

Proseminar Lexikalische Semantik

Automatic Word Sense Discrimination

Patricia Helmich

Basiert auf dem Paper: Schütze, Hinrich. (1998): Automatic Word
Sense Discrimination. Computational Linguistics 24. S.97-123

Gliederung

1. Einführung
2. Der Algorithmus: Context-Group Discrimination
3. Experimente
4. Information Retrieval
5. Begründung und Probleme des Algorithmus

1. Einführung

Word Sense Disambiguation:

zwei Teilprobleme

- **Sense discrimination:**

Teilt Vorkommen eines ambigen Wortes in Klassen, wobei die Elemente einer Klasse dieselbe Bedeutung haben.

- **Sense labeling:**

Weist jeder Klasse eine Bedeutung zu.

Das hier behandelte Teilproblem:

Sense Discrimination

- Leichter als vollständige Word Sense Disambiguation
- Keine externen Wissensquellen notwendig
 - **Automatic** Word Sense Discrimination

Anwendungen

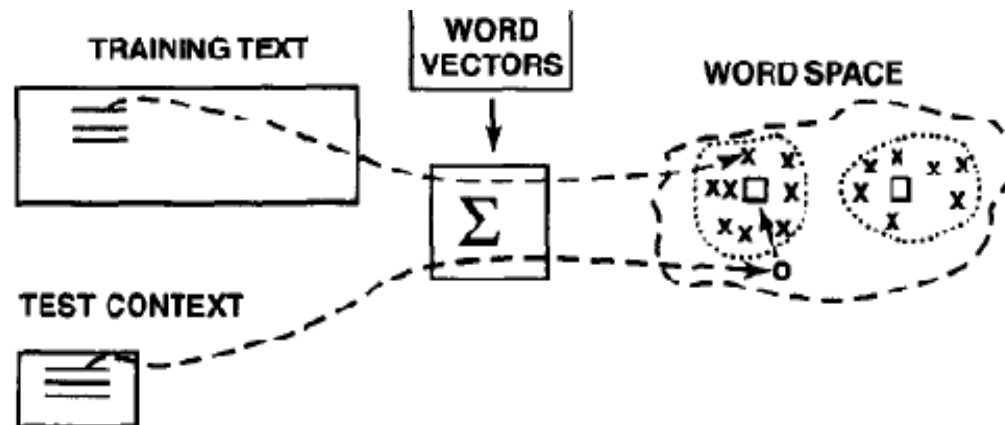
- Vollständige Word Sense Disambiguation:
 - z.B. Maschinelle Übersetzung
- Teilkomponente Word Sense Discrimination:
 - Information Acces

2. Der Algorithmus:

Context-Group Discrimination

Grundidee

- Gruppirt die Vorkommen eines ambigen Wortes, welche in ähnlichen Kontexten vorkommen in Cluster
→ Cluster wird als eine Bedeutung („Sense“) interpretiert
- Wörter, Kontexte, Bedeutungen: dargestellt als Vektoren
- jedes Wort wird durch einen Word Vector dargestellt
- Vorkommen eines ambigen Wortes: wird auf seinen Kontext abgebildet
- Kontext wird durch Context Vector dargestellt, Mittelpunkt aller Word Vectors der Kontextwörter
- Context Vectors werden in Gruppen geclustert
- Eine Gruppe („Cluster“) entspricht bestimmter Bedeutung



Drei Einheiten: Wörter, Kontexte, Bedeutungen

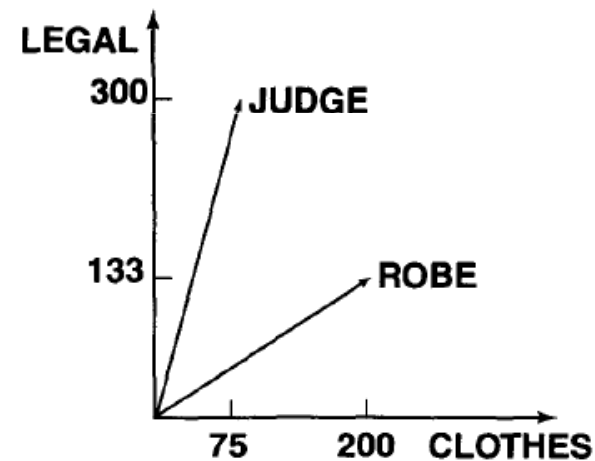
- **Word Vectors:** abgeleitet aus Nachbarwörtern
- **Context Vectors:** abgeleitet von Word Vectors der Kontextwörter
- **Sense Vectors:** abgeleitet durch die Verteilung der Context Vectors und den entstehenden Clustern

→ **Vektorraum:** jede Dimension stellt ein Wort da

Word Vector

- abgeleitet von den Nachbarwörtern von Wort w
- einfachster Fall: Vektor hat einen Eintrag für jedes Wort im Korpus
→ *Word Space*

Dimension	Vector	
	<i>judge</i>	<i>robe</i>
<i>legal</i>	300	133
<i>clothes</i>	75	200

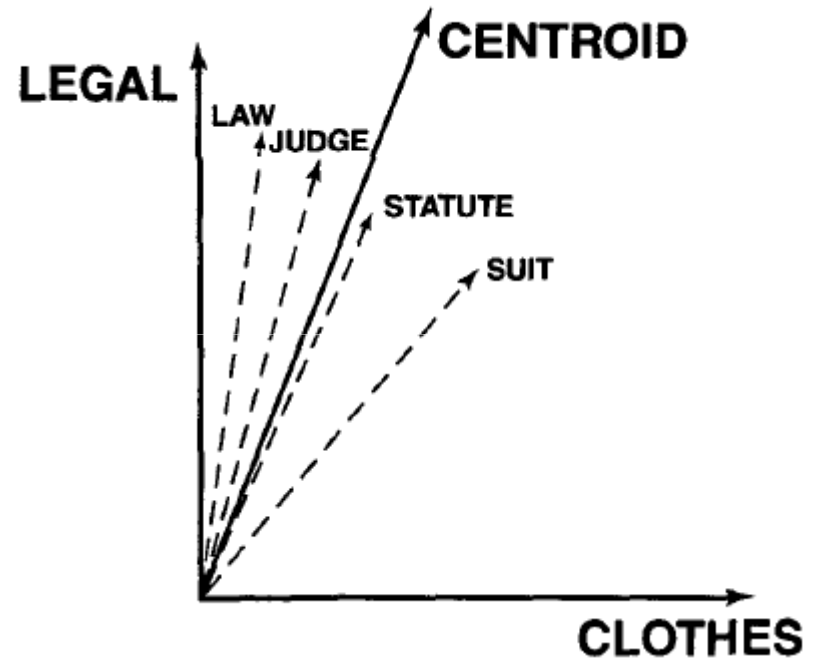


- Semantische Ähnlichkeit von zwei Wörtern:
→ Winkel zwischen den Vektoren

$$\cos \lambda = \frac{\vec{a} * \vec{b}}{|\vec{a}| * |\vec{b}|}$$

Context Vector

- *Word Vectors* fassen verschiedene Bedeutungen zusammen
- *Context Vectors* stellen einen bestimmten Kontext dar
- der *Context Vector* eines Vorkommens t eines ambigen Wortes ist das Zentrum (der Mittelpunkt) aller Word Vectors der Worte, die im Kontext von t vorkommen



Unterscheidungspotential der Word Vectors

- Gewichtung der Word Vectors bei Berechnung eines Context Vectors:
→ grober Maßstab, wie gut ein Wort w bei der Bedeutungsunterscheidung des Vorkommens eines ambigen Wortes hilft:

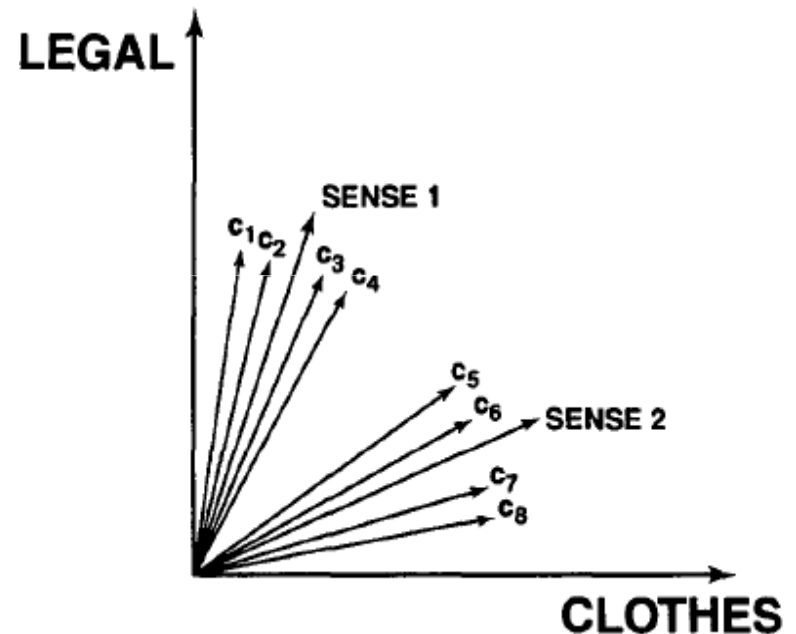
$$a_i = \log \left(\frac{N}{n_i} \right)$$

n_i = Anzahl der Dokumente in denen w vorkommt
 N = Gesamtanzahl der Dokumente

- Wörter mit großem Unterscheidungspotenzial: z.B. *automobile* oder *China*
- Wörter mit kleinem Unterscheidungspotenzial: z.B. *idea* oder *help*

Sense Vector

- eine Bedeutung („Sense“) ist eine Gruppe von ähnlichen Kontexten
- für jeden Kontext (jedes Vorkommen) eines ambigen Wortes im Korpus wird ein Context Vector berechnet
- diese Context Vectors werden in n kohärente Gruppen geclustert (mit z.B. $n = 2$),
mithilfe von *Buckshot*, eine Kombination aus dem *EM-Algorithmus* und dem *group-average agglomerative clustering (GAAC)*
- eine Bedeutung wird dann durch das Zentrum (Mittelpunkt) eines Clusters dargestellt



Anwendung der Context Group Discrimination

Sei t ein Vorkommen des ambigen Wortes v im Testkorpus:

- Bilde t durch Nutzen der Word Vectors aller Wörter in t 's Kontext auf seinen Context Vector c in Word Space ab
- Suche alle Sense Vectors s_j von v
- Weise t der Bedeutung j zu, deren Sense Vector s_j dem Context Vector c am nächsten liegt

Ergebnis: Vorkommen t hat die Bedeutung j

3. Experimente

Evaluation

- Context-Group Discrimination wird mit 10 natürlichen und 10 künstlichen ambigen Wörtern getestet
- Warum künstliche Wörter und wie sehen solche Wörter aus?
 - ein gutes Mittel, um Algorithmen zur Bedeutungsunterscheidung zu testen
 - zeitaufwändig, alle Vorkommen eines ambigen Wortes zu klassifizieren
 - bei künstlichen Wörtern ist Bedeutungsüberprüfung kein Problem, man „bastelt“ sich z.B. aus *banana* und *door* ein neues Wort (*banana/door*) und ersetzt alle Vorkommen der beiden Wörter durch dieses Wort
 - Bedeutung eines jeden Vorkommens ist klar, man braucht nur im Ursprungstext nachzuschauen

Beispiel für 4 künstliche und 4 natürliche Wörter

Word	Training	Test	Rare Senses	Baseline	Frequent Senses
<i>wide range / consulting firm</i>	1,422	149	0%	62%	wide range consulting firm
<i>urban development / cease fire</i>	1,582	101	0%	50%	urban development cease fire
<i>economic development / right field</i>	1,030	88	0%	68%	economic development right field
<i>national park/ judiciary committee</i>	1,279	122	0%	70%	national park judiciary committee
<i>motion</i>	2,705	200	0%	55%	movement proposal for action
<i>suit/s</i>	7,467	200	12,5%	57%	an action or process in a court a set of garments
<i>tank/s</i>	3,909	200	4,5%	90%	a combat vehicle a receptacle for liquids
<i>train/s</i>	4,271	200	1,5%	74%	a line of railroad cars to teach

Ergebnisse der Experimente

	Global	
	Frequency	
	SVD	
	2	
	μ	σ
wide range / consulting firm	87	3
urban development / cease fire	100	0
economic development / right field	100	0
national park/ judiciary committee	100	0
motion	86	5
suit/s	95	0
tank/s	85	1
train/s	73	8

Variierbare Parameter

Dimensionen von Word Space

Local Strategy

- konzentriert sich auf den Kontext des ambigen Wortes und ignoriert den Rest des Korpus
- **Welche Wörter werden als Dimensionen ausgewählt?**
 - die n häufigsten Wörter (Frequency)
 - die Wörter, deren Auftreten vom Vorkommen des ambigen Wortes abhängt, getestet durch χ^2 -Test

Global Strategy

- konzentriert sich auf alle Wörter im Korpus, unabhängig vom ambigen Wort
- **Welche Wörter werden als Dimensionen ausgewählt?**
 - die n häufigsten Wörter (Frequency)

Variierbare Parameter

Dimensionen von Word Space

- **Term representation**

→ alle ausgewählten Dimensionen werden als Dimensionen beibehalten

- **SVD representation**

→ SVD = singular value decomposition

→ Dimensionen werden reduziert, Verhältnis der Vektoren zueinander ist im reduzierten Vektorraum dasselbe wie im ursprünglichen Vektorraum

Hier: bei Anwendung von SVD wird auf 100 Dimensionen reduziert

Variierbare Parameter

Clustering

- **2 cluster**

→ „grobes Clustering“

→ die zwei groben
Bedeutungsrichtungen des ambigen
Wortes

- **10 cluster**

→ „feines Clustering“

→ feinere Kontextunterschiede können
festgehalten werden

→ feinere
Wortbedeutungsunterscheidungen
(wie z.B. in Wörterbüchern)

→schwerer die Bedeutungen von
Vorkommen zu bestimmen,
daher: two-way discrimination

Vergleich der Parameter

	Local								Global	
	χ^2				Frequency				Frequency	
	Terms		SVD		Terms		SVD		SVD	
	2	10	2	10	2	10	2	10	2	10
Average accuracy	72,1	77,9	84,1	88,5	77,8	81,8	82,9	88,3	89,7	90,6

- Local, χ^2 , terms < local, frequency, terms < local, frequency, SVD < local, χ^2 , SVD < global, frequency, SVD
- 10 Cluster besser als 2 Cluster

4. Information Retrieval

Anwendung im Information Retrieval

- → Anwendung der Word Sense Disambiguation im Information Retrieval ist die Hauptmotivation für die Konzentration auf die Teilkomponente CGD
- Obwohl viele Experimente dagegen sprechen → CGD verbessert die Anwendung des Information Retrieval
- Experiment in Schütze and Pedersen (1995), das dafür spricht, dass CGD nützlich für Information Retrieval ist:
 - testet sense-based retrieval, eine Modifizierung des Standardvektorraummodells (word-based retrieval)
 - Dokumente und Anfragen als Vektoren dargestellt in Vektorraum mit Bedeutungen (anstatt Wörtern) als Dimensionen
 - CGD wird auf Wörter angewandt und die Query-Vektoren haben an den Stellen mit den Wörtern zugeordneten Bedeutungen einen Wert ungleich 0
 - sense-based Retrieval verbesserte die Qualität der Treffer um 7,4% verglichen mit word-based Retrieval
 - bei Kombination von sense-based und word-based Retrieval sogar Verbesserung um 14,4%

Sense-based Retrieval vs. Term Expansion

- Term Expansion weitet Suche auf semantisch ähnliche Begriffe aus
- Beide Anwendungen scheinen ähnlich zu sein: sie verändern beide den Maßstab, welche Dokumente als Treffer erachtet werden und welche nicht
- Eigentlich sind beide aber genau gegensätzlich:
→ Term Expansion erhöht die Anzahl der Treffer, während Sense-based Retrieval die Anzahl erniedrigt

- **Beispiel:**

- **(Term Expansion)**

- Anfrage beinhaltet „cosmonaut“:

- Term Expansion weitet Suche auf „astronaut“ aus, Vektoren für Dokumente, die „astronaut“ beinhalten, bekommen für die Stelle „cosmonaut“ auch Werte ungleich 0

- **(Sense-based Retrieval)**

- Anfrage beinhaltet „suit“:

- „suit“ wird hier als Bedeutung mit Kontext „legal“ klassifiziert; Dokumente, die „suit“ mit anderer Bedeutung enthalten, werden nicht mehr als Treffer gezählt

5. Begründung und Probleme des Algorithmus

Grundidee der CGD – Warum Bedeutungen aus ähnlichen Kontexten herleiten?

- Zwei Gründe:
 - Die aus CGD resultierenden Bedeutungsgruppen stimmen gut mit den Grundbedeutungen, welche in Wörterbüchern definiert werden, überein.
 - Auch bei der semantischen Bedeutungsunterscheidung beim Menschen spielt die Klassifizierung von ähnlichen Kontexten eine entscheidende Rolle

Strong Contextual Hypothesis: Two words are semantically similar to the extent that their contextual representations are similar. (Miller and Charles (1991), p.8)

- Hypothese, die CGD begründet, ist eine Erweiterung der Strong Contextual Hypothesis:

Contextual Hypothesis for Senses: Two occurrences of an ambiguous word belong to the same sense to the extent that their contextual representations are similar.

Noch offene Probleme

- Experimente haben sich nur auf Wörter mit zwei Hauptbedeutungen konzentriert
 - Algorithmus muss auch für Wörter mit mehr als zwei Hauptbedeutungen bzw. mehreren selteneren Bedeutungen getestet werden
- Experimente beinhalteten nur relativ wenige natürliche ambige Wörter
 - Algorithmus muss mit größeren Testsets geprüft werden
- Implementierung der CGD basiert nur auf thematischer Ähnlichkeit, Probleme bei Wörtern, die wenig thematisch aussagen („to train“, „wide range“)
 - andere, strukturellere Informationen müssen miteinbezogen werden
 - Beispiel: „interest *in*“ – „interest *to*“

Quellenangabe

- Schütze, Hinrich. (1998): Automatic Word Sense Discrimination. Computational Linguistics 24. S.97-123
- www.wikipedia.de