ZUR AUTOMATISCHEN SCHÄTZUNG VON KOSTENFUNKTIONEN AUS DIALOGEN

Benjamin Weiss, Stefan Hillmann, Thilo Michael, Tilo Himmelsbach

Technische Universität Berlin
benjamin.weiss@tu-berlin.de

Kurzfassung: In diesem Kurzbeitrag wird der Ansatz von Reinforcement Learning für die Lernen von Systemverhalten von Sprachdialogsystemen kurz vorgestellt. Dabei wird insbesondere auf das Potential der automatische Rekonstruktion von Kostenfunktionen aus Dialogdaten, dem Inversen Reinforcement Learning (IRL) eingegangen, und eine Forschungsaufgabe postuliert, um IRL auf seinen Nutzen für die Untersuchung und Anwendung von Meta-Kommunikation in Sprachdialogsystemen hin zu überprüfen.

1 Einleitung

Für die Mensch-Maschine-Kommunikation kann das interaktive Verhalten sprachbasierter Systeme auf verschiedene Arten realisiert werden. Während für einfache Dienste mit kleinen und begrenzten Anwendungsdomänen Zustandsautomaten direkt von Experten implementiert werden können, verlangen komplexere Anwendungen andere Methoden. So können grafische Editoren die Erstellung und Wartbarkeit von Zustandsautomaten erleichtern. Für aufgabenorientierte Systeme, deren Dienst Datenbankabfragen nutzt, kann Systemverhalten teilweise automatisiert werden, indem notwendige und optionale Informationen definiert werden, die vom Nutzer erfragt bzw. bereitgestellt werden müssen. Regelbasiert können diese Informationen zur Datenbankabfrage, und die Merkmale der Datenbankantwort, bspw. die Anzahl der Treffer, zur Auswahl der nachfolgenden Systemaktion genutzt werden. Solche Ansätze sind bspw. in der VXML Spezifikation [1] standardisiert und werden üblicherweise auch auf dem Markt genutzt. Andere Ansätze, die Verfahren des maschinellen Lernens nutzen, sind bislang weitestgehend auf die Forschung beschränkt, auch wenn es insbesondere für Dienste der „One-Shot Interaction“, wie etwa „Frequently-Asked-Questions“ eine Veränderung im Stand der Technik gibt.

Eine Beschränkung eines einfachen, überwachten Lernens von Systemverhalten aus Daten liegt in der Generalisierbarkeit über die Daten hinweg, was insbesondere für echte Dialoge deutlich wird, die im Gegensatz zur One-Shot-Interaction mehrere Gesprächsrunden (Turns) aufweisen. Für die statistische Modellierung von Systemverhalten solcher Aktionssequenzen wird deshalb in der Forschung, neben komplexen Architekturen tiefer neuronaler Netze, Reinforcement Learning (RL) verwendet. Dieser Ansatz soll Optimierungsprobleme von Sequenzen lösen, für die erst zum Sequenzende deutlich wird, ob das Systemverhalten erfolgreich war oder nicht. Dabei erlernt ein Computeragent das eigene Verhalten durch Ausprobieren in Interaktion mit seiner Umgebung selbst und (im ursprünglichen Sinne) ohne Datenbasis. Verbreitete Aufgabengebiete für RL sind künstliche Spieler von Computerspielen [2, 3] und Robotermotorik [4]. Für solche Optimierungsprobleme wird die Lösungsspezifikation durch menschliche Experten als sehr (zu) komplex angesehen, während die Umgebung einfach simuliert werden kann oder bereits implementiert ist. Die Resultate sind beeindruckend und übersteigen, bspw. für einige Computerspiele, menschliche Leistungen.

Für das Gebiet der Sprachinteraktion bietet sich die Anwendung von RL durchaus an, allerdings ist die Umgebung, hier also der menschliche Gesprächspartner, nicht nur variabel, sondern auch noch schwer zu modellieren. Da RL auf große Datenmengen, also eine hohe Anzahl von Gesprächen zum Lernen, angewiesen ist, muss das Lernen mit echten Gesprächspartnern, zumindest weitestgehend, ausgeschlossen werden. Abhilfe schaffen hier bspw. simulierte Nutzer als Interaktionspartner oder ein Vortrainieren mit annotierten Daten.

2 Temporal-Difference Reinforcement Learning

RL wird im einfachsten Fall verwendet, um Systemverhalten als Markov Decision Process (MDP) [5] zu modellieren. Dabei wird ein MDP spezifiziert als S, A, R, T; mit dem Raum aller Zustände des Systems S, dem Raum aller Systemaktionen A, den Übergangswahrscheinlichkeiten T, der Kostenfunktion R für das Erreichen eines Zustands s'. Für aufgabenorientierte Sprachdialogsysteme werden Systemzustände typischerweise über eine Auswahl und Abstraktion von relevanten Attributen der Domäne (bspw. *Art der Küche* wurde bereits vom Nutzer für eine Restaurantauskunft angegeben) und zur Metakommunikation (die Art der Küche wurde vom Nutzer bestätigt) definiert. Ziel von RL ist es nun, eine Policy $\pi(s,a)$ zu bestimmen, die dem Systemverhalten entspricht, also die durchzuführende Aktion a in Zustand s benennt, und zudem optimal (π^*) im Sinne der Kostenfunktion ist.

Eine etablierter Ansatz zur Bestimmung von π^* ist das *Q-learning* [6], das hier exemplarisch dargestellt wird. Mit Q-learning werden die erwarteten kumulativen Kosten zu Dialogende minimiert bzw. der Ertrag (*return*) maximiert. Dazu wird eine, dem Verfahren namensgebende, Action-Value-Matrix Q(s,a) definiert, die für jede Aktion a in jedem Zustand s einen Wert enthält. Um aus beendeten Dialogen und der Kostenfunktion die einzelnen Zustände und durchgeführten Aktionen zu bewerten, müssen iterativ Werte dieser Matrix anhand der folgenden Funktion zugewiesen bzw. erneuert werden, da die anfänglichen Erwartungswerte invalide sind. Als neuen Wert wird Q(s,a) die Summe von altem Wert und der zeitlichen Differenz der Zustände s und s' zugewiesen.

$$Q_{neu}(s,a) = Q(s,a) + \alpha[r + \gamma \max_a Q(s',a) - Q(s,a)] \tag{1}$$

Dabei entspricht r den direkten Kosten oder der direkten Belohnung für die Transition in Zustand s' aus der Kostenfunktion, während die temporale Differenz von $Q(s,a)$ zu $Q(s',a)$ für die beste Aktion in s' gewählt wird. Der sogenannte *discount factor* γ steuert hierbei die Relevanz des direkten gegenüber späteren Kosten, während die Lernrate α die Relevanz der neuen Information gegenüber dem alten Wert gewichtet.

Neben MDP werden Policies von Sprachdialogsystemen auch als Partially Observable Markov Decision Process (POMDP) [7] realisiert, um der Unsicherheit beim Verstehen der Nutzeräußerung Rechnung zu tragen, und damit der Unsicherheit, in welchem Zustand sich das System tatsächlich befindet, was von der korrekten Interpretation der Nutzeräußerung abhängt. Dazu werden anstatt diskreter Zustände, Wahrscheinlichkeitsverteilungen über alle oder eine Menge von Zuständen verwendet. Andere Implementierungen, allerdings nicht für Sprachdialogsysteme, betreffen neuronale Netze [2].

3 Die Kostenfunktion im Reinforcement Learning

Das Lernen von Systemaktionen auf Nutzereingaben selbst geschieht über eine Kostenfunktion. Solche Kostenfunktionen sind üblicherweise von Experten definiert und einfach gehalten. Interaktionserfolg wird bspw. einmalig stark belohnt (etwa 100), während Interaktionsdauer über

geringe Kosten per Aktion (etwa -1) abgebildet werden. Ziel wäre in dem gerade genannten Beispiel, ein Systemverhalten (Policy) zu erlernen, das erfolgreiche, aber kurze Dialoge erzeugt.

Die Definition von Kostenfunktionen stellt eine der zentralen, händischen Aufgaben bei einem RL-basierten – und damit einem weitestgehend automatischen – Ansatz zur Dialoggenerierung dar. Es wird dabei die Meinung vertreten, dass Kostenfunktionen robuster für Umgebungsveränderungen sind, also generalisierbarer – etwa beim Domänenwechsel, als Policies [8]. Alternativ zur direkten Spezifikation lassen sich Kostenfunktionen auch aus Daten lernen. Dies erlaubt die Analyse von Daten aus menschlichen Dialogen, die in diesem Forschungsfeld als "optimal" angesehen werden können. Dies wird im nächsten Abschnitt dargestellt.

4 Inverses Reinforcement Learning

Die Rekonstruktion einer Kostenfunktion für RL wurde mit unterschiedlichen Verfahren bspw. in Ng und Russel [8] beschrieben. Das Ziel ist es, aufgrund einer beobachtbaren oder austestbaren Policy eine zugrundeliegende Kostenfunktion zu rekonstruieren für die diese Policy optimal ist. Dabei ist die Kostenfunktion nicht eindeutig, sondern es können verschiedene Lösungen gefunden werden, von denen die nicht-trivialen bevorzugt werden sollten.

Allerdings sind die drei in Ng und Russel [8] beschriebenen Ansätze nicht ohne Nutzersimulation umsetzbar, da sie alle eine Umgebung voraussetzen, mit der interagiert werden kann. Die wenigen Forschungsarbeiten zu IRL bei Sprachdialogsystemen konzentrieren sich auf die Nutzung, zumindest teilweiser, annotierter Korpora und damit auf die Nutzung von Verfahren des *boot-strappings* aus solchen Daten.

So wurde für POMDP-basierte Sprachdialogsysteme menschliche Dialoge annotiert. Diese entstammen einem experimentellen Aufbau, in dem ein menschlicher Experte relevante Teile des Sprachdialogsystems ohne Wissen der Nutzer ersetzt (Wizard-of-Oz Paradigma) um so „optimales Systemverhalten" zu generieren [9]. Im Vergleich zeigt sich die rekonstruierte Kostenfunktion in Bedingungen mit mittleren und niedrigen Spracherkennungsfehlern einer von Experten definierten Kostenfunktion überlegen.

Asri [10] wendet IRL nicht auf menschliche Dialogkorpora an, sondern nutzt teilweise bewertete Dialoge der Mensch-Maschine-Interaktion, um Kostenfunktionen zu rekonstruieren. Mit ihrem Ansatz zeigt sie, dass Sprachdialogsysteme grundsätzlich mit IRL optimiert werden können, indem sowohl der Zustandsraum, also auch die Kostenfunktion aus solchen Daten geschätzt werden. Bei der Anwendung zweier Verfahren für IRL, dem *Reward Shaping* und dem *Distance Minimisation* zur Ermittlung einer Kostenfunktion erweist sich das Lerntempo der Systeme höher als bei dem Vergleichs-System, welches direkt das verwendete Bewertungskriterium nutzt. Allerdings benötigt dieser Ansatz entweder annotierte und bewertete Dialogdaten mit einem echten Sprachdialogsystem, oder einen validen Schätzer des Kriteriums, bspw. für Nutzerzufriedenheit, und simulierte Daten.

Solche automatisch rekonstruierten Funktionen sind jedoch komplexer als übliche manuell definierte, da sie ohne weitere Aggregierung und Vereinfachung, als Kosten für jeden erreichten Zustand (oder für jeden Zustandsübergang [11]) rekonstruiert werden [10]. Allerdings passen sie für den jeweiligen Datensatz potentiell besser, benötigen aber auch (annotierte) Daten. Zudem muss die Übertragbarkeit einer solchen Policy für den neu zu lernenden Agenten erst abgeschätzt bzw. überprüft werden, und es gibt für viele, potentiell optimale, Policies bzw. Datensätze zahlreiche Kostenfunktionen.

Ein potentieller konzeptueller Vorteil gegenüber manuell definierten Kostenfunktionen betrifft aber die Einbeziehung von Systemaktionen zur Meta-Kommunikation, falls die rekonstruierte Kostenfunktion Meta-Kommunikation zur Zustandsbeschreibung nutzt. Dies würde in einer Kostenfunktion resultieren, die ein Lernen von Bestätigungsverhalten und Fehlerbehand-

lung erlaubt, das in den vorliegenden Daten von menschlichen Nutzern intuitiv „optimal“ umgesetzt wurde. Bei einem datenlosen Ansatz mittels Nutzersimulation müsste die Bewertung von Meta-Kommunikation explizit modelliert werden, oder ist implizit in der eingangs beispielhaft genannten Kostenfunktion enthalten (-1 pro Turn, 100 pro Dialogerfolg), bei der jeder zusätzlicher Turn durch Meta-Kommunikation leicht bestraft wird, solange er nicht den Aufgabenerfolg sichert.

5 Fazit

Den wenigen Arbeiten zum Trotz sollte überprüft werden, inwieweit sich Strategien zur Meta-Kommunikation mit IRL aus Daten als Kostenfunktionen lernen lassen, und ob dies zu besseren/anderen Policies als eine implizite Bewertung in Kostenfunktionen führt. Dabei ist auch aus praktischer Sicht zu beachten, ob „optimale“ menschliche Dialoge tatsächlich notwendig sind, oder auch bestehende Systeme mit echten, aber Mensch-Maschine-Daten weiter optimiert werden können. Zudem sollte der Erkenntnisgewinn aus Kostenfunktionen zur Meta-Kommunikation über die Nutzung in RL-basierten Sprachinteraktionssystemen hinaus untersucht werden. Können hier bspw. solche Kostenfunktionen oder deren Vereinfachungen auch für Entwickler und Designer hilfreich sein, die den aktuellen Stand der Technik zur Policy-Spezifikation nutzen? Dafür können zuerst freie annotierte Dialogkorpora zur Rekonstruktion der Kostenfunktion durch IRL genutzt werden, um dann RL-Ansätze für die Erzeugung von meta-kommunikativen Systemverhalten zu evaluieren, indem automatisch gelernte Kostenfunktionen mit solchen, die durch Experten spezifiziert wurden, verglichen werden.

Literatur

[1] McGlashan, S., J. Carter, K. Rehor, P. Danielsen, J. Ferrans, D. Burnett, B. Lucas, A. Hunt, B. Porter, und S. Tryphonas: *Voice extensible markup language (VoiceXML) version 2.0*. W3C recommendation, W3C, 2004. URL `http://www.w3.org/TR/2004/REC-voicexml20-20040316/`.

[2] Mnih, V., K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, und M. A. Riedmiller: *Playing atari with deep reinforcement learning*. *CoRR*, abs/1312.5602, 2013. URL `http://arxiv.org/abs/1312.5602. 1312.5602`.

[3] Silver, D., J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. Chen, T. Lillicrap, F. Hui, L. Sifre, G. van den Driessche, T. Graepel, und D. Hassabis: *Mastering the game of Go without human knowledge*. *Nature*, 550(7676), S. 354–359, 2017. doi:10.1038/nature24270.

[4] Peters, J., S. Vijayakumar, und S. Schaal: *Reinforcement learning for humanoid robotics*. In *Proc. third IEEE-RAS International Conference on Humanoid Robots*, S. 1–20. 2003.

[5] Rieser, V. und O. Lemon: *Reinforcement Learning for Adaptive Dialogue Systems*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. doi:10.1007/978-3-642-24942-6.

[6] Watkins, C. J. C. H.: *Learning from Delayed Rewards*. Ph.D. thesis, King's College, 1989.

[7] Young, S., B. Thomson, J. D. Williams, und et al.: *POMDP-based statistical spoken dialogue systems: a review*. *Proc. IEEE*, 101(5), 2013.

[8] NG, A. Y. und S. RUSSEL: *Learning for inverse reinforcement learning*. In *Proc. International Conference on Machine Learning (ICML)*, S. 663–670. 2000.

[9] BOULARIAS, A., H. R. CHINAEI, und B. CHAIBDRAA: *Learning the reward model of dialogue POMDPs from data*. In *Proc. Conference on Neural Information Processing Systems*. 2009.

[10] ASRI, L. E.: *Learning the Parameters of Reinforcement Learning from Data for Adaptive Spoken Dialogue Systems*. Ph.D. thesis, Université de Lorraine, 2016.

[11] LAROCHE, R., G. PUTOIS, P. BRETIER, und B. BOUCHON-MEUNIER: *Hybridisation of expertise and reinforcement learning in dialogue systems*. In *Proc. Interspeech*. 2009.