Entwurf 1

Bachelorarbeit Proposal

Probabilistische online Wissensgraphkonstruktion aus natürlicher Sprache

Clemens Damke

Matrikelnr. 7011488

28. März 2017

betreut von

Prof. Dr. Eyke Hüllermeier
Intelligente Systeme
Institut für Informatik
Universität Paderborn

1 Motivation und Hintergrund

In den letzten Jahren hat die Repräsentation von Wissensbasen durch Graphen immer mehr an Bedeutung gewonnen. Google benutzt solche sog. Wissensgraphen z. B. zum Beantworten von komplexen Suchanfragen.

Die Grundidee dabei ist, Entitäten durch Knoten und Relationen durch Kanten abzubilden. Entitäten können konkrete Dinge, wie z. B. Personen, aber auch abstrakte Konzepte, wie z. B. historische Epochen, sein. Relationen beschreiben beliebige Beziehungen zwischen den Entitäten, z. B. $Person(\text{Da Vinci}) \xrightarrow{\text{lebte in}} Epoche(\text{Renaissance})$. Die Entität, von der eine solche Relation ausgeht, wird als Subjekt und die Zielentität als Objekt der Relation bezeichnet.

Da solche Graphen in zahlreichen Domänen einsetzbar sind, wird deren automatisierte Konstruktion bereits seit Jahren erforscht. Manuelles Konstruieren und vor allem anschließendes Warten und Aktualisieren von Wissensgraphen ist aufgrund der abzubildenden Datenmengen nicht praktikabel. Bei einer maschinellen automatisierten Konstruktion sind insbesondere zwei Anforderungen problematisch:

1. Das Verarbeiten von unstrukturierten Eingaben, wie z. B. natürlichsprachlichen Texten.
2. Effizientes Eingliedern neuer Informationen in einen bestehenden Wissensgraphen. Dieses Eingliedern von Informationen umfasst im Speziellen:
   a) **Entity Resolution:** Hinzukommende Entitäten, die bereits im Graphen enthalten sind, müssen als Duplikate erkannt werden. Dies ist i. d. R. nicht trivial, da die selbe Entität durch viele verschiedene, oftmals vom Kontext abhängige, Token repräsentiert werden kann; z. B. *Bob* vs. *Robert* oder *Der Papst* vs. *Franziskus*.
   b) **Link Prediction:** Hinzukommende Entitäten müssen mit bereits bestehenden Entitäten in Relation gesetzt werden. Hinzukommende Relationen können zudem benutzt werden um andere Relationen zu inferieren; z. B.

$$Weiblich(A) \wedge B \xrightarrow{\text{Sohn von}} A \implies A \xrightarrow{\text{Mutter von}} B$$

Die Kombination dieser beiden Anforderungen ist interessant, da das meiste verfügbare Wissen in natürlichsprachlicher Textform vorliegt und zudem permanent neues Wissen entsteht. Ein automatisiertes Wissensgraphkonstruktionsverfahren sollte daher beide Anforderungen berücksichtigen.

Neben diesen Anforderungen bzgl. der Extraktion von Wissen ist zudem wichtig, wie genau der Graph repräsentiert wird. Zusätzlich zu Knoten bzw. Entitäten und Kanten bzw. Relationen sind oftmals weitere Metadaten relevant. Dazu zählt insbesondere die Inferernzkonfidenz des Link Predictors. Da natürlichsprachliche Eingabeinformationen häufig unvollständig oder fehlerhaft sind, ist es für die Interpretation und Analyse des resultierenden Graphen hilfreich jeder Relation eine Konfidenz $\in [0, 1]$ zuzuordnen. Das Ergebnis ist ein sog. probabilistischer Wissensgraph.

2 Ziele der Arbeit

Das beschriebene Problem der online Wissensgraphkonstruktion aus natürlicher Sprache soll im Kontext von textueller Kommunikation zwischen Menschen näher untersucht werden. Gegeben sei ein Stream von Textnachrichten, denen jeweils ein Inhalt, ein Absender, eine Menge von Empfängern und ggf. weitere Metadaten, wie z. B. Absendezeit, Absendeort oder IP-Adresse, zugeordnet ist. Ziel ist es ein skalierbares Verfahren zu entwickeln, welches aus diesem Nachrichtenstrom einen Wissensgraph konstruiert. Insbesondere folgende Informationen sollen im resultierenden Graphen abgebildet werden:

1. Jede Nachricht ist eine Entität und soll mit allen zugehörigen Attributen als Knoten eingefügt werden.
2. Jede Person, die eine Nachricht sendet, empfängt oder darin vorkommt, soll als Entität erkannt werden. Personen-Entitäten, die durch eine Entity Resolution mit einer gewissen Konfidenz als identisch erkannt werden, sollen durch eine entsprechend konfidente Äquivalenzkante in Relation gesetzt werden.
3. Neben Personen sollen auch Orte, die in Nachrichten vorkommen, erkannt werden. Idealerweise werden erkannten Orten zudem Geokoordinaten zugeordnet.
4. Jeder Kante soll, sofern sinnvoll und möglich, ein Zeitpunkt oder Zeitraum zugeordnet werden. So kann z. B. die Anwesenheit einer Person an einem Ort temporal einsortiert werden.

Das gesuchte Verfahren soll zudem erweiterbar sein. Es sollen Schnittstellen eingeplant werden, um neben dem Textextraktor auch andere Extraktoren, z. B. für Bilder und Audioaufnahmen, hinzufügen zu können. Außerdem sollen die Entity Resolution und Link Prediction Verfahren um domänenspezifische Expertensysteme erweiterbar sein. Ein Beispiel hierfür ist die Geokoordinatenzuordnung zu extrahierten Ortsnamen. Durch das Erweitern der Link Prediction, kann diese Zuordnung mithilfe eines entsprechenden Expertensystems erfolgen.

Wie zuvor erwähnt, soll das gesuchte Verfahren skalierbar sein. Dies bedeutet konkret, dass die Laufzeit für das Verarbeiten einer Nachricht idealerweise ausschließlich von der Größe der Nachricht und nicht von der bisherigen Größe des Wissensgraphen abhängt. Sofern dies nicht möglich ist, soll der Einfluss der Graphgröße auf die Laufzeit soweit wie möglich reduziert werden. Die Skalierbarkeit des Verfahrens umfasst zudem auch die Eignung für ein massivparalleles Ausführen im Cluster.

Im Rahmen der Arbeit soll ein derartiges Verfahren zuerst entworfen und anschließend implementiert werden. Die Skalierbarkeitsanforderungen sollen hierbei primär im Entwurf des Verfahrens berücksichtigt werden. Die Parallelisierbarkeit der Implementation ist für diese Arbeit nachrangig.

3 Verwandte Arbeiten

3.1 Natural Language Processing

Bevor der natürlichsprachliche Inhalt einer Nachricht in den zu konstruierenden Wissensgraphen eingefügt werden kann, muss dieser zuerst zerlegt werden. Diese Zerlegung, genannt Information Extraction (IE), ist ein Teilgebiet des Natural Language Processings (NLP). IE liefert eine Menge von Extraktionstripeln, die zusammen einen Extraktionsgraphen bilden.

Zwei konkrete IE Verfahren, die gute Ergebnisse liefern, sind Stanfords OpenIE Verfahren und das OpenIE Verfahren der University of Washington. Stanfords Verfahren ist Teil des CoreNLP Projektes, welches neben der Tripelextraktion zudem Module zum Kategorisieren von Entitäten (Person, Ort, Zeitpunkt, etc.) und zum Erkennen von Koreferenzen beinhaltet. Im Rahmen der Arbeit ist zu evaluieren, welche Kombination von Verfahren sich gut als Ausgangspunkt für die Wissensgraphkonstruktion eignet.

3.2 Wissensgraphkonstruktion

Das zuvor beschriebene Problem der Wissensgraphkonstruktion lässt sich als Teilgebiet des Statistical Relational Learnings (SRL) beschreiben. In der Literatur finden sich im Wesentlichen drei Klassen von Verfahren, die auf verschiedenen Annahmen über die Korrelation der zu verarbeitenden Informationen basieren:

1. **Latent Feature Models:** Alle Relationen werden als bedingt unabhängig angenommen, sofern bestimmte Eigenschaften über Subjekte und Objekte der Relationen gegeben sind.
2. **Graph Feature Models:** Alle Relationen werden als bedingt unabhängig angenommen, sofern bestimmte Eigenschaften der Struktur des Graphen im Allgemeinen gegeben sind.
3. **Markov Random Fields:** Es wird angenommen und erlaubt, dass alle Relationen lokale Abhängigkeiten voneinander haben können.

Ein Beispiel für ein Latent Feature Modell ist das RESCAL-Verfahren, welches auf Tensorfaktorisierung basiert. Die Grundidee dabei ist es allen Entitäten i einen Feature-Vektor $e_i \in \mathbb{R}^H$ zuzuordnen und für alle Relationen k eine Gewichtsmatrix $W_k \in \mathbb{R}^{H \times H}$ zu finden. Die Konfidenz in die Existenz einer Relation $i \xrightarrow{k} j$ wird durch $e_i^T \; W_k \; e_j$ beschrieben. Diese Definition ermöglicht eine sehr schnelle Link Prediction, da lediglich ein Vektor-Matrix-Vektor-Produkt berechnet werden muss. RESCAL liefert gute Ergebnisse, wenn die vorherzusagenden Relationen globale Abhängigkeiten aufweisen. Lokal stark zusammenhängende Teilgraphen werden allerdings schlecht erkannt, da nur der Feature-Vektor und nicht die Nachbarschaft einer Entität berücksichtigt wird; ein Beispiel hierfür sind symmetrische Relationen.

$$A \xrightarrow{\text{verheiratet mit}} B \implies B \xrightarrow{\text{verheiratet mit}} A$$

Komplementär zu den Latent Feature Modellen sind die Graph Feature Modelle. Statt Entitäten in einen Feature-Raum einzubetten, wird hier die Nachbarschaft der Entitäten betrachtet.

Ein Beispiel hierfür ist der Path Ranking Algorithmus (PRA). PRA ermittelt Relationen durch zufälliges Durchwandern des Graphen. Um die Stärken der Latent Feature und Graph Feature Modelle zu kombinieren, wurden Hybrid-Modelle, wie z. B. das Additive Relational Effects (ARE) Verfahren, entwickelt, welches die Konfidenzen von RESCAL und PRA addiert.

Fundamental verschieden von diesen beiden Verfahren sind Markov Random Fields (MRFs). Hier sind prinzipiell Abhängigkeiten zwischen allen Relationen möglich, was MRFs sehr flexibel macht. Da dies hinsichtlich der Laufzeit schnell impraktikabel wird, wird das Modell um einen Abhängigkeitsgraphen erweitert, der die Anzahl von betrachteten Abhängigkeiten zu reduzieren. Der Abhängigkeitsgraph darf dabei nicht mit dem Wissensgraphen verwechselt werden: Ersterer beschreibt statistische Abhängigkeiten zwischen Klassen von Relationen, während letzterer Abhängigkeiten zwischen spezifischen Fakten beschreibt. Zur Modellierung von Abhängigkeitsgraphen werden i. d. R. Kalküle verwendet, die an eine Prädikatenlogik erster Ordnung angelehnt sind. Das Finden eines Wissensgraphen ist in diesem Modell äquivalent zum Lösen des MAX-SAT-Problems. Wählt man ein Kalkül in dem die Atome stetig, also aus $[0, 1]$ statt aus $\{0, 1\}$ sind, und Formeln zudem ausschließlich Disjunktionen und Negationen enthalten, erhält man ein sog. Hinge-Loss-MRF. Ein konkretes Kalkül zur Spezifikation von HL-MRFs ist die Probabilistic Soft Logic (PSL). MAX-SAT lässt sich für solche HL-MRFs effizient und parallelisierbar mit dem konvexen Alternating Direction Method of Multipliers (ADMM) Optimierungsverfahren lösen. In dessen ursprünglichen Form ist ADMM allerdings ausschließlich für offline Inferenz geeignet; der Wissensgraph müsste also bei jeder Eingabe neu konstruiert werden. Um dieses Problem zu lösen, wurde das Budgeted Online Collective Inference (BOCI) Verfahren entwickelt. BOCI nutzt Metadaten, die während der Ausführung von ADMM anfallen, um eine Bewertung für jedes Atom zu berechnen. Die Bewertung eines Atoms beschreibt, wie groß die erwartete Wertveränderung beim Eintreffen neuer Informationen ist. Kommen nun neue Informationen an, müssen diese ausschließlich zusammen mit den m höchstbewerteten bereits existierenden Atomen betrachtet werden, die Werte aller anderen Atome werden fixiert. Je höher das Budget m, desto höher ist die Qualität in Relation zu einer Neukonstruktion des Graphen. Es wurde empirisch gezeigt, dass die Inferenzqualität mit BOCI oft nur unwesentlich schlechter ist als bei einer kompletten offline Inferenz.

Mittels ADMM und BOCI lässt sich PSL daher für die online Wissensgraphkonstruktion einsetzen. Der Vorteil dieses Ansatzes gegenüber eines Latent Feature oder Graph Feature Modells ist, dass sich andere domänenspezifische Expertensysteme leicht in eine PSL Inferenz integrieren lassen. PSL erlaubt nämlich die Definition von Funktionen und Prädikaten. Diese können benutzt werden, um z. B. die Levenshtein-Distanz zweier Zeichenketten mit in die Entity Resolution einfließen zu lassen. Aufgrund des Ziels der Erweiterbarkeit wird PSL daher als Grundlage für diese Arbeit gewählt.

4 Vorläufige Struktur

5 Zeitplan