

## Beispiel

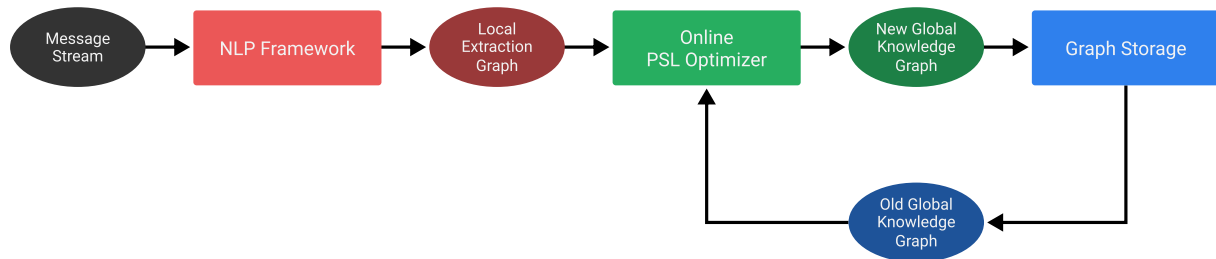


Abbildung 1: Vorläufiges grobes Strukturdiagramm des Verfahrens

Der vorgestellte Ansatz zur Wissensgraphkonstruktion lässt sich in zwei Hauptschritte unterteilen. Diese beiden Schritte werden beim Eintreffen einer Nachricht sequentiell ausgeführt. Im ersten Schritt wird mittels eines NLP-Systems ein lokaler Konzeptgraph extrahiert, der strukturiert den Inhalt der eingetroffenen Nachricht repräsentiert. Im zweiten Schritt wird dieser lokale Konzeptgraph dann in den bereits existierenden Wissensgraphen eingefügt. Mittels eines PSL-Systems werden dann Beziehungen zwischen dem eingefügten Graphen und dem bestehenden Graphen ermittelt.

Zur Veranschaulichung dieses Verfahrens folgt ein Beispiel. Gegeben sei folgende Nachricht, die am 12.06.2017 von Alice an Bob gesendet wurde: *"I don't think I saw you yesterday."*

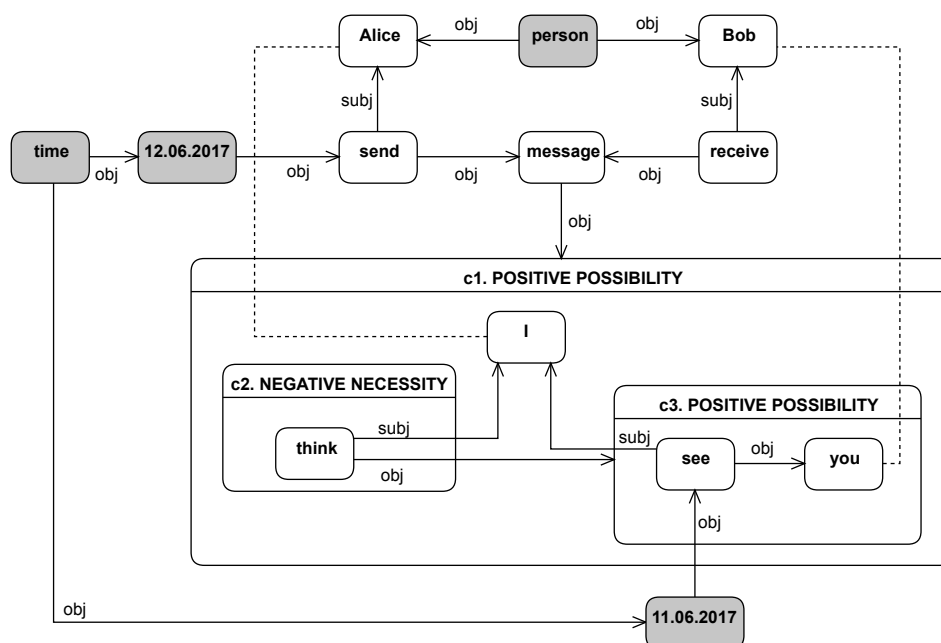


Abbildung 2: Lokaler Konzeptgraph der obigen Nachricht

Da Konzeptgraphen ein formales Kalkül von der Mächtigkeit einer Prädikatenlogik erster Ordnung sind, lässt sich dieser Graph auch durch folgenden äquivalenten Ausdruck beschreiben:

$$\begin{aligned}
& \exists person, time, alice, bob, msg, send, receive, d_1, d_2, c_1 : \\
& \quad label(person, 'person') \wedge label(time, 'time') \wedge label(alice, 'Alice') \wedge label(bob, 'Bob') \\
& \quad \wedge label(msg, 'message') \wedge label(send, 'send') \wedge label(receive, 'receive') \\
& \quad \wedge label(d_1, 12.06.2017) \wedge label(d_2, 11.06.2017) \wedge class(time) \wedge class(person) \wedge class(d_1) \\
& \quad \wedge class(d_2) \wedge obj(person, alice) \wedge obj(person, bob) \wedge obj(time, d_1) \wedge obj(time, d_2) \\
& \quad \wedge obj(d_1, send) \wedge obj(send, msg) \wedge obj(receive, msg) \wedge subj(send, alice) \\
& \quad \wedge subj(receive, bob) \wedge obj(msg, c_1) \\
& \quad \wedge (real(c_1) \leftrightarrow \exists c_3, i : label(i, 'I') \wedge i = alice \\
& \quad \quad \wedge (\neg \exists think : label(think, 'think') \wedge subj(think, i) \wedge obj(think, c_3)) \\
& \quad \quad \wedge (real(c_3) \leftrightarrow \exists see, you : \wedge label(see, 'see') \wedge label(you, 'you') \wedge you = bob \\
& \quad \quad \quad \wedge obj(see, you) \wedge obj(d_2, see) \wedge subj(see, i)))
\end{aligned}$$

Da Sprache nicht notwendigerweise Wahrheiten beschreibt, wird allen aus einer Nachricht extrahierten Konzepten eine Existenz- bzw. Nonexistenz-Aussage, sowie eine modale Notwendigkeit bzw. Möglichkeit zugeordnet. Diese Zuordnung erfolgt mittels der Einordnung aller Konzepte in einen Kontextbaum. Wurzelkonzepte, die außerhalb eines Kontextes stehen werden per Definition als notwendigerweise existent angenommen. Im Beispiel sind u. a. *alice*, *bob* und *msg* Wurzelkonzepte.

Die Nachricht *msg* führt im Beispiel nun einen möglicherweise existenten Kontext  $c_1$  ein. Alle Konzepte innerhalb eines möglichen Kontextes sind höchstens so real, wie dieser Kontext. Die Realität von  $c_1$  hängt primär von der Interpretation des einleitenden Konzeptes *msg* ab, d. h. inwiefern der Inhalt von *msg* der Realität entspricht. Um dies abschätzen zu können, ist zusätzliches domänenspezifisches Wissen erforderlich, welches aus dem kompletten Wissensgraphen stammen muss; da in der NLP-Phase Nachrichten isoliert betrachtet werden, kann eine Abwägung der Realität von extrahierten Möglichkeiten also erst in der PSL-Phase erfolgen.

Im Beispiel enthält  $c_1$  einen notwendigerweise nonexistenten Kontext  $c_2$ . Dies bedeutet, dass *think* nicht existiert gdw.  $c_1$  real ist. *think* führt nun wiederum einen möglicherweise existenten Kontext  $c_3$  ein. Um die Realität von  $c_3$  abzuschätzen ist, wie schon bei  $c_1$ , domänenspezifisches Wissen aus dem kompletten Wissensgraphen erforderlich; konkret muss ermittelt werden, welche Implikationen die Nonexistenz des Denkens von  $c_3$  auf die Realität von  $c_3$  hat.

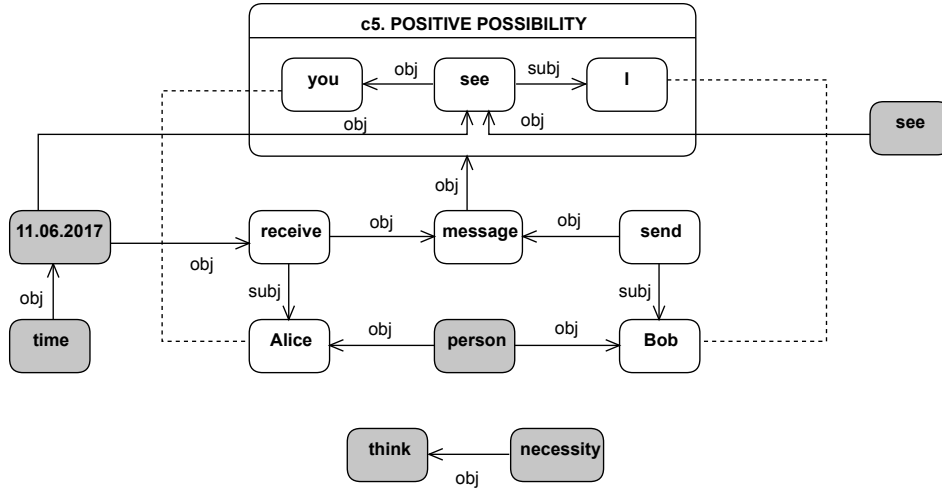


Abbildung 3: Wissensgraph vor Einfügen der Nachricht

Abb. 3 zeigt den kompletten Wissensgraphen vor Einfügen des Extraktionsgraphen aus Abb. 2. In diesem Beispiel wird davon ausgegangen, dass Bob zuvor am 11.06.2017 eine Nachricht an Alice mit dem Inhalt *“I saw you today.”* geschickt hat. Außerdem wird davon ausgegangen, dass im Wissensgraphen bereits Informationen über das Konzept des Denkens vorhanden sind; konkret wird durch die Verknüpfung mit *necessity* abgebildet, dass das Denken als Notwendigkeit verstanden werden kann, d. h. dass für jede Aussage gilt, dass sie entweder gedacht wird oder ihr Gegenteil gedacht wird. Diese Interpretation ist selbstverständlich nur eine von vielen; sie wurde gewählt, da sie oftmals zutrifft und das Modell dabei einfach hält.

Das Einfügen der Nachricht erfolgt im Rahmen der PSL-Phase (siehe Abb. 4). Im ersten Schritt wird der aktuelle Wissensgraph und der einzufügende Extraktionsgraph vereinigt. Knoten mit gleicher Identität werden dabei durch Vereinigen der Inzidenzkantenmengen zusammengefasst, dies ist geschehen bei *time*, *person*, *alice*, *bob* und 11.06.2017.

Anschließend erfolgt eine PSL-Optimierung, während der neue Kanten zwischen eingefügten und bestehenden Knoten entstehen. Eine der grundlegendsten PSL-Regeln ist das Erkennen des Auftretens bekannter Konzepte mittels Label-Ähnlichkeit.

$$w_1 : \text{similar}(A, B) \wedge \text{class}(A) \rightarrow \text{obj}(A, B)$$

Mit dieser Regel lässt sich z. B. das globale Konzept des Denkens mit der lokalen Repräsentation des Denkens in  $c_1$  verknüpfen.  $\text{similar}(A, B)$  könnte hierbei z. B. über die Levenshtein-Distanz der Label von  $A$  und  $B$  definiert werden. Da PSL eine probabilistische Logik ist, werden während der PSL-Phase alle logischen Aussagen als Wahrscheinlichkeiten  $\in [0, 1]$  interpretiert. Dementsprechend sind PSL-Regeln nicht als Forderungen zu verstehen, die unbedingt eingehalten werden müssen, sondern eher als Indikatoren, die die Wahrscheinlichkeiten der implizierten Atome heben oder senken.

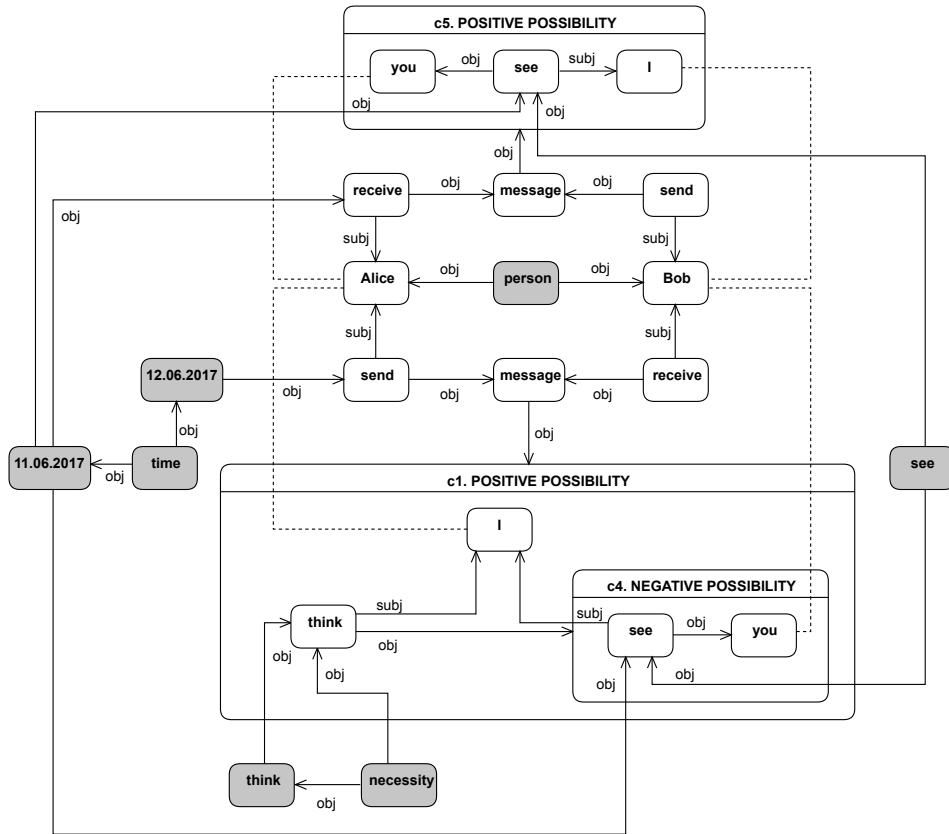


Abbildung 4: Wissensgraph nach Einfügen der Nachricht

Um die Notwendigkeit der lokalen Denken-Repräsentation aus der Notwendigkeit des globalen Konzeptes zu folgern, wird eine weitere grundlegende PSL-Regel zur Beschreibung der Transitivität der *obj*-Relation benutzt:

$$w_2 : class(A) \wedge class(B) \wedge obj(A, B) \wedge obj(B, C) \rightarrow obj(A, C)$$

Zur Erklärung des Wissensgraphen aus Abb. 4 ist nun noch eine weitere Regel notwendig. Über die Relation des lokalen Nicht-Denkens zum *necessity*-Konzept, wurde gefolgert, dass ein lokales Denken des Gegenteils von *c3* existiert. Für diese Schlussfolgerung muss das *necessity*-Konzept eine Bedeutung erhalten. Idealerweise sollte diese Bedeutung als Teil des Wissensgraphen kodiert sein und mittels weiterer grundlegender PSL-Regeln zu der gewünschten Interpretation führen. Da PSL-Regeln allerdings weniger mächtig als aussagenlogische Ausdrücke und somit als prädikatenlogische Ausdrücke sind, ist dies nicht ohne Weiteres möglich. Um das PSL-System einfach zu halten, werden daher stattdessen domänenspezifische Regeln eingeführt.

$$w_3 : class(N) \wedge label(N, 'necessity') \wedge obj(N, A) \wedge neg(A) \wedge obj(A, C) \wedge possib(C) \rightarrow \neg real(C)$$

In einem vollständigen PSL-System existieren selbstverständlich noch andere Regeln und die drei genannten sind primär als Beispiel zu verstehen. Die genaue zu verwendende Regelmeng muss noch spezifiziert werden. Einen Ausgangspunkt für die Wahl der Regeln bietet dieses Paper: <https://arxiv.org/pdf/1607.00992v1.pdf>.