

Machine Learning zum Cocktailmixen

Florian Grassl

Fachsemester: 5

Betreuer: Moritz Fuchsloch, Kim Glumann
Institut für Strahlwerkzeuge (IFSW)
Universität Stuttgart

Stuttgart, 7. März 2025

Inhaltsverzeichnis

1	Einleitung	2
1.1	Motivation	2
1.2	Stand der Technik	2
2	Hauptteil	3
2.1	Methodik	3
2.1.1	Datenvorverarbeitung	3
2.1.2	Modellierung	3
3	Zusammenfassung und Ausblick	5

Kapitel 1

Einleitung

1.1 Motivation

Seit Veröffentlichung von GPT-4 im November 2022 ist das öffentliche Interesse an Künstlicher Intelligenz (KI) und Machine Learning (ML) enorm gestiegen. Cocktails bieten ein ideales Anwendungsszenario für ML, da sie über einen großen Parameter-raum verfügen und die menschliche Geschmackswahrnehmung sehr komplex und nicht-linear ist. Diese Arbeit demonstriert, wie ML eingesetzt werden kann, um automatisch Cocktails zu mischen, indem komplexe Geschmacksprofile analysiert und in individuelle Rezepturen übersetzt werden.

1.2 Stand der Technik

Im Bereich des maschinellen Lernens dominieren aktuell Transformer-Modelle aufgrund ihrer Fähigkeit, komplexe Zusammenhänge in umfangreichen Datensätzen effektiv zu erfassen [VSP⁺17]. Neben Transformern werden auch andere Modelle wie CNNs (für Bilder), RNNs (für sequentielle Daten) und GANs (zur Daten-Generierung) genutzt [GPAM⁺14]. Für diese Arbeit wurde jedoch bewusst ein Autoencoder gewählt, da dieser robust bei kleinen Datensätzen ist und besonders gut zur Rekonstruktion latenter Muster geeignet ist [GBC16]. Neuere Trends beinhalten hybride Modelle und Reinforcement Learning, um Nutzerfeedback einzubeziehen [SB18].

Kapitel 2

Hauptteil

2.1 Methodik

2.1.1 Datenvorverarbeitung

Zu Beginn wurde eine JSON-basierte Datenbank angelegt, die alle relevanten Zutaten sowie ihre zugehörigen Geschmacksprofile (süß, sauer, bitter, fruchtig, würzig) enthält. Diese Datenbank dient als Grundlage für sämtliche Analysen und Optimierungen. Um eine effiziente Datenverarbeitung zu ermöglichen, wurde der JSON-Datensatz zunächst mithilfe von Python-Bibliotheken wie JSON und pandas eingelesen und in ein Dictionary überführt. Anschließend wurde das Dictionary mit der Bibliothek pandas in ein DataFrame konvertiert, um eine strukturierte Datenverarbeitung zu ermöglichen. Da unterschiedliche Zutaten teils stark voneinander abweichende Wertebereiche in ihren Geschmacksprofilen aufweisen können, wurden die Daten mittels Standardisierung normalisiert. Hierfür wurde der StandardScaler aus der Bibliothek scikit-learn verwendet, um die Daten auf einen Mittelwert von null und eine Standardabweichung von eins zu skalieren. Abschließend erfolgte die Aufteilung der normalisierten Daten in Trainings- und Testsets im Verhältnis 80/20, um das Modell auf Generalisierung zu prüfen und Overfitting vorzubeugen. Dies ermöglicht eine zuverlässige Bewertung der Modellqualität und Robustheit gegenüber unbekannten Daten.

2.1.2 Modellierung

Die Entwicklung des Systems basiert auf einer Kombination aus Machine-Learning-Techniken und mathematischer Optimierung.

Datenbasis und Nutzerprofil

Zunächst werden Zutaten mit Geschmacksprofilen und nutzerbezogene Fragen aus einer JSON-Datei geladen, um ein individuelles, gewichtetes Nutzerprofil zu erstellen.

Der Nutzer durchläuft hierfür einen Fragebogen. Jede Antwort wird zu einem fünfdimensionalen Vektorprofil aggregiert:

$$\mathbf{p}_{\text{user}} = \frac{1}{N} \sum_{i=1}^N \mathbf{q}_i,$$

wobei \mathbf{q}_i das Profil der i -ten Frage ist und N die Gesamtanzahl beantworteter Fragen. Zusätzlich entscheidet der Nutzer, ob der Cocktail alkoholisch sein soll, und legt die maximale Zutatenanzahl k fest.

Autoencoder zur Profilrekonstruktion

Das zentrale Element des Systems ist ein Autoencoder, der das Nutzerprofil komprimiert und rekonstruiert. Der Autoencoder besteht aus:

- **Encoder:** Komprimiert das Nutzerprofil auf einen dreidimensionalen Latent-Space:

$$\mathbf{z} = \text{Encoder}(\mathbf{p}_{\text{user}})$$

- **Decoder:** Rekonstruiert das Profil aus dem Latent-Space:

$$\mathbf{p}_{\text{rekon}} = \text{Decoder}(\mathbf{z})$$

Das Training erfolgt unüberwacht anhand der Geschmacksdaten mittels Loss-Funktion:

$$\mathcal{L} = \|\mathbf{p}_{\text{rekon}} - \mathbf{p}_{\text{user}}\|^2$$

Optimale Zutatenmischung

Ausgehend von $\mathbf{p}_{\text{rekon}}$ wird die optimale Mischung per Least-Squares berechnet:

$$\min_{\mathbf{w}} \|\mathbf{A}\mathbf{w} - \mathbf{p}_{\text{rekon}}\|^2$$

wobei:

- $\mathbf{A} \in \mathbb{R}^{5 \times M}$: Matrix der Zutatenprofile,
- $\mathbf{w} \in \mathbb{R}^M$: Gewichtsvektor der Zutaten.

Negative Gewichte werden eliminiert, anschließend werden nur die Top- k -Zutaten verwendet und deren Anteile auf ein Gesamtvolumen von 200 ml normiert.

Kapitel 3

Zusammenfassung und Ausblick

Die in dieser Arbeit vorgestellte Implementierung zeigt eine modulare, gut strukturierte Softwarelösung, die erfolgreich Methoden des maschinellen Lernens und der mathematischen Optimierung kombiniert, um individuelle Cocktailrezepte automatisiert zu generieren. Der entwickelte Ansatz zeichnet sich besonders durch seine Flexibilität und Anpassungsfähigkeit aus, wodurch er leicht auf neue Zutaten oder geänderte Anforderungen erweitert werden kann. Ein entscheidender Vorteil des Systems ist seine Skalierbarkeit. Die modulare Architektur erlaubt es, weitere Komponenten, wie etwa neue Modelle zur Geschmacksanalyse oder zusätzliche Optimierungsverfahren, mit geringem Aufwand zu integrieren. Trotz dieser Stärken bestehen jedoch auch einige Herausforderungen. Vor allem die Qualität und Vollständigkeit der zugrundeliegenden Datenbasis hat erheblichen Einfluss auf die Präzision und Zuverlässigkeit der generierten Rezeptvorschläge. Fehlende oder schlecht gewählte Zutatenprofile können zu weniger zufriedenstellenden Ergebnissen führen. Um diesen Herausforderungen künftig besser zu begegnen, könnten mehrere Erweiterungen sinnvoll sein. Beispielsweise wäre die Implementierung einer benutzerfreundlichen, grafischen Benutzeroberfläche denkbar, die die Interaktion mit dem System erheblich vereinfacht und den Nutzern die Möglichkeit bietet, unmittelbar Feedback zu geben. Darüber hinaus könnte der Einsatz komplexerer Modelle wie Transformer-Architekturen oder hybrider Ansätze, die traditionelle regelbasierte Systeme mit maschinellen Lernverfahren kombinieren, die Genauigkeit der Rezeptgenerierung weiter steigern. Zusätzlich könnte das Modell durch weitere Geschmacksdimensionen ergänzt werden, um die individuellen Präferenzen der Nutzer noch differenzierter abzubilden. Auch eine Erweiterung um Nutzerfeedback und adaptive Lernmechanismen wie Reinforcement Learning könnte perspektivisch die Genauigkeit und Personalisierung des Systems nachhaltig verbessern. Solche Ansätze würden nicht nur die Qualität der generierten Rezepte verbessern, sondern auch das Potenzial des Systems im kommerziellen Einsatz, etwa in Bars oder gastronomischen Betrieben, deutlich erhöhen.

Literaturverzeichnis

- [GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT press, 2016.
- [GPAM⁺14] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014.
- [SB18] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2018.
- [VSP⁺17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Selbstständigkeitserklärung

Ich versichere, dass ich diese Arbeit selbstständig verfasst habe.

Stuttgart, 7. März 2025

(Florian Grassl)