultate

TECHNISCHE UNIVERSITÄT DARMSTADT

- Vergleich der Tests mit Probitmodell zeigt
  - Sematische Feature verbessern das Ergebnis gegenüber bag-of-words
  - Hyponymie basierte Feature in diesem Fall nicht informativ
- Güte zwischen den Klassen variiert stark
  - SNH verbessert Ergebnis für die meisten Klassen
  - Politics und Technology scheinbar schwer zu verbessern
    - Oft Kommentare zu neueste Entwicklungen/Trends
    - Entitäten noch nicht gut abgedeckt durch YAGO
- Güte des Naive Bayes schlecht
  - NBSH gibt die schlechtesten Resultate

# Fazit

TECHNISCHE UNIVERSITÄT DARMSTADT

- Ansatz um Lücke zwischen Maschinellem Lernen und Semantischen Technologien zu schließen
- Verbesserungspotentiale vorhanden
- Selektion der Feature kaum beachtet
  - Konstruktion der semantischen Merkmale nicht komplett automatisch
  - Geeignete Queries müssen manuell gestellt werden
  - Erfordert Wissen über die zugrunde liegende Wissensbasis (vgl. [2])
  - Keine Aussage wer die Abfragen konzipieren sollte

TECHNISCHE UNIVERSITÄT DARMSTADT

Vielen Dank für die

Aufmerksamkeit!

# Literatur


[1] Cheng, W., Kasneci, G., Graepel, T., Stern, D., Herbrich, R.: *Automated Feature Generation from Structured Knowledge*. In: Proceedings of the 20th ACM Conference on Information and Knowledge Management, pp. 1395-1404 (2011)

[2] Paulheim, H., Fürnkranz, J.: *Unsupervised Generation of Data Mining Features from Linked Open Data.* Technical Report TUD–KE–2011–2 (2011)

[3] Paulheim, H.: Skript *Teil 5: SPARQL,* Vorlesung „Semantic Web“, WS 2011/12

[4] Kasneci, G., Suchanek, F. M., Ifrim, G., Ramanath, M., Weikum, G.: *NAGA: Searching and Ranking Knowledge*. In: 24th International Conference on Data Engineering, pp. 953–962. IEEE (2008)

# Anhang - Restricted Entity Domain

TECHNISCHE UNIVERSITÄT DARMSTADT

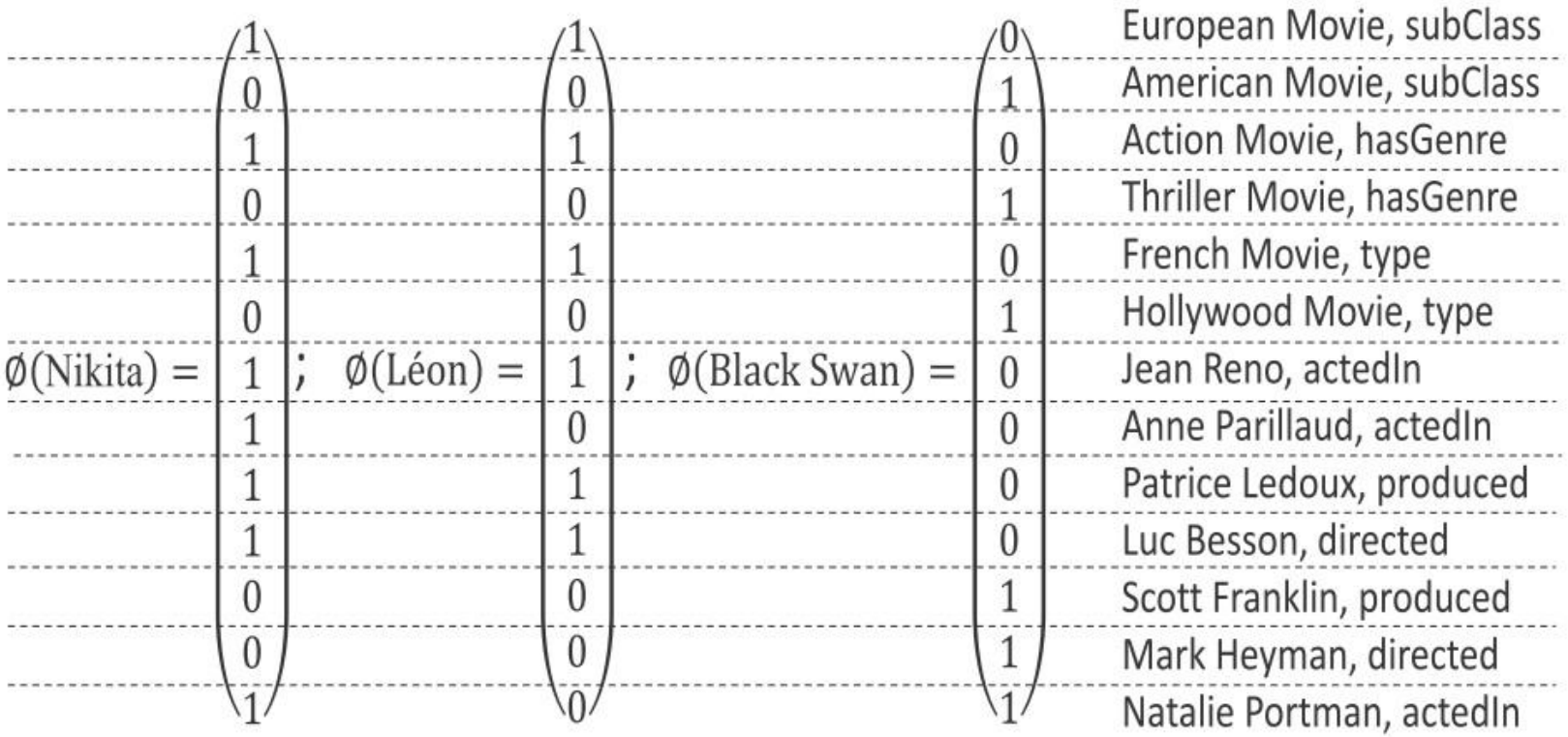
DEFINITION 3.4. **Restricted Entity Domain.** *Let* $T \subseteq \mathcal{E}$ *be a set of types from a knowledge base* $\mathcal{K} = (\mathcal{G}, \mathcal{E}, \mathcal{R}, \mathcal{Q})$. *For a class* $c \in T$, *let* $D_c$ *denote the set of entities that can be substituted for the variable* `?x` *in the following ESPARQL query on* $\mathcal{K}$*:*

```
SELECT ?x
WHERE { ?x  subClass|(type subClass*)  c }
```

*The* restricted entity domain $\mathcal{D}_T$ *for* $T$ *is defined as* $\mathcal{D}_T = \bigcup_{c \in T} D_c$.

Nur die Typen von Entitäten in Wissensbasis betrachten, die für vorliegende Arbeit relevant sind, d.h. alle Unterklassen und Individuen einer Teilmenge von T vor $\mathcal{E}$

# Anhang – Allgemeine Feature Vektoren




Die Elemente der Vereinigung der semantischen Nachbarschaftsinformation ergibt die Dimensionen der Feature Vektoren der Filme Nikita, Leon, Black Swan.
Die größere Ähnlichkeit von Nikita und Leon zeigt sich auch in der Ähnlichkeit ihrer Vektoren

# Anhang: Learning Model

TECHNISCHE UNIVERSITÄT DARMSTADT

› Generalisiertes lineares Bayes‘sches Probitmodell

$$\phi(x) = (\phi_1(x), \ldots, \phi_n(x))^T \in \mathcal{F}_S$$

Binärer Feature Vektor

$$\mathbf{w} = (w_1, \ldots, w_n)$$

Gewichtungsvektor

$$p(\mathbf{w}) = \prod_i \mathcal{N}(w_i; \mu_i, \sigma_i^2)$$

Wahrscheinlichkeit eines Gewichts

$$p(s|\mathbf{w}, x) = \mathcal{N}\left(s; \sum_{i=1}^{n} \phi_i(x)\mu_i, \sum_{i=1}^{n} \phi_i(x)\sigma_i^2\right)$$

Score einer Entität x

$$P(y \mid s) = \Phi\left(\frac{ys}{\beta}\right)$$

Wahrscheinlichkeit für Klasse y

# Anhang: Abfrage und Antwort

TECHNISCHE UNIVERSITÄT DARMSTADT

DEFINITION 3.1. **Basic Query.** *A query over a set of variables $\mathcal{V}$ for a knowledge graph $\mathcal{G} \subseteq \mathcal{E} \times \mathcal{R} \times \mathcal{E}$ is a semantic connected graph $q \subseteq (\mathcal{E} \cup \mathcal{V}) \times (\mathcal{R} \cup \mathcal{V}) \times (\mathcal{E} \cup \mathcal{V})$.*

DEFINITION 3.2. **Query Answer.** *An answer to a query $q$ on a semantic graph $\mathcal{G}$ is a graph homomorphism $\sigma : q \mapsto \sigma_q \subseteq \mathcal{G}$ that preserves the given entity and relationships in $q$, and substitutes the variables with entities and relationships from $\mathcal{G}$.*

Quelle: [1]

# Anhang: NLL und AUC

TECHNISCHE UNIVERSITÄT DARMSTADT

$$\mathrm{NLL} = -\left(y_i \log(h(x_i)) + (1 - y_i) \log(1 - h(x_i))\right)$$

$$\mathrm{AUC} = \frac{1}{|P|\,|N|} \sum_{x_i \in P} \sum_{x_j \in N} \omega(h(x_i) - h(x_j)),$$

$y_i \in \{0, 1\}$ Echte Klasse von i

$h(x_i) \in [0, 1]$ Vorhergesagte Klasse

$$\omega(z) = \begin{cases} 1 & \text{if } z > 0, \\ 0.5 & \text{if } z = 0, \\ 0 & \text{if } z < 0. \end{cases}$$