

Zusammenfassung

Das Thema dieser Arbeit ist die Erläuterung und Anwendung der agilen Softwareentwicklung mit dem Test-Driven Development (TDD) für die Entwicklung einer Conversational AI.

Diese Arbeit wurde im Rahmen eines Projekts der Firma INNOQ erstellt, daher wird der Schwerpunkt auf die Anwendung mit dem Test-Driven Development und einer den Voraussetzungen entsprechende Implementierung des Anwendungsfalls gelegt. Die Anpassung von Softwareentwicklungsmethoden an Machine Learning (ML) ist ökonomisch sinnvoll, da ML immer häufiger in Anwendungen Verwendung findet (vgl. [?]).

Die Entwicklung von Software stellt oft einen großen finanziellen Aufwand für Unternehmen dar. Daher besteht in der Softwareentwicklung ein Bedarf an standardisierten Vorgehensweisen, wie dem TDD, da diese eine simple, strukturierte und zügige Bearbeitung der Projekte erleichtern. Die Problematik hierbei ist im Wesentlichen, dass das TDD für die Entwicklung von gleichbleibenden, eher deterministischen Systemen ausgelegt ist. Dem gegenüber stehen die ML-Systeme, welche durch die Notwendigkeit des regelmäßigen Trainierens der Modelle eine sich verändernde und statistische Natur aufweisen (vgl. [?]).

Für die Entwicklung einer Conversational AI wurde sich entschieden, da dies zum einen von der praktischen Anwendung nahegelegt ist und zum anderen bietet die Entdeckung von Transformern mit einem Aufmerksamkeitsmechanismus eine kürzliche, sehr interessante Neuerung in diesem Bereich (vgl. [?]). Des Weiteren stellt der Einsatz von Conversational AI durch die Automatisierung des Kundensupports eine große finanzielle Erleichterung für Unternehmen dar (vgl. [?]).

In dieser Arbeit wird daher eine Vorgehensweise des TDD vorgestellt, um eine Conversational AI zu implementieren. Hierbei müssen sowohl die Voraussetzungen an die Conversational AI als auch die Unterschiede und Besonderheiten des Machine Learnings beachtet werden, sodass das TDD zu einer validen Lösung führt. Bei der Vorgehensweise wurde, unter Berücksichtigung der Unterschiede, ein Fokus darauf gelegt, die Struktur des TDD zu erhalten, um somit auch möglichst viele der Vorteile bei der Entwicklung beizubehalten.