

Künstliche neuronale Netze und Computerlinguistik

Ansätze zur Simulation
der natürlichen Sprachverarbeitung

ein Vortrag von
Martin Pyka

1. Einführung in neuronale Netze

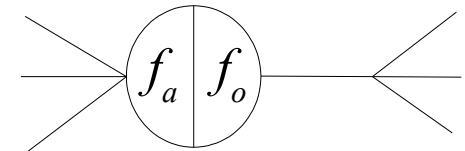
- Aufbau und Funktionsweise
- Klassen neuronaler Netze
- Lernverfahren

2. Ansätze zur Sprachverarbeitung mit neuronalen Netzen

- Das Past-Tense-Modell
- Sentence Processing
- Paralleles Parsing
- Sequentielle Netze

Elementare Bestandteile:

- Neuronen
- Verbindungen zwischen Neuronen

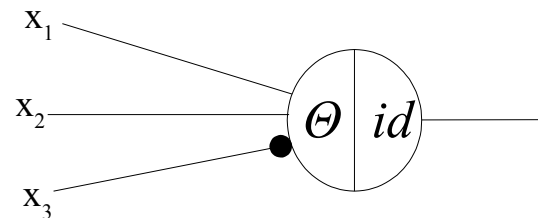


Neuron besteht aus:

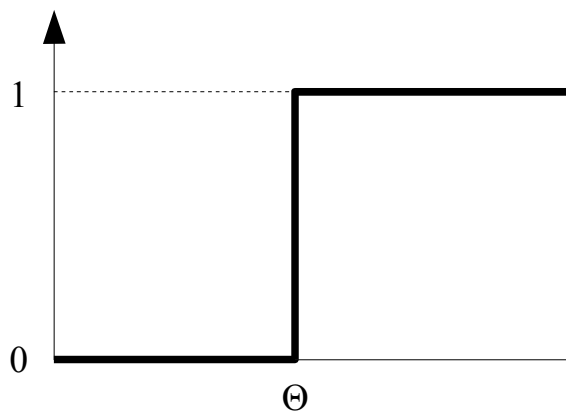
- Gewichteten Eingaben
- Aktivierungsfunktion
- Ausgabefunktion
- Optionalen inneren Zustand

Beispiel: Mc Culloch-Pitts-Zelle

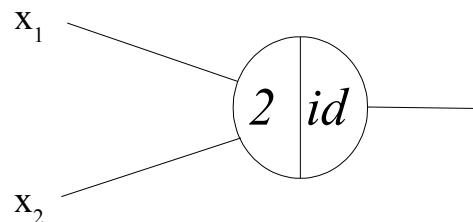
- Aktivierungsfunktion ist die Treppenfunktion mit einem Schwellenwert
- Ausgabefunktion ist die Identität
- Es gibt erregende und hemmende Zellen



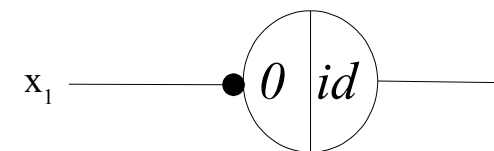
Schwellenwertfunktion



AND

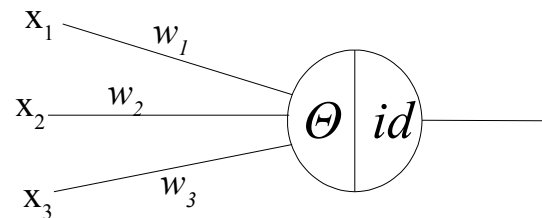


NOT

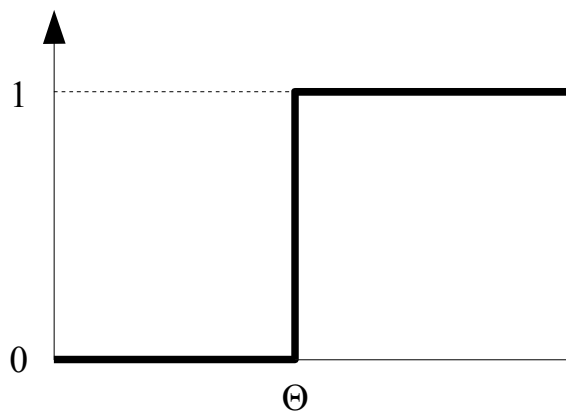


Beispiel: Das Perzeptron-Neuron

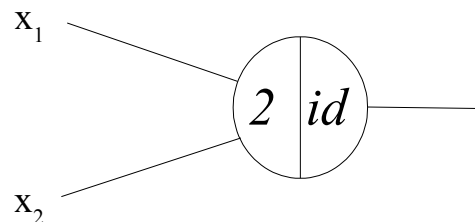
- Gewichtete Verbindungen
- Positive oder negative Gewichtung bestimmt Grad der Erregung oder Hemmung



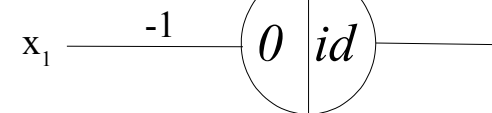
Schwellenwertfunktion



AND



NOT

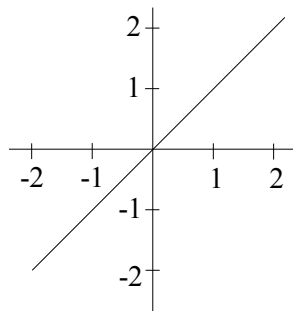


Beispiel: Der allgemeine Fall

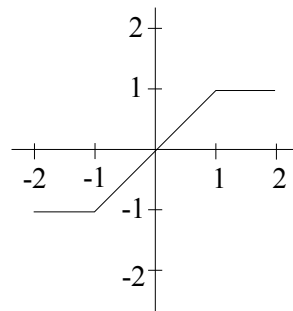
- gewichtete Summe als Aktivierungsfunktion
- Anforderungsabhängige Ausgabefunktion

Verschiedene Ausgabefunktionen

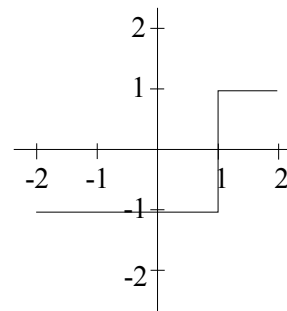
Lineare Funktion



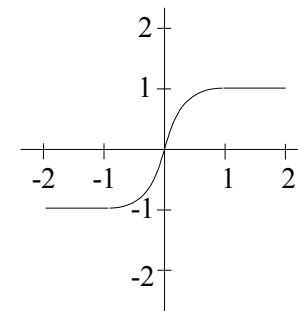
Linear bis Sättigung



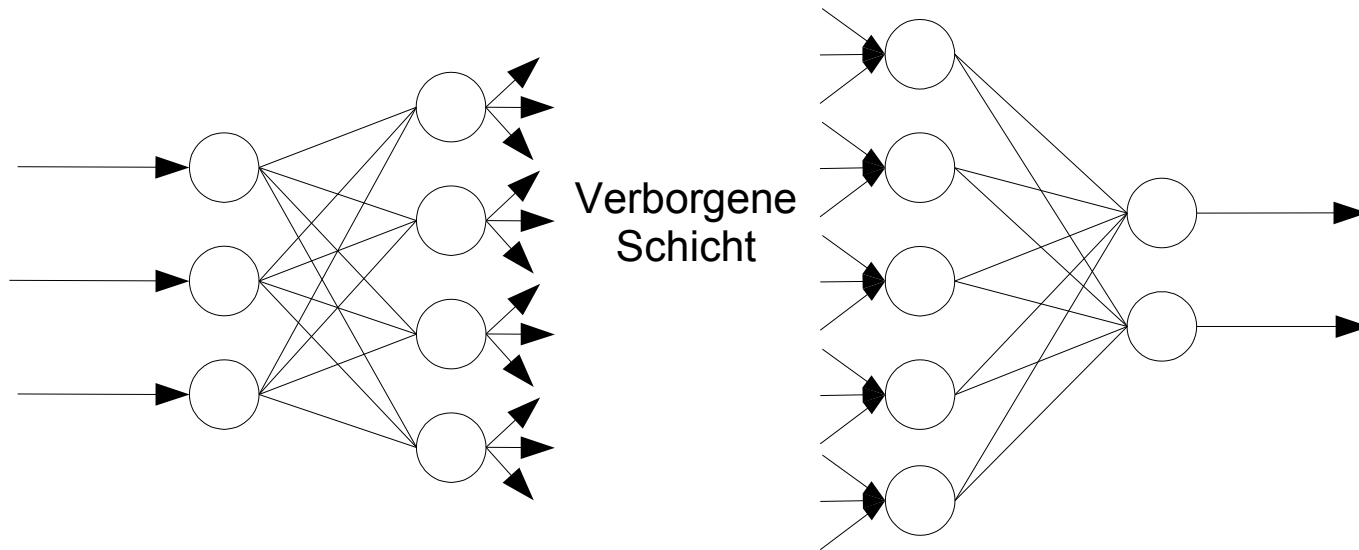
Binäre
Schwellenwertfunktion



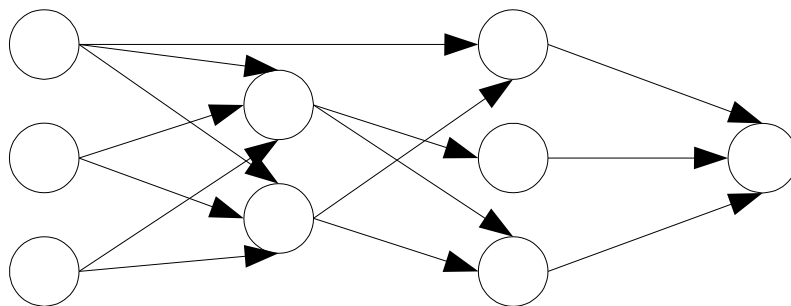
sin(x) bis Sättigung



Feedforward-Netze

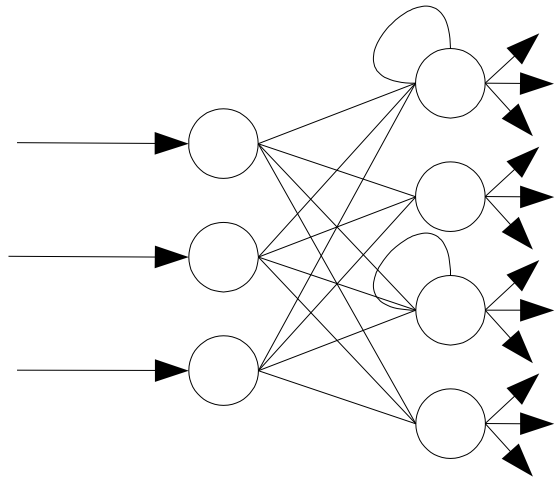


Feedforward-Netze mit Shortcut

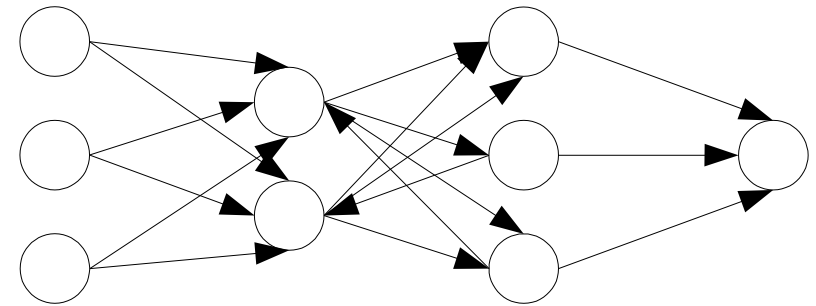


Vernetzungsstruktur mit Rückkopplung

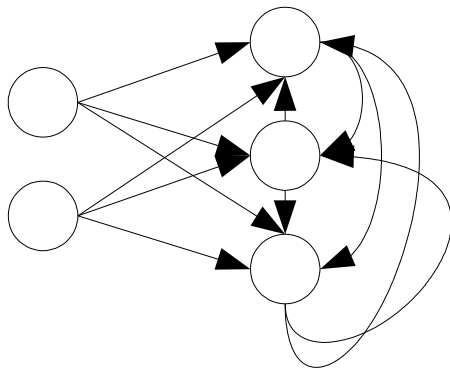
Direct Feedback



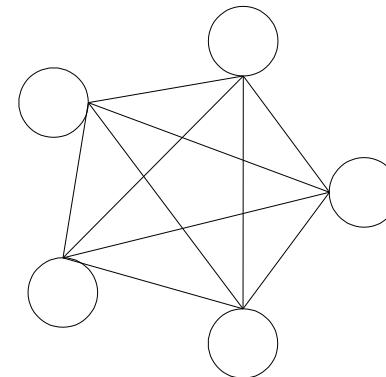
Indirect Feedback



Lateral Feedback (winner takes all)



Vollständig verbundene Netze



Typen des Lernens

- Entwicklung neuer Verbindungen (Topologie-Aspekt)
- Modifikation der Stärken der Verbindungen
- Modifikation des Schwellenwertes
- Modifikation der Aktivierungs- bzw. Ausgabefunktion
- Entwicklung neuer Zellen

Arten des Lernens:

Überwachtes Lernen (*supervised learning*)

- Tatsächliche Ausgabe des Netzes wird mit gewünschter verglichen
- Je nach Abweichung werden Änderungen an den Gewichten vorgenommen
- Setzt Existenz von Trainingsdaten voraus

Arten des Lernens:

Bestärkendes Lernen (*reinforcement learning*)

- Netz bekommt lediglich mitgeteilt, ob die Ausgabe korrekt oder inkorrekt ist
- Der exakte Wert der Abweichung ist dem Netz nicht bekannt
- Für nicht binäre Probleme definiert man einen Toleranzrahmen

Arten des Lernens:

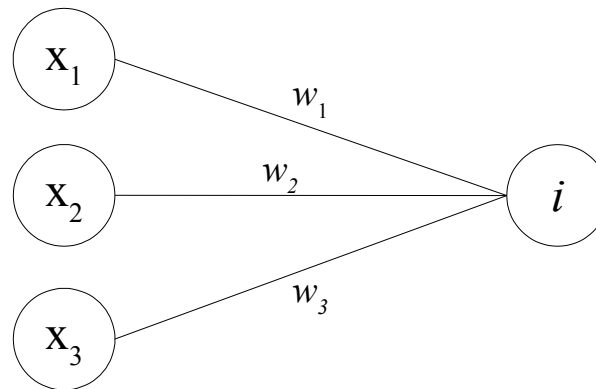
Unüberwachtes Lernen (*unsupervised learning*)

- Netz versucht Eingaben in Ähnlichkeitsklassen aufzuteilen
- Kein Einfluss von Aussen durch korrigierende Veränderungen

Beispiel für Überwachtes Lernen – Die Delta Regel

$$\vec{w}_i^{neu} = \vec{w}_i^{alt} + \eta \cdot \frac{1}{n} \cdot \sum_{k=1}^n \delta_i^{(k)} \cdot \vec{x}^{(k)}$$

$$\delta = |t_i^{(k)} - o_i^{(k)}|$$



Anwendungsbereiche

- Bilderkennung
- Spracherkennung (basierend auf Audio-Eingaben)
- Erforschung wirtschaftlicher Zusammenhänge
- Prognoseprobleme (in der Wirtschaft)
- Steuerung von Prozessen (Bsp.: Flugzeug oder Autosteuerung)

- **Ansätze zur Sprachverarbeitung mit neuronalen Netzen**
 - Das Past-Tense-Modell
 - Sentence Processing
 - Paralleles Parsing
 - Sequentielle Netze

Beschreibung

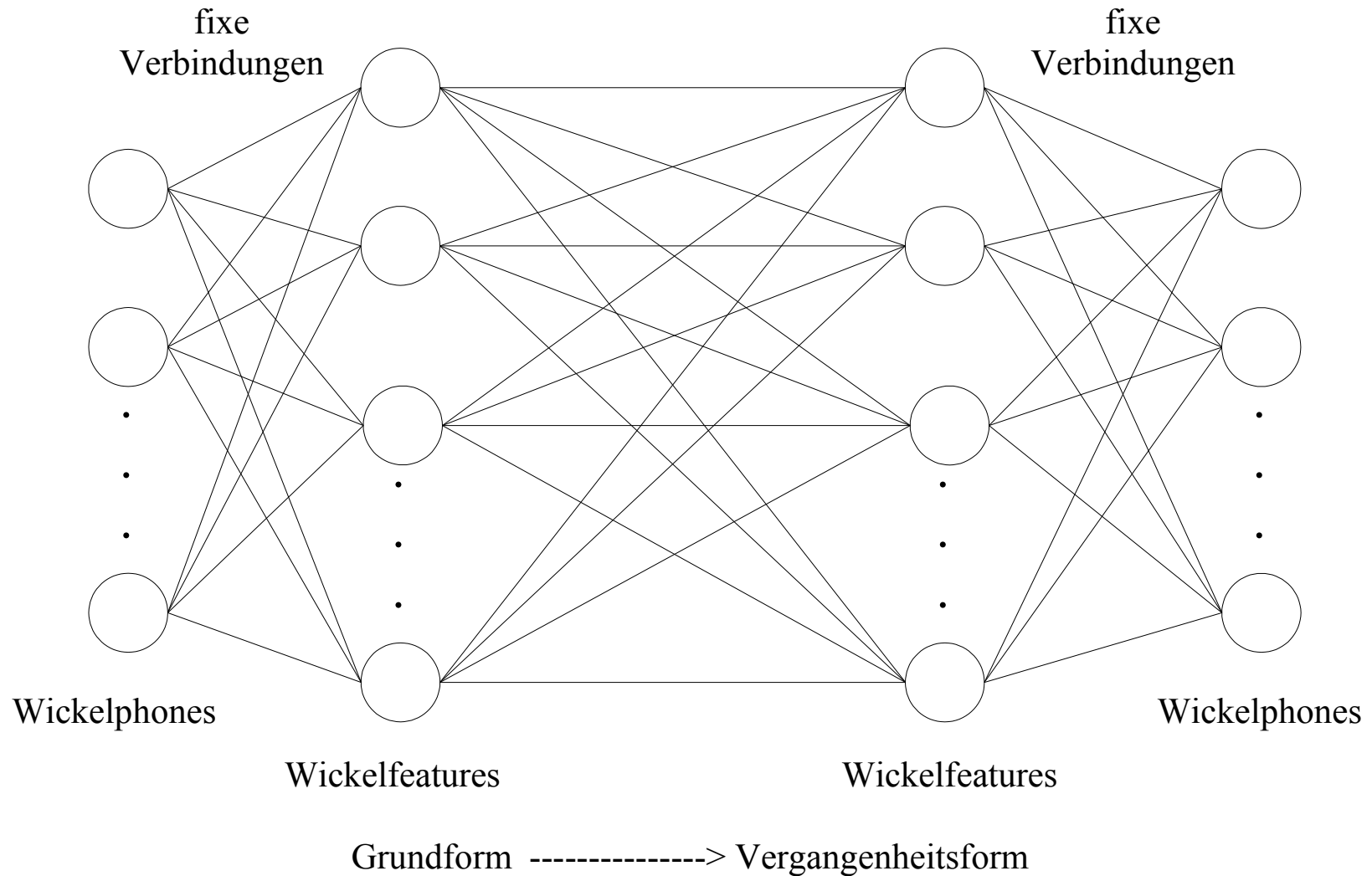
- Ausgangspunkt ist eine Beobachtung im natürlichen Spracherwerbsprozess
 - Kinder bilden beim Lernen des Past Tense gelegentlich falsche Verbformen (*goed, camed*)
 - Folge von Übergeneralisierung einer gelernten Regel
 - Im Lernprozess müssen Regeln und deren Ausnahmen begriffen werden

entwickelt von Rumelhart/McClelland (1986)

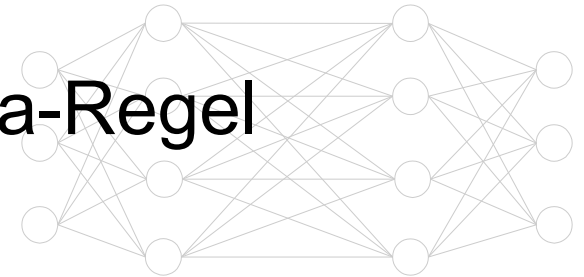
Beschreibung

- Ein künstliches neuronales Netzwerk soll zeigen, wie ein solcher Erwerbsprozess ohne Rückgriff auf explizite Regeln oder diskrete Symbolstrukturen erfolgt
 - Verben werden in Form von *Wickelphones* repräsentiert
 - Jedes Wickelphone besteht aus 33 Einheiten
 - zur Reduzierung der Neuronen werden Wickelphones auf Wickelfeatures reduziert

Netzwerkarchitektur des Past Tense Modells



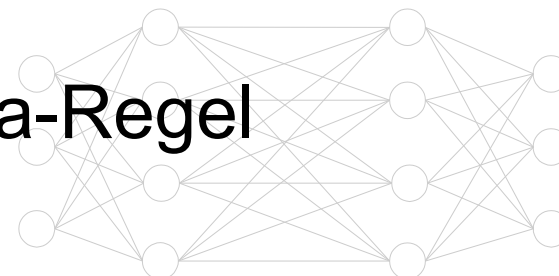
Das Adaptionungsverfahren durch die Delta-Regel



1. Präsentation von Stammformen

- 10 hochfrequente Verben werden antrainiert (Kinder lernen zunächst nur wenige hochfrequente Verben mit zumeist unregelmässigen Past-Tense-Formen)
 - das Netz adaptiert die Past-Tense Bildung korrekt
 - eine *Bildungsregel* ist nicht erkennbar (da es sich um Vollformen handelt)

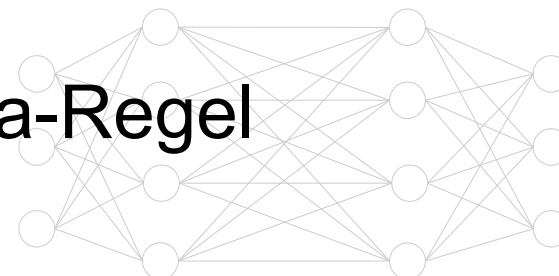
Das Adaptionungsverfahren durch die Delta-Regel



2. Wortschatz wächst an

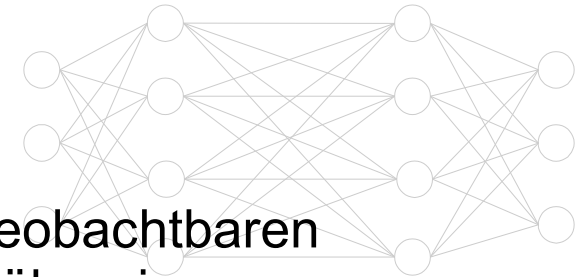
- 420 mittelfrequente Verben werden dem Netz antrainiert
- die meisten Past-Tense Formen sind regelmässig
 - Es bildet sich eine Tendenz zur Regularisierung
 - es kommt zur Übergeneralisierung und damit zur Bildung von Formen, wie *go+ed*
- gleiches Phänomen bei Kindern

Das Adaptionungsverfahren durch die Delta-Regel



3. Wortschatz wächst an weiter an

- alter Wortschatz wurde weiter trainiert
 - die in Phase 1 erlernten Verbformen wurden wieder korrekt gebildet
 - die Bildung neuer Verbformen wurde adaptiert
- 86 neue niederfrequente Verben wurden eingegeben
 - 92% aller regelmässigen Past-Tense-Formen wurden korrekt gebildet
 - 84% aller unregelmässigen Past-Tense wurden korrekt gebildet



Schlussfolgerungen

- im groben stimmt das Past-Tense-Modell mit beobachtbaren empirischen Phänomenen des Spracherwerbs überein
- der Erwerb ist ohne Zuhilfenahme expliziter Regeln durchführbar

Mögliche Einwände gegen das Modell

- es fehlt semantische Informationen beim Lernprozess
- die Repräsentation in Form von Wickelfeatures stellt eine weitgehende Beeinflussung des Lernprozesses dar

Beschreibung

- im Mittelpunkt steht die Zuordnung von Konstituenten eines Satzes zu thematischen Rollen
- diese Zuordnung unterliegt verschiedenen Kriterien bzw. Beschränkungen (*constraints*)
- das Netz soll die Rollen der Wörter anhand der kontextuellen Informationen bestimmen und Verbrahmen mit semantisch spezifizierten Argumentmustern ausfüllen
- bei unbekanntem Wörtern soll durch die Generalisierungsfähigkeit des Netzes eine entsprechend wahrscheinliche Zuordnung stattfinden

entwickelt von McClelland/Kawamoto (1986)

Beispielsatz

The boy broke the window.

The rock broke the window.

→ *The boy* hat die Rolle des Agens

→ *The rock* ist das Instrument.

Arbeitsweise

- dem Netz können folgende Oberflächenstrukturen übergeben werden:
 - Subjekt Prädikat
 - Subjekt Prädikat Objekt
 - Subjekt Prädikat Objekt Präpositionalphrase
- Jedes Wort wird durch einen binären Merkmalsvektor codiert (sogenannte Microfeatures)
 - für Nomen: HUMAN, SOFTNESS, GENDER etc.
 - für Verben: DOER, CAUSE, TOUCH
 - Ein Microfeature kann die Werte 1, ? oder 0 haben

Arbeitsweise

- das Netz soll als Ausgabe die Kasusstruktur des Satzes wiedergeben

Beispiel:

The boy broke the window with the hammer.

Oberflächenstruktur:

Subjekt Prädikat Objekt Präpositionalphrase

Kasusstruktur:

- Broke Agent Boy
- Broke Patient Window
- Broke Instrument Hammer

Arbeitsweise

- Innerhalb der Trainingsphase wird ein Korpus an Sätzen generiert
- Das Netz wird durch überwachtes Lernen mittels der Delta-Regel trainiert
- Es wird eine probabilistische Aktivierungsfunktion und ein modifizierbarer Schwellenwert benutzt

Nach 50 Trainingszyklen hatte das Netz eine Entscheidungssicherheit von 98,5%

The dog broke the plate. *dog* ist Agens, *plate* ist Patiens

The hammer broke the vase. *hammer* ist Instrument

The plate broke. *plate* ist Patient.

→ Semantik geht vor Satzstellung

Sind beide Argumente eines Verbrahmens semantisch gleichwertig, bestimmt die Satzstellung die Zuweisung der jeweiligen Rolle

The boy hit the girl

The girl hit the boy

Die erste NP wird als Agens interpretiert

Bei unvollständigen Eingabesätzen zeigte das Modell die Tendenz, fehlende Rollen zu ergänzen

The boy broke.

Die ausgebenene Kasusstruktur zeigte ein Objekt mit den Microfeatures NONHUMAN, NEUTER, FRAGILE

The man moved.

man kann sowohl Agens als auch Patiens sein.

Auflösung von Ambiguitäten

The chicken ate the carrot

The man ate the chicken with the fork.

Das Netz erkennt im ersten Satz *chicken* als lebendes Objekt und im zweiten Satz als gekochtes Objekt an.

Schwächen im System:

The lion ate the chicken. 1)

The wolf ate the chicken. 2)

In 1) wird *chicken* als *living-chicken* in 2) jedoch als *cooked-chicken* gesehen (Generalisierung, da *ate* öfter mit *cooked-chicken* in Verbindung gebracht wird)

Auflösung von Ambiguitäten

The bat broke the window.

The bat wird als Agens und als Instrument gesehen was mit der semantischen Mehrdeutigkeit, dem Fehlen an Kontextwissen und den vorhandenen Trainingsdaten zusammenhängt

The ball broke the vase

Ball wird durch das Microfeature *hard* codiert, obwohl es eigentlich durch *soft* beschrieben wurde.

Grund:

Alle Instrumente im Kontext „brechen“ besitzen das Merkmal *hard*

Auflösung von Ambiguitäten (*Creative Errors*)

The doll moved.

Es wurden die Rollen *Agens* und *Patiens* aktiviert.

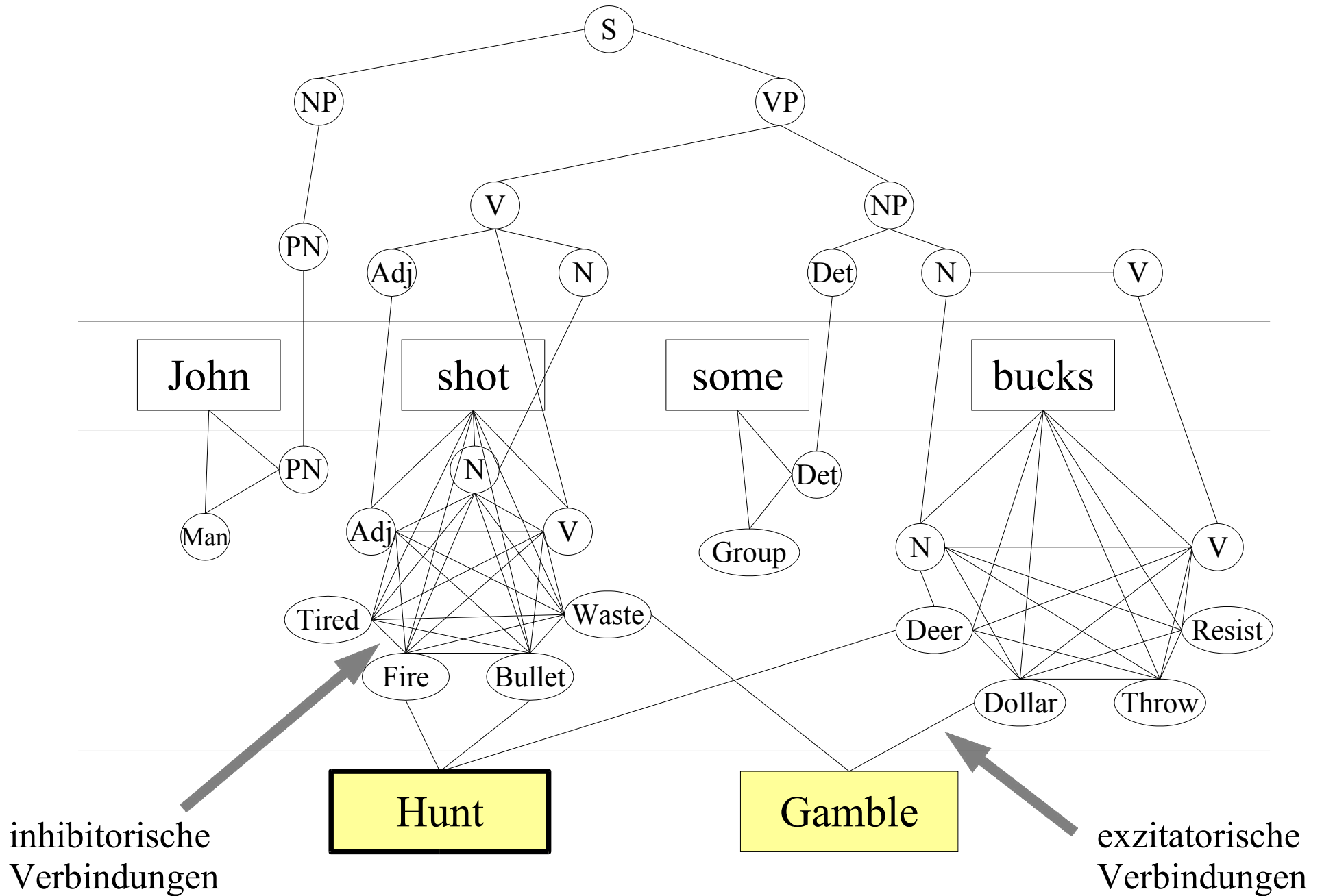
Grund:

Das Auftreten von *moved* setzte im Trainingskorpus immer ein belebte Agens voraus.

Unbelebte Objekte (wie *doll*) hatten immer die Rolle des *Patiens*.

Ansätze zur Sprachverarbeitung mit NN

Paralleles Parsing



Arbeitsweise

- durch inhibitorische und exzitatorische Verbindungen wird die Bedeutung anhand des Kontextes und der syntaktischen Struktur analysiert
- Kontext wird durch Microfeatures der Wörter assoziiert. Dies setzt die Annahme voraus, dass bestimmte Schlüsselwörtern in einem Kontext auftreten.

Beispiel: Hunting

starke Assoziation: Forrest, Outside

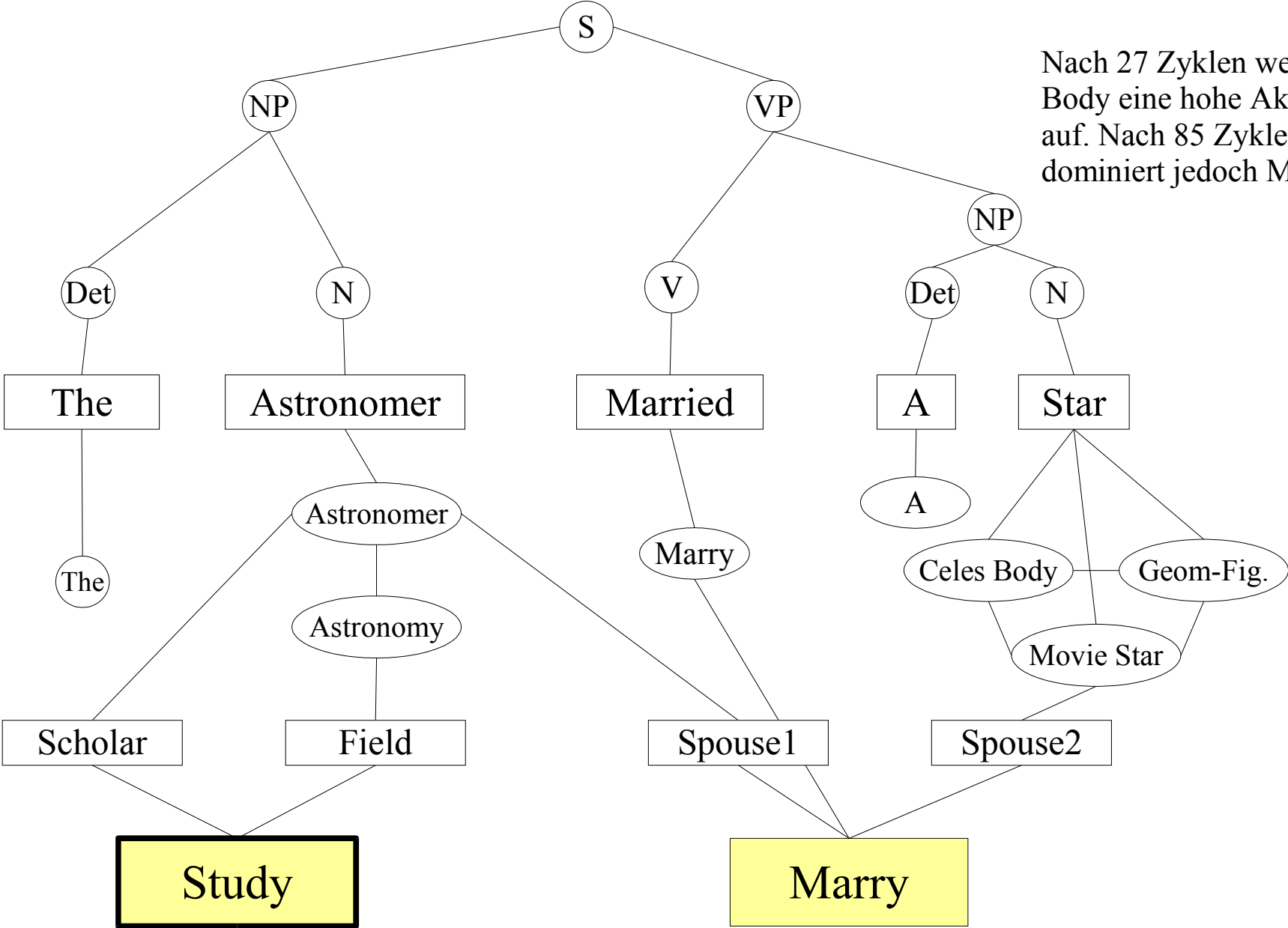
schwache Assoziation: Lake, Desert, Mountain

keine Assoziation: City Park, Inside

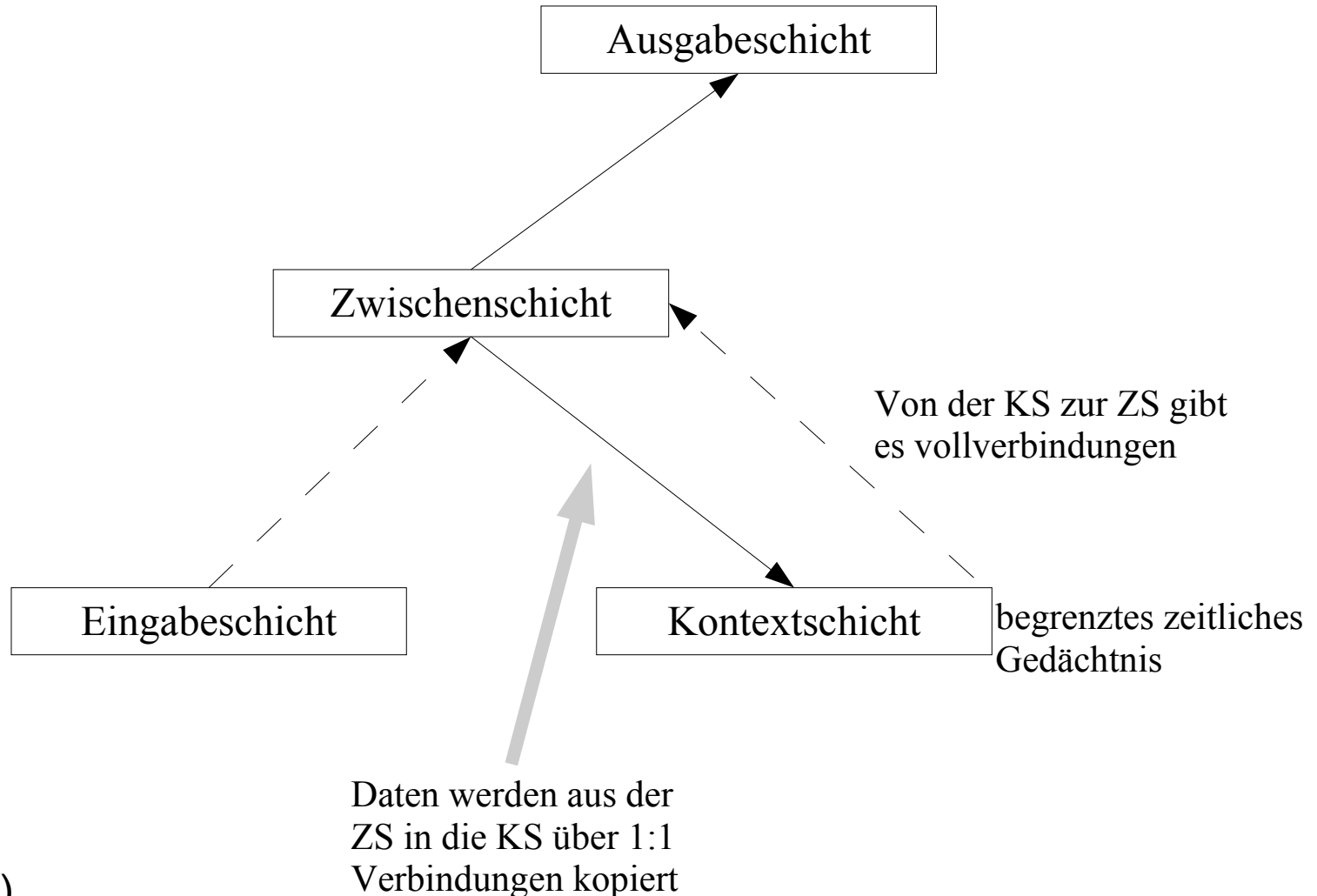
- Auswahl ist willkürlich, erhebt keinen Anspruch auf Vollständigkeit (Systemabhängig)

Ansätze zur Sprachverarbeitung mit NN

Paralleles Parsing



Aufbau des Netzes



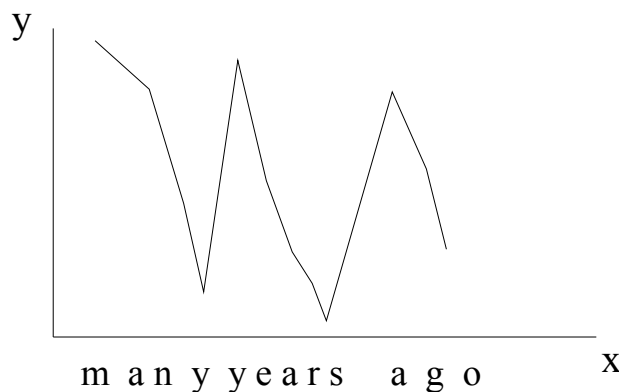
Elman (1990)

Beschreibung

- es werden dem Netz Buchstaben als Sequenzen von 5-Bit-Vektoren ohne trennende Elemente eingegeben
- 5 Elemente in der Eingabe- und Ausgabeschicht sowie 20 Elemente in der Zwischen- und Kontextschicht
- Das Netz wurde aus einem mit 15 Wörtern bestehenden Lexikon mit 200 Sätzen unterschiedlicher Länge trainiert (keine Trennzeichen)
- das Netz wurde darauf trainiert, den nächsten Buchstaben vorherzusagen

Ergebnis

- Unsicherheit des Systems bei der Vorhersage des nächsten Buchstaben nimmt mit jedem weiteren Buchstaben eines Wortes ab



- Das Ergebnis drückt keine kategorialen Zustände aus, sondern reflektiert das graduelle Auftreten der einzelnen Buchstaben in einer Folge
- Informationen lassen sich aus einer Eingabekette als auch ohne Zuhilfenahme von Kategorien extrahieren

Weiteres Experiment

- Annahme: die Satzoberfläche ist der einzige für den Hörer zugängliche Input. Die Struktur der Eingabekette ist implizit in dieser Oberfläche enthalten. (generative Grammatik setzt eine zweite Strukturebene voraus)
- es wird ein Korpus von Trainingssätzen aus 29 lexikalischen Einheiten generiert (z.B.: *man, woman, cat, see, eat, etc.*), die unter Kategorien, wie NOUN-HUMAN, NOUN-ANIM, VERB-INTRAN oder VERB-EAT eingeordnet sind.
- Satzgenerator erzeugt aus vorgegebenen Mustern 10 000 Zwei- und Drei-Wort-Sätze

Beispiel-Muster:

NOUN-HUM

VERB-EAT

NOUN-FOOD

man

eat

cookie

Weiteres Experiment

- jedes Wort wird durch einen einzelnen 31-Bit-Vektor repräsentiert
- alle Sätze werden zu einer Kette mit 27354 31-Bit-Vektoren zusammengesetzt
- Eingabe- und Ausgabeschicht: 31 Einheiten
Zwischen- und Kontextschicht: 150 Einheiten
- das Netz wird darauf trainiert, das nächste Wort vorher zu sagen (über die verallgemeinerte Delta-Regel)

Ergebnis

- Das System war in der Lage interne Repräsentationen über die einzelnen Wortklassen und deren Abfolge zu entwickeln
- die kategoriale Struktur ist dabei nur implizit in der Struktur der Zwischenschicht enthalten und musste über eine Clusteranalyse nachgewiesen werden
- es wurde eine Gruppierung der Neuronen sichtbar, bei der z.B. alle Nomen, Verben oder Elemente, die als ANIMATE zu klassifizieren sind, beieinander liegen.
- die kategorialen Informationen waren nicht expliziter Bestandteil des Trainingskorpus, sondern implizit durch die automatische Generierung des Korpus vorgegeben.