

ZEITSCHRIFT FÜR VERSICHERUNGSWESEN

UNABHÄNGIGES FACHORGAN FÜR DIE VERSICHERUNGSPRAXIS

Jahrgang 77
Juli 2026

"KI macht den Makler nicht überflüssig – sie macht ihn wieder zum Makler"

267

Ein neuer Ansatz, um große Altbestände endlich besser zu erschließen

MARINA BAUMGARTNER

275

Agentic Context Engineering als Schlüssel für selbstlernende KI

PROF. DR. MATTHIAS BEENKEN / PROF. DR. LUKAS LINNENBRINK

281

AVRG: Herausforderungen für die Branche

Die Reform der Riester-Rente setzt die Lebensversicherer und ihre Vermittler unter Druck.
Ab 1. Januar 2027 ergibt sich eine völlig neue Wettbewerbslandschaft.

GEORG NIEDERREITER / CHRISTIAN LÜBBEN

293

Aufgaben des Maklers und Grenzen der Maklertätigkeit bei Eintritt eines Versicherungsfalles

MICHAEL GRIESE

298

Die Renaissance des Assekuradeurs: Von hanseatischen Traditionen zur Speerspitze spezialisierter Versicherungslösungen

Wirtschaftskommentar	Im Reformstrudel	265
Assekuranz aktuell	"KI macht den Makler nicht überflüssig - sie macht ihn wieder zum Makler"	
	Ein neuer Ansatz, um große Altbestände endlich besser zu erschließen	267
	Eigenständig bleiben mit strategischer Nachfolgeplanung	
	125 Jahre Hagen & Kruse - und was der Maklermarkt von den Hanseaten lernen kann	269
	Transform to Last (7): Transformation in der Kompositversicherung beginnt im Produkt	270
	Neue Run-off-Treiber BU und Riester?	271
	Rentendebatte: Ideologische Abgründe	271
	Riester-Reform: Erfolgsentscheidend ist das Backend	272
	Strategiemeeting Lebensversicherung	272
KI	Marina Baumgartner	
	Agentic Context Engineering als Schlüssel für selbstlernende KI	275
	Markus Stumpe	
	Kundenservice als Vertriebshebel: Wie Versicherer Potenziale mit KI wirklich erschließen	277
Altersvorsorgereformgesetz	Prof. Dr. Matthias Beenken / Prof. Dr. Lukas Linnenbrink	
	AVRG: Herausforderungen für die Branche	
	Die Reform der Riester-Rente setzt die Lebensversicherer und ihre Vermittler unter Druck.	
	Ab 1. Januar 2027 ergibt sich eine völlig neue Wettbewerbslandschaft.	281
Solvency II	Markus Müller	
	Solvency II Reform: neue Anforderungen an Versicherungsgruppen	
	Stärkere Gruppenaufsicht, mehr Governance und neue Anforderungen an Holdingstrukturen	288
Makler	Prof. Dr. Hans-Peter Schwintowski	
	Die neue Maschinenverordnung (EU) 2023/1230 - worüber Makler die Gewerbekunden unterrichten sollten	291
	Georg Niederreiter / Christian Lübben	
	Aufgaben des Maklers und Grenzen der Maklertätigkeit bei Eintritt eines Versicherungsfalles	293
Assekuradeure	Michael Griese	
	Die Renaissance des Assekuradeurs: Von hanseatischen Traditionen zur Speerspitze spezialisierter Versicherungslösungen	298
Personalmanagement	Alexander Hundeshagen	
	Talente gewinnen und halten mit starker Arbeitgeberreputation	301
Gesellschaften	Allianz Partners Deutschland	303
	Continental	303
	Deutsche Rück	304
	Hallesche	304
	Ideal	305
	Mecklenburgische	305
	Signal Iduna	306
	VGH	307
	Volkswahl Bund	308
Impressum	308

Marina Baumgartner

Agentic Context Engineering als Schlüssel für selbstlernende KI

Der Einsatz großer Sprachmodelle (LLMs) und intelligenter Agenten eröffnet neue Potenziale zur Automatisierung und Steuerung komplexer Geschäftsprozesse sowie zur Unterstützung im Bereich Software Engineering. Bei msg life setzen wir bereits heute spezialisierte Agenten ein und entwickeln diese kontinuierlich weiter, um beispielsweise Unternehmenswissen leichter zugänglich zu machen. Als Softwarehersteller sehen wir in der gezielten Weiterentwicklung und Erforschung derartiger intelligenter Lösungen ein erhebliches Zukunftspotenzial. Mit Agentic Context Engineering (ACE) steht eine neue Methodik zur Verfügung, die einen entscheidenden Entwicklungsschritt auf dem Weg zu teil-autonomen KI-Systemen ermöglicht. In diesem Artikel werden die Potenziale von ACE am Beispiel der Versicherungsindustrie erläutert und es wird aufgezeigt, wie dieser Ansatz die Grundlage für zukunftsfähige, selbstlernende KI-Systeme schafft.

Zugriff auf Domänenwissen ist essenziell

Für eine optimale Integration von KI in bestehende Abläufe ist es essenziell, dass Agenten auf Domänenwissen zugreifen können. Dieses Expertenwissen wurde durch aufwendiges Nachtrainieren der Modellgewichte integriert. Im weiteren Verlauf wurde dieser Ansatz zunehmend durch Retrieval-Augmented Generation (RAG) ergänzt oder abgelöst. Mit dieser Methode kann dem Modell externes Wissen gezielt als Kontext bereitgestellt werden. Darauf aufbauend entstanden Agentensysteme, in welchen LLMs nicht nur Text generieren, sondern mithilfe von Werkzeugen selbstständig mit ihrer Umgebung interagieren.

Kontext der Modelle gewinnt an Bedeutung

Mit diesem Wandel gewinnt der Kontext der Modelle zunehmend an Bedeutung: Statt die Modelle neu zu trainieren, wird der Kontext optimiert, also alle Informationsquellen, wie Aufgabenbeschreibung, Beispiele, externes Wissen und Werkzeugdefinitionen. Klassische Methoden der Kontextanpassung, wie Prompt Engineering, erfordern dabei häufig manuelle oder iterative Optimierungsprozesse. Im Gegensatz dazu ermöglicht Agentic Context Engineering¹ Agenten, sich weiterzuentwickeln und Domänenwissen sukzessive aufzubauen. Das ist ein entscheidender Entwicklungsschritt auf dem Weg zu teil-autonomen KI-Systemen.

Funktionsweise von ACE

Insbesondere in dynamischen Branchen wie der Versicherungswirtschaft, die geprägt ist von immateriellem Wissen, verändern sich äußere Rahmenbedin-

gungen kontinuierlich. Mithilfe von ACE können Agenten flexibel auf diese Dynamik reagieren, ohne dass manuell der Kontext angepasst werden muss. Anders als statische Methoden nutzt ACE dynamische, sich weiterentwickelnde Playbooks. In diesen Playbooks sammeln, verfeinern und strukturieren Agenten kontinuierlich bewährte Strategien und erweitern bestehendes Domänenwissen.

Der gesamte Kontext, Strategien, Fakten, Domänenwissen und Werkzeuge, wird in einem fortlaufenden Wissenssystem kombiniert. Dieser Ansatz verhindert, dass wichtige Details verloren gehen („Brevity Bias“, „Context Collapse“) und sorgt dafür, dass Wissen strukturiert gesammelt wird und für zukünftige Entscheidungen verfügbar ist. Dabei folgt ACE einem iterativen Prozess mit drei Agenten-Rollen:

1. *Generator*: Führt Aufgaben aus, interagiert mit der Umgebung und erzeugt detaillierte Reasoning-Trajektorien, also nachvollziehbare Denk-, Entscheidungs- und Handlungsschritte. Diese enthalten sowohl erfolgreiche Strategien als auch typische Fehler.

2. *Reflector*: Analysiert diese Trajektorien und destilliert konkrete, wiederverwendbare Einsichten, d.h., was funktioniert, was nicht, welche Regeln lassen sich ableiten. Die Reflexion kann iterativ verfeinert werden, um die Qualität der Erkenntnisse zu erhöhen.

3. *Curator*: Übersetzt die Einsichten in kompakte „Delta“-Updates, die deterministisch in das bestehende Playbook eingepflegt werden. Durch diese lokalisierten Änderungen wird verhindert, dass der gesamte Kontext ständig neu geschrieben und damit komprimiert oder verfälscht wird.

KI-Agenten entwickeln sich selbstständig weiter

Ein entscheidender Vorteil von ACE ist, dass die Agenten sich selbstständig durch jede Ausführung von Aufgaben weiterentwickeln. Während der Bearbeitung konkreter Aufgaben halten die Agenten fest, welche Strategien in der jeweiligen Situation erfolgreich waren und welche nicht. Auf dieser Basis lernen sie fortlaufend aus ihrem Verhalten: Sie vermeiden wiederkehrende Fehler und verbessern sich aktiv mithilfe von Feedback.

Feedback kann dabei aus verschiedenen Quellen stammen: maschinell, z.B. erfolgreiche oder fehlgeschlagene Code-Ausführung; agentisch, z.B. Rückmeldungen anderer Agenten oder Tools; oder menschlich, z.B. Kommentare, Korrekturen oder Bewertungen von Nutzern und Fachexperten. So vermeiden Agenten wiederkehrende Fehler, übernehmen bewährte Muster in ihr Playbook und erzielen über die Zeit bessere und konsistentere Ergebnisse, bei gleichbleibendem Basismodell.

Nachvollziehbarkeit der Entscheidungen

Ein weiterer Vorteil ist die vollständige Nachvollziehbarkeit der Entscheidungen: Da Reasoning-Schritte, Annahmen und Aktionen im Kontext in Form des Playbooks festgehalten werden, können Menschen den Entscheidungsprozess jederzeit prüfen, verstehen und bei Bedarf eingreifen. Denn ein wesentliches Merkmal

Marina Baumgartner

Data Scientist im Bereich Research & Development
msg life ag

"Agentic Context Engineering markiert einen zentralen Schritt in der Weiterentwicklung KI gestützter Versicherungsprozesse: Weg von statischen Prompts hin zu lernenden Agentensystemen, die ihr Domänenwissen aus realen Interaktionen kontinuierlich ausbauen"

von ACE ist, dass die Playbooks in für Menschen lesbarer Form vorliegen, das heißt, sie enthalten keine numerischen Gewichte oder codierten Repräsentationen, sondern sind als Anleitungen in Klartext formuliert.

Dies ermöglicht eine Art der Aufsicht, bei welcher der Mensch die strategische Kontrolle behält, und die Agenten autonom innerhalb ihrer Grenzen agieren („Human-on-the-loop“)². Dadurch entstehen nachweisbare und transparente agentische Systeme, ohne dass der Mensch laufend eingreifen muss.

In der Softwareentwicklung werden Agenten bereits heute als Programmierhilfe eingesetzt. Ohne ACE reagieren sie zwar bereits auf Fehler (z.B. fehlgeschlagene Tests), vergessen diese gewonnenen Erkenntnisse aber nach Abschluss der Aufgabe wieder. Mit ACE wird derartige Feedback systematisch im Playbook verankert und dauerhafte Strategien wie z.B. Coding Guidelines können abgeleitet werden. Bei einer nächsten, ähnlichen Aufgabe können dann häufig auftretende Fehler- und Feedbackmuster von Beginn an berücksichtigt werden, was sowohl Qualität als auch Geschwindigkeit erhöht und zugleich das Risiko wiederkehrender Fehler reduziert.

Potenziale von Agentic Context Engineering für die Versicherungsindustrie

In der Versicherungsbranche stellen die umfassenden Geschäftsprozesse sowie komplexe Logiken eine große Herausforderung dar. Ein agentisches System, das Kundenanfragen automatisiert bearbeitet, muss zunächst entscheiden, um welche Art von Anliegen es sich handelt, ob alle relevanten Informationen vorliegen und ob ein konkreter Fachprozess sicher angestoßen werden kann. In der Praxis müssen die Agenten dabei mit existierenden Bestands- und Fachanwendungen im Hintergrund agieren, automatisierte Workflows anstoßen - und sie erhalten wiederum Rückmeldungen aus diesen Systemen (z.B. Fehlermeldungen, Validierungsfehler, widersprüchliche Daten).

Mit ACE gehen solche Erfahrungen nicht verloren, sondern werden als wiederkehrende Muster gespeichert. So entstehen nach und nach domänenspezifische „Best Practices“, z.B. für die Dun-

kelverarbeitung von Anfragen. Dadurch werden Prozesse robuster, weil der Agent aus echten Produktionsfällen lernt. Gleichzeitig steigt die Transparenz, weil wichtige Entscheidungsgrundlagen dauerhaft im Kontext mitgeführt werden. Im Versicherungssektor verbessert ACE so die Interpretierbarkeit von automatisierter Bearbeitung von Benutzeranfragen durch Agenten.

Fazit

Agentic Context Engineering markiert einen zentralen Schritt in der Weiterentwicklung KI gestützter Versicherungsprozesse: Weg von statischen Prompts hin zu lernenden Agentensystemen, die ihr Domänenwissen aus realen Interaktionen kontinuierlich ausbauen. Indem Erfahrungen aus täglichen Interaktionen, Systemrückmeldungen und Prozessfehlern strukturiert im Kontext gesammelt werden, entsteht ein wachsendes, nachvollziehbares Playbook versicherungsspezifischer Best Practices. So können nicht nur Automatisierungsgrade und Dunkelverarbeitungsquoten erhöht werden, sondern zugleich Transparenz und Qualität verbessert werden.

ACE bildet damit eine wichtige Grundlage, um schrittweise zu robusten, selbstlernenden KI-Systemen in der Versicherungswirtschaft überzugehen. Darüber hinaus birgt dieser Ansatz auch in anderen Branchen das Potenzial, KI-Systeme nachhaltig und domänenspezifisch weiterzuentwickeln.

¹ Q. Zhang, C. Hu, S. Upasani, B. Ma, F. Hong, V. Kamanuru, J. Rainton, C. Wu, M. Ji, H. Li, U. Thakker, J. Zou, and K. Olukotun, "Agentic Context Engineering: Evolving Contexts for Self-Improving Language Models," 2025.

² J. Fischer, C. Greenhalgh, W. Jiang, S. Ramchurn, F. Wu, and T. Rodden, "In-the-loop or on-the-loop? Interactional arrangements to support team coordination with a planning agent," *Concurrency and Computation: Practice and Experience*, vol. 33, doi: 10.1002/cpe.4082.