

ENDLICHE GEWICHTETE TRANSDUKTOREN ALS SEMANTISCHER TRÄGER

Christian Kölbl¹, Markus Huber¹, Günther Wirsching²

*¹Universität Augsburg, ²Universität Eichstätt-Ingolstadt
vorname.nachname@{uni-augsburg|ku-eichstaett.de}*

Kurzfassung: Endliche gewichtete Transduktoren werden im Bereich der Computerlinguistik bereits bei Erkennung und Synthese eingesetzt. Wir stellen hier die Verwendung von Transduktoren als semantische Träger in Dialogsystemen vor. Sowohl Erkennungsergebnisse, als auch der Informationsstatus, welcher durch jene ständig aktualisiert wird und den “Wissensstand” des Systems repräsentiert, werden als Transduktoren modelliert. Dabei entspricht die Kantenbeschriftung des Transduktors der Bedeutung des Erkennungsergebnisses und die Kantengewichte repräsentieren Wahrscheinlichkeit und Konfidenz der Bedeutungen. Die Aktualisierung erfolgt mit geeigneten Operationen auf einem Konfidenz-Wahrscheinlichkeitshalbring. Weiter erfüllen gewichtete Transduktoren als Semantikrepräsentation die Eigenschaften der Verifizierbarkeit, Eindeutigkeit, Kanonische Form und Inferenz.

1 Einleitung

In einem aktuellen Forschungsprojekt beschäftigen wir uns seit 2 Jahren mit Semantikrepräsentation im Rahmen des Entwurfes eines statistischen hierarchischen domainspezifischen Dialogsystems. In [1] haben wir die Übersetzung von relationalen Datenbanken in Merkmal-Werte-Relationen (MWRen) beschrieben. MWRen erfüllten bisher in unserem Entwurf die Rolle des semantischen Trägers. In dieser Publikation soll gezeigt werden, wie gewichtete Transduktoren (im Folgenden FST, für Finite State Transducer, genannt) als semantische Träger - szsg. als Implementierung von MWRen - genutzt werden können, wodurch eine einheitliche Datenstruktur zwischen Erkennung, Synthese und Semantik geschaffen werden soll. In [6] haben wir einen dafür geeigneten Konfidenz-Wahrscheinlichkeits Halbring definiert. Außerdem werden wir die in [2] erwähnten notwendigen Eigenschaften zur Bedeutungsrepräsentation nachweisen und zeigen wie man mit bekannten FST-Operationen (vgl. [3]) eine Wissensbasis durch Erkennungsergebnisse erweitern und aktualisieren kann, wenn beides als FST dargestellt wird.

Im nächsten Kapitel werden wir die für dieses Paper notwendigen Definitionen aufführen und im Anschluss daran unseren semantischen Transduktor beschreiben. Im vierten Abschnitt wird erklärt, wie ein semantischer Transduktor mit einem anderen aktualisiert werden kann und schließlich im vorletzten Kapitel, warum die semantischen Transduktoren die genannten Eigenschaften aus der Kurzfassung erfüllen.

2 Definitionen und Operationen

Soweit nicht anders markiert beziehen sich die genannten Definitionen auf [3].

Definition 1 (Halbring) *Ein System $(H, \oplus, \otimes, 0, 1)$ ist ein Halbring, wenn $(H, \oplus, 0)$ ein kommutatives Monoid mit 0 als neutralem Element und $(H, \otimes, 1)$ ein Monoid mit neutralem Element*

1 ist. Außerdem muss das Distributivgesetz für beliebige $a, b, c \in H$ gelten:

$$(a \oplus b) \otimes c = a \otimes b \oplus b \otimes c \text{ und } c \otimes (a \oplus b) = c \otimes a \oplus c \otimes b,$$

und 0 muss bezüglich der Multiplikation absorbierend sein: $0 \otimes a = a \otimes 0 = 0$.

Definition 2 (Transduktor) Ein gewichteter Transduktor T über einem Halbring $(S, \oplus, \otimes, 0, 1)$ ist ein 6-Tuple $T = (\Sigma, \Delta, Q, I, F, E)$ ¹:

Σ endliches Eingabealphabet, Δ endliches Ausgabealphabet, Q die endliche Menge aller Zustände, $I \subseteq Q$ die Menge aller Startzustände, $F \subseteq Q$ die Menge aller Endzustände und E eine endliche Multimenge der Übergänge aus $Q \times (\Sigma \cup \{\varepsilon\}) \times (\Delta \cup \{\varepsilon\}) \times S \times Q$.

- Für $q \in Q$ bezeichnet $E[q]$ alle von q ausgehenden Kanten und mit $E[Q']$ alle ausgehenden Kanten aller Zustände q der Teilmenge $Q' \subseteq Q$
- Eine ε -Kante ist eine Kante, bei der sowohl Eingabe- wie auch Ausgabebeschriftung gleich ε sind.
- Ein Pfad π ist ein Element aus E^* mit aufeinanderfolgenden Kanten. Das Gewicht eines Pfades ist das sukzessive \otimes -Aufmultiplizieren der Kantengewichte und wird mit $w[\pi]$ notiert.
- $P(Q_1, Q_2)$ ist die Menge aller Pfade von $Q_1 \subseteq Q$ nach $Q_2 \subseteq Q$, $P(Q_1, x, Q_2)$ sind alle Pfade aus $P(Q_1, Q_2)$ mit der Eingabe x , $P(Q_1, x, y, Q_2)$ sind alle Pfade aus $P(Q_1, x, Q_2)$ mit der Ausgabe y .
- Ein Pfad in $P(I, F)$ heisst *akzeptierend*.
- Wir definieren außerdem $T(x, y) = \bigoplus_{\pi \in P(I, x, y, F)} w[\pi]$.

Definition 3 (Vereinigung) Die Vereinigung berechnet die Summe zweier Transduktoren:

$$\forall (x, y) \in \Sigma^* \times \Delta^*, (T_1 \oplus T_2)(x, y) = T_1(x, y) \oplus T_2(x, y).$$

Definition 4 (Determinierung) Diese Operation konstruiert zu einem nicht deterministischen Transduktor einen äquivalenten deterministischen Transduktor². Ein Transduktor ist deterministisch, wenn es nur einen Startzustand gibt, kein Zustand existiert mit mehreren ausgehenden Kanten mit gleichen Eingabebuchstaben, wobei ε -Übergänge als reguläre Übergänge betrachtet werden. Jeder Pfad ist also eindeutig bestimmt. Hierfür muss der im Transduktor verwendete Halbring schwach links dividierbar sowie 0-Summen frei sein.

Definition 5 (Minimierung) Diese Operation minimiert einen deterministischen gewichteten Transduktor in der Anzahl der Zustände äquivalent zum Ausgangstransduktor. Zwei Zustände eines deterministischen gewichteten Automaten³ sind äquivalent, wenn genau dieselbe Menge an Zeichenketten, ausgehend von diesen Zuständen mit den gleichen Gewichten zu einem Endzustand führen.

¹Mohri beschreibt in [3] den Transduktor als 8-Tupel. Da wir weder die initialen noch die finalen Gewichtungs-funktionen benötigen, verzichten wir in unserer Definition darauf

²Nicht jeder Transduktor ist determinierbar. Darüber hat Christian Choffrut (ein französischer Informatiker) 1977/78 seine Dissertation geschrieben und die "twins property" als charakterisierend für die Determinierbarkeit erkannt. Mohri erwähnt das in einem Paper von 1997, wo er dies auf gewichtete Transduktoren verallgemeinert.

³Später verwenden wir bei den Transduktoren gleiche Ein- und Ausgabelabels, daher genügt diese Einschränkung

Person			
ID	Vorname	Nachname	Tel.
1	Peter	Maier	1111
2	Peter	Müller	2222
3	Paul	Maier	3333
4	Paul	Müller	4444

bewohnt	
Pers.ID	Adr.ID
1	2
2	1
3	4
4	3

Adresse		
ID	Straße	Nr.
1	Arlstraße	2
2	Bergerstraße	1
3	Caesarstraße	4
4	Denning	3

Tabelle 1: Andeutung einer relationalen Datenbank

Definition 6 (ε -Kanteneliminierung) Diese Operation entfernt alle ε -Kanten eines Transduktors. Das Ergebnis ist ein äquivalenter Transduktor ohne ε -Kanten.

Definition 7 (Komposition) Diese Operation berechnet die Komposition zweier Transduktoren.

$$T_1 \circ T_2(x, y) = \bigoplus_{z \in \Delta^*} T_1(x, z) \otimes T_2(z, y).$$

Wenn T_1 einen String x nach z mit Gewicht a übersetzt, und T_2 einen String z nach y mit Gewicht b , dann übersetzt die Komposition von T_1 und T_2 den String x nach y mit dem Gewicht $a \otimes b$.

Definition 8 (Konkatenation) Diese Operation berechnet das Produkt zweier Transduktoren T_1, T_2

$$\forall (x, y) \in \Sigma^* \times \Delta^*, (T_1 \otimes T_2)(x, y) = \bigoplus_{\substack{x=x_1x_2 \\ y=y_1y_2}} T_1(x_1, y_1) \otimes T_2(x_2, y_2).$$

Für einen Summanden gilt also: Wenn T_1 x mit Gewicht a in y und T_2 w mit b in v überführt, so überführt die Konkatenation von T_1 und T_2 xw mit $a \otimes b$ in yv .

3 Aufbau eines semantischen Transduktors

Wir wollen in dieser Arbeit die Anwendung von Transduktoren als semantische Träger beschreiben und definieren. Entgegen der bisherigen Verwendungen nutzen wir die Transduktoren nicht zur Überprüfung von Eingabewörtern (Generatoren, Übersetzer) oder der Ermittlung von Pfadkosten (Sequenzklassifikatoren). Für uns ist im Folgenden ausschließlich die *Struktur* des Transduktors entscheidend. Um dies zu motivieren, betrachten wir ein sprachgesteuertes Automotive-System, welches nur Telefonate mit Personen aus einer Datenbank zulässt, als laufendes Beispiel. Zur Vereinfachung gehen wir davon aus, dass es genügt, den Namen des Teilnehmers zu nennen. Außerdem gebe es noch eine Adresstabelle und eine Assoziationstabelle zwischen Person und Adresse, sodass die gesamte Datenbank wie in Tabelle 1.

Eine Entitätstabelle (im Bsp. sind Person und Adresse die einzigen Entitäten) modellieren wir als *Elementaren FST*, die wir im Anschluss erklären. Assoziationstabellen modellieren wir als *Anker*, die wir im zweiten Unterkapitel beschreiben.

3.1 Elementare FSTs

Elementare semantische Transduktoren sind azyklische deterministische gewichtete Transduktoren über dem Konfidenz-Wahrscheinlichkeitshalbring [6]. Die Konfidenzen geben an, wie sicher das genannte aus einer bestimmten Kategorie ist, also Vorname oder Nachname bspw. und die Wahrscheinlichkeiten entscheiden, welcher Eintrag einer Kategorie gemeint war.

Ein- und Ausgabebuchstaben sind an jeder Kante gleich ⁴. Die erste Kante wird immer mit dem Bezeichner derjenigen Entität, die durch den FST dargestellt werden soll, beschriftet (vgl. Abbildung 1):

$$e_0 = ((0, \sim.\text{Tabellenname}, \sim.\text{Tabellenname}, (0, 0), 1)$$

Anschließend ist es möglich sog. “semantische Kanten” einzufügen, die Kategorien zu Oberkategorien zusammenfassen. Im Beispiel ist das *Name*, welche *Vorname* und *Nachname* bündelt. Die Anzahl solcher semantischen Kanten ist beliebig. Anschließend fächern wir die Spaltennamen der Tabelle auf. Die Kanten mit den Spalteninhalten (Werte-Kanten) treffen gemäß ihrer Zusammengehörigkeit in der Tabellenzeile in einen gemeinsamen Zustand zusammen. Von dort aus gehen dann finale Kanten mit der Beschriftung des Entitätsbezeichners und der entsprechenden ID in den Endzustand (an dieser Stelle sind übrigens keine semantischen Kanten mehr erlaubt). Diese letzten Kanten ermöglichen uns eine Wahrscheinlichkeitsverteilung auf den eingehenden Kanten des Endzustandes, also den IDs, zu ermitteln.

Um eine semantische und auch tatsächliche Eindeutigkeit der jeweiligen Kantenbeschriftungen zu gewährleisten, besteht jede Beschriftung einer Kante, aus der Beschriftung der im Pfad vorgegangenen Kante, einem “.” und dem Bezeichner des aktuellen Objekts. In Abb. 1 ist dies durch die \sim angedeutet. Die Beschriftung $\sim.\text{Maier}.1$ meint eigentlich *Person.Name.Nachname.Maier.1*. Die Bezeichner der Werte-Kanten werden außerdem noch um die zugehörige ID erweitert.

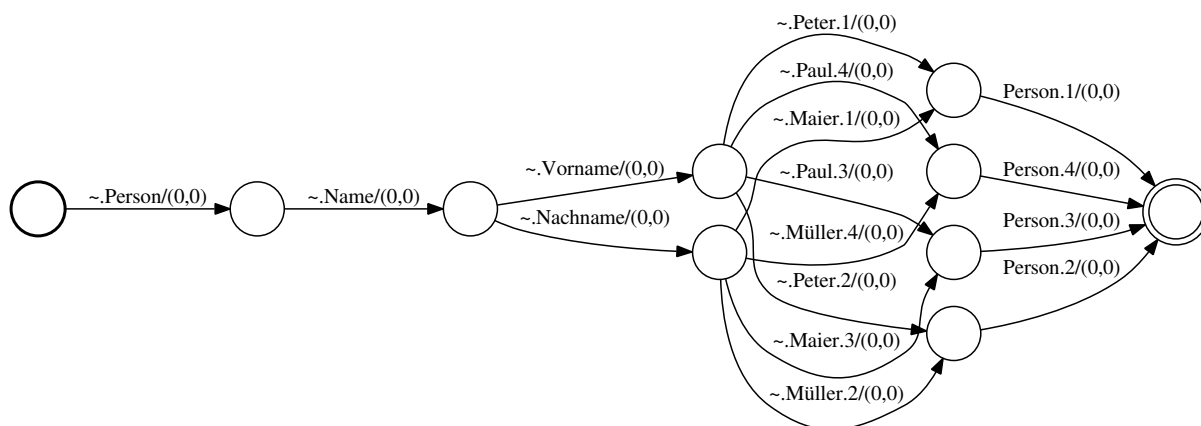


Abbildung 1: FST zur Entitätstabelle Person, ohne die Spalten *ID* und *Tel.*, der Datenbank in Tabelle 1 mit den, im Sinne des verwendeten Halbrings, neutralen Initialgewichten.

Um eine Intuition zur Interpretation der semantischen Transduktoren zu bekommen, muss man zunächst sich eine finale Kante und damit eine konkrete Person herausuchen. Alle Gewichte entlang der Pfade, die vom Startzustand über diese Kante zum Endzustand führen, geben an, welchen Beitrag die einzelnen Attribute und Werte zur Gewichtung der Person liefern.

Wir sind jetzt also in der Lage Entitäten einer Datenbank semantisch zu repräsentieren. Allerdings soll es auch möglich sein, die ID einer Entität über deren assoziierte Entität(en) zu iden-

⁴Daher wird die Beschriftung in Abb. 1 nur einmal notiert

tifizieren. Dies ist mit der Erweiterung der semantischen Transduktoren um Anker möglich, die im folgenden Abschnitt erklärt werden.

3.2 Anker

Ein Anker ist ebenfalls ein azyklischer deterministischer gewichteter Transduktor. Er verbindet einen Elementaren FST mit einem anderen elementaren FST gemäß einer Assoziationstabelle der Datenbank. Ein Anker besteht aus einem Präfix, dem assoziierten Elementaren FST und einem Suffix.

Ein Präfix besteht aus einer Kante, die mit dem Bezeichner des primären Elementaren FSTs (von dem ausgegangen wird) beschriftet ist und einer Kante mit dem Tabellennamen der Assoziationstabelle (vgl. Abb. 2).

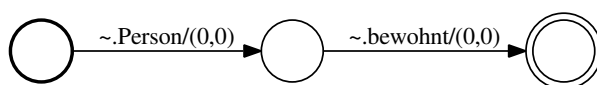


Abbildung 2: Präfix zum Anker "bewohnt" von "Person" ausgehend.

Die im Suffix vom Startzustand ausgehenden Kanten sind mit dem Bezeichner des primären FST, dem Tabellennamen der Assoziationstabelle, dem Bezeichner des eingehängten FSTs und der Zuordnung gemäß der Assoziationstabelle beschriftet. Jede Kante führt zu einem neuen Zustand. Von dort aus führen wiederum Kanten in einen gemeinsamen Endzustand. Die Beschriftung folgt dabei den finalen Kanten des primären FSTs (vgl. Abb. 3).

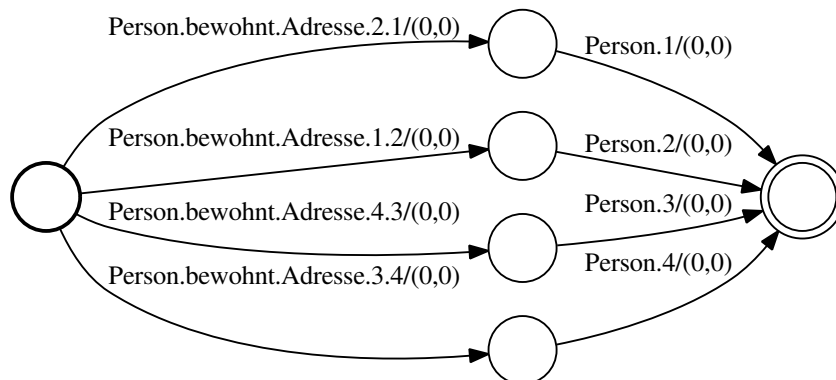


Abbildung 3: Suffix zum Anker *bewohnt* von *Person* ausgehend.

Der vollständige Anker ergibt sich dann aus Konkatenation von Präfix \otimes Elem. FST \otimes Suffix, wobei die Kanten des Elementaren FSTs gemäß seines Präfix' umbenannt werden (vgl. Abb. 4).

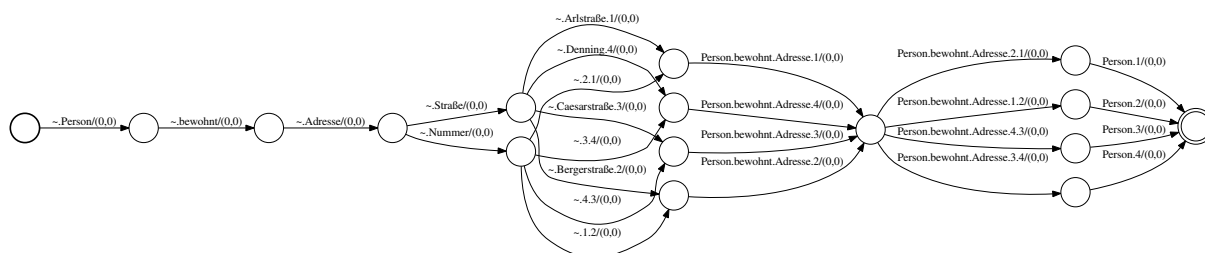


Abbildung 4: Anker für *Person* zur Assoziation mit *Adresse* über die Beziehung *bewohnt*.

Anschließend wendet man die Union-Operation auf den Anker und den primären Elementaren FST an und determiniert und minimiert das Ergebnis (vgl. Abb. 5):

$$\text{Min}(\text{Det}(\text{primärer FST} \oplus (\text{Präfix} \otimes \text{Elem. FST} \otimes \text{Suffix}))).$$

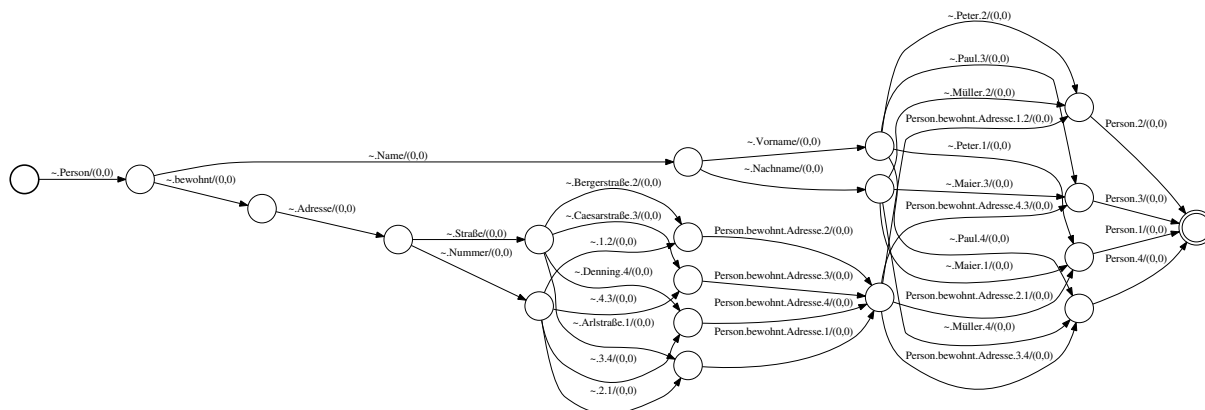


Abbildung 5: *Person* erweitert um den Anker *bewohnt*.

Es ist noch zu bemerken, dass durch das Prinzip des Ankers eine beliebig tiefe Struktur erzeugt werden kann:

Nachdem eine Person an einer Adresse wohnt, kann eine Adresse auch von einer Person bewohnt werden. Somit ist es offensichtlich, dass auch die eben eingehängte Struktur wiederum über Adresse um eine Person erweitert werden kann. Allerdings ist die Tiefe der Struktur durch die Erkenntiefe bzw. die Grammatik in jedem Dialogschritt beschränkt. Diese gibt vor wie tief verschachtelte Sätze pro Erkennvorgang verstanden werden können.

4 Update-Algorithmen

Wir beginnen den Dialog mit einem leeren - alle Gewichte entsprechen dem 1-Element des Halbrings - elementaren FST, etwa den aus Abb. 1. Dieser FST dient als Informationsstatus und wird pro Dialogschritt aktualisiert. In jedem Dialogschritt erhält man eine semantische Repräsentation des Erkennenergebnisses als FST (vgl. 6).

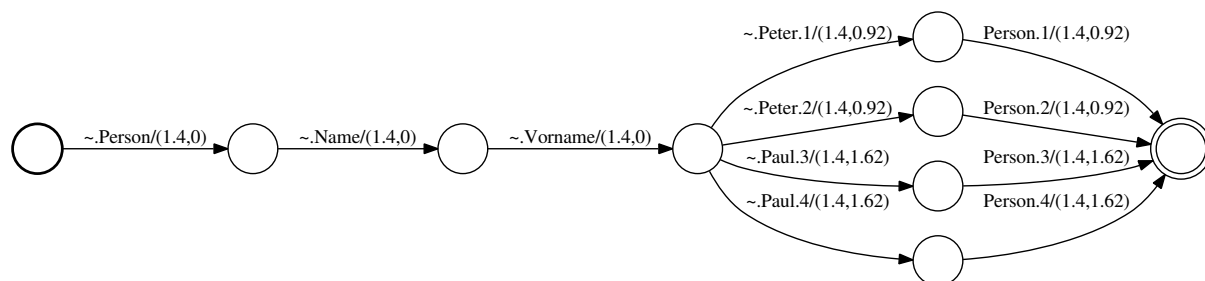


Abbildung 6: Ergebnis eines Dialogschritts in dem *Peter* gesagt wurde.

Eine Aktualisierung des Informationsstatus ist also eine Zusammenführung der Informationen aus dem Ergebnis und dem Informationsstatus. Um einen semantischen FST mit einem anderen zu aktualisieren haben wir zwei Algorithmen entworfen. Der erste ist eine Aneinanderreihung bekannter FST-Operationen, der zweite ein von uns entworfenes modifiziertes Vorgehen.

4.1 Update mit Standard-FST-Operationen

Gegeben sei ein (elementarer) FST I , in dem wir unseren aktuellen Wissenstand gespeichert ist. Außerdem haben wir ein Erkennungsergebnis R - ebenfalls als FST modelliert.

Im ersten Schritt wird je eine Kopie R' und I' von beiden FSTs erzeugt und die Gewichte der Kopien werden auf das multiplikative 1-Element des Halbrings gesetzt. Anschließend bildet man die Vereinigung $X = R' \cup I'$. X wird dann determiniert, minimiert und die ε -Kanten werden entfernt. X wird dann ebenfalls kopiert und wir erhalten X' . X und X' sind also jetzt die strukturelle Unifikation von I und R mit neutralen Gewichten. Abschließend tragen wir nun in X die Gewichte von I wieder ein - also an den Pfaden, die auch in I vorkommen und gehen analog für X' mit den Gewichten von R vor.

Der letzte Schritt ist dann die Komposition von X und X' wodurch die entsprechenden Gewichte aktualisiert werden. Das Ergebnis ist dann unser neuer Informationsstand I . Die Kopien können entweder verworfen oder in einer Dialoghistorie gespeichert werden, um bei späteren Entscheidungen miteinbezogen zu werden.

Aktualisiert man etwa den FST aus Abb. 1 mit dem aus Abb. 6, so erhält man den FST aus Abb. 7

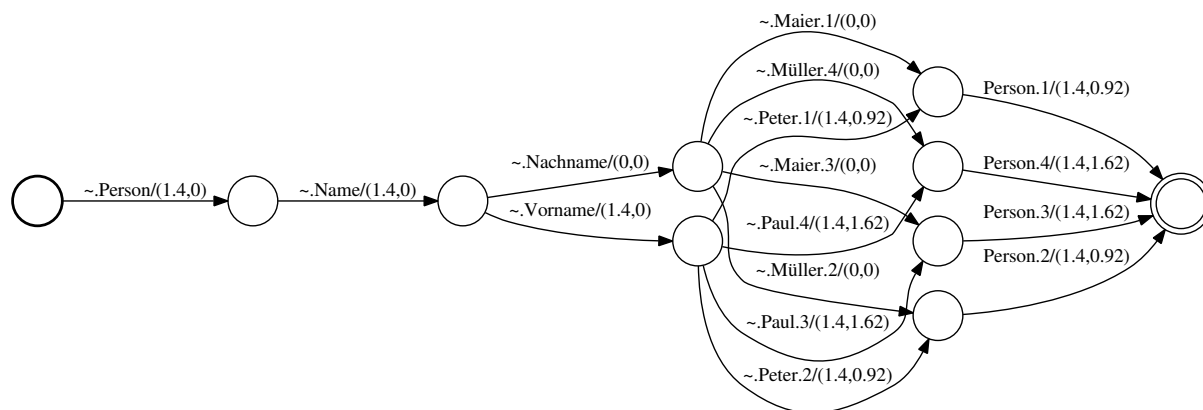


Abbildung 7: Initialer Informationsstatus aktualisiert mit der Äußerung *Peter*.

4.2 Update durch Unifikation

Der zweite Algorithmus setzt voraus, dass jeder Transduktor eine eindeutige Zustandsmenge hat. Haben zwei Transduktoren den gleichen Zustand in ihrer Zustandsmenge, dann hat dieser in beiden die gleiche Bedeutung. Daher liegt eine semantische Zustandsbezeichnung nahe, anstelle der natürlichen Zahlen. Gegeben also wieder ein Informationsstatus I und ein Erkennungsergebnis R .

Ist eine Kante k in beiden Transduktoren enthalten, also gilt $k \in E_I \cap E_R \neq \emptyset$ so werden die Kanten zu einer zusammengefasst und die Kantengewichte multipliziert. Alle anderen Kanten bleiben unberührt und werden in die Struktur übernommen.

5 Bedingungen für eine Semantikepräsentation

Jurafsky beschreibt in [2] eine Reihe von Bedingungen, die eine Semantikepräsentation erfüllen soll. Im Folgenden werden wir zeigen, dass unsere semantischen Transduktoren allen Bedingungen genügen. Dazu werden wir u.a. unsere Utterance-Meaning Pairs [4] verwenden. Ein

Utterance-Meaning Pair ist die Abbildung einer Wortfolge auf ihre Bedeutungen. Dabei wird das Konzept des statistischen Sprachmodells mit dem der Grammatik auf Basis des Skinner'schen Stimulus-Response Ansatzes kombiniert.

Verifizierbarkeit:

Da in unserem Entwurf ein Dialogsystem immer domainspezifisch ist, kann auch nur das verstanden werden, was tatsächlich in der Wissensbasis existiert. Dies wird durch die Utterance-Meaning Pairs sichergestellt.

Unambige Repräsentation:

Am Anfang eines jeden Dialogs stehen für alle Entitäten der Wissensbasis Elementare FSTs bereit. In jedem Dialogschritt können für sich widersprechende Semantiken weitere FSTs erzeugt oder nach Bewertung verworfen werden.

Kanonische Form:

Durch die Utterance-Meaning Pairs wird sichergestellt, dass jede semantisch äquivalente Repräsentation auf den gleichen Transduktor abgebildet wird.

Folgerungen und Variablen:

Folgerungen werden durch das Anker-Prinzip unterstützt. Variablen werden so modelliert, dass bspw. gleiche Werte gleiche Wahrscheinlichkeiten bekommen ("alle Peters anrufen").

Ausdruckskraft:

"The ideal situation, of course, would be to have a single meaning representation language that could adequately represent the meaning of any sensible natural language utterance" ([2], S. 585).

Ist eine Äußerung domänenspezifisch sinnvoll, sind wir in der Lage diese semantisch abzubilden. Ein wichtiger Teil der semantischen Transduktoren ist das Sicherstellen der Wahrscheinlichkeitsverteilung auf den entsprechenden Kanten. Hierfür haben wir eine geeignete Normierung entworfen (vgl. [5]).

Literatur

- [1] HUBER, M., C. KÖLBL, R. LORENZ, R. RÖMER und G. WIRSCHING: *Semantische Dialogmodellierung mit gewichteten Merkmal-Werte-Relationen*. In: *ESSV 2009, Tagungsband der 20. Konferenz, Dresden, 21. bis 24. September 2009*, S. 25–32.
- [2] JURAFSKY, D. und J. H. MARTIN: *Speech and Language Processing (2nd Edition) (Prentice Hall Series in Artificial Intelligence)*. Prentice Hall, 2000.
- [3] MOHRI, M.: *Weighted Automata Algorithms*. Springer, 2009.
- [4] WIRSCHING, G. und C. KÖLBL: *Language Modeling with Utterance-Meaning Pairs*. Techn. Ber., Angewandte Informatik, Universität Augsburg, 2011.
- [5] WIRSCHING, G., C. KÖLBL und M. HUBER: *Zur Logik von Bestenlisten in der Dialogmodellierung*. Beitrag zur ESSV 2011, Aachen, 28. bis 30. September 2011.
- [6] WIRSCHING, G., C. KÖLBL und M. HUBER: *The confidence-probability semiring*. Techn. Ber., Angewandte Informatik, Universität Augsburg, 2010.