edv-buchversand.de

Language Models kompakt

Praxisorientierte Sprachmodellierung mit PyTorch

» Hier geht's direkt zum Buch

DAS VORWORT

Dieses Vorwort wird Ihnen von www.edv-buchversand.de zur Verfügung gestellt.

Einführung

Mein Interesse an Text begann in den späten 1990er-Jahren, als ich als Teenager dynamische Websites mit Perl und HTML erstellte. Diese frühen Erfahrungen mit der Programmierung und der Organisation von Text in strukturierten Formaten weckten meine Faszination dafür, wie man Text verarbeiten und transformieren könnte. Die Herausforderung der Verarbeitung von Texten und des Textverständnisses brachte mich dazu, komplexere Anwendungen zu erforschen. Dazu gehörte auch die Entwicklung von Chatbots, die die Anforderungen der Benutzerinnen und Benutzer verstehen und erfüllen können.

Fasziniert hat mich die Aufgabe, die Bedeutung von Wörtern zu extrahieren. Die Komplexität dieser Aufgabe hat meine Entschlossenheit, sie zu »knacken«, nur noch verstärkt. Dabei habe ich alle mir zur Verfügung stehenden Werkzeuge eingesetzt – von regulären Ausdrücken und Skriptsprachen bis hin zu Textklassifizierern und Modellen zum Erkennen benannter Entitäten.

Das Aufkommen großer Sprachmodelle (*Large Language Models*, LLMs) hat alles verändert. Zum ersten Mal konnten sich Computer fließend mit uns unterhalten und verbale Anweisungen mit bemerkenswerter Genauigkeit befolgen. Doch wie bei jedem Werkzeug stößt auch ihre immense Leistung an Grenzen. Einige sind leicht zu erkennen, andere Einschränkungen wiederum sind subtiler und erfordern ein hohes Maß an Fachwissen, um sie richtig zu handhaben. Wenn man versucht, einen Wolkenkratzer zu errichten, ohne die dafür erforderlichen Werkzeuge vollständig zu verstehen, wird das Ergebnis nur ein Haufen Beton und Stahl sein. Das Gleiche gilt für Sprachmodelle. Will man umfangreiche Texte verarbeiten oder zuverlässige Produkte für zahlende Benutzer erstellen, sind Genauigkeit und Wissen unabdingbar – Rätselraten ist einfach keine Option.

Für wen dieses Buch gedacht ist

Dieses Buch habe ich für diejenigen geschrieben, die wie ich von der Herausforderung fasziniert sind, Sprache mithilfe von Maschinen zu verstehen. Sprachmodelle sind im Grunde genommen nichts anderes als mathematische Funktionen. Allerdings wird ihr wahres Potenzial in der Theorie nicht voll ausgeschöpft – man muss sie implementieren, um zu sehen, wie leistungsfähig sie sind und wie ihre

Fähigkeiten mit zunehmender Skalierung wachsen. Deshalb habe ich mich entschlossen, dieses Buch praxisorientiert zu gestalten.

Dieses Buch richtet sich an Softwareentwicklerinnen und Softwareentwickler, Data Scientists, Machine Learning Engineers und alle, die sich für Sprachmodelle interessieren. Unabhängig davon, ob Ihr Ziel darin besteht, bestehende Modelle in Anwendungen zu integrieren oder eigene Modelle zu trainieren, finden Sie neben theoretischen Grundlagen auch praktische Anleitungen.

Angesichts des kompakten Formats sollten die Leserinnen und Leser dieses Buchs einige Voraussetzungen mitbringen. Dazu gehört auch Programmiererfahrung, da alle praktischen Beispiele mit Python realisiert sind.

Zwar ist es von Vorteil, mit PyTorch und Tensoren – den grundlegenden Datentypen von PyTorch – vertraut zu sein, doch ist das nicht unbedingt erforderlich. Für Einsteiger in diese Tools bietet das englischsprachige Wiki des Buchs (*https://thelmbook.com/wiki*) eine kompakte Einführung mit Beispielen und weiterführenden Links für Lehrzwecke. Dieses Wiki-Format stellt sicher, dass der Inhalt aktuell bleibt und die Fragen der Leser auch nach der Veröffentlichung beantwortet werden.

Kenntnisse in höherer Mathematik sind ebenfalls hilfreich, aber Sie müssen sich nicht an jedes Detail erinnern und auch keine Erfahrungen in Machine Learning mitbringen. Das Buch führt Konzepte systematisch ein, es beginnt mit den Notationen, Definitionen und fundamentalen Vektor- und Matrixoperationen. Von dort aus geht es über einfache neuronale Netze zu fortgeschritteneren Themen. Die mathematischen Konzepte werden intuitiv dargestellt – mit klaren Diagrammen und Beispielen, die das Verständnis erleichtern.

Was dieses Buch nicht ist

Der Schwerpunkt dieses Buchs liegt auf dem Verständnis und der Implementierung von Sprachmodellen. Dagegen wird Folgendes *nicht* behandelt:

- *Training im großen Maßstab:* Mit diesem Buch lernen Sie *nicht*, wie man umfangreiche Modelle auf verteilten Systemen trainiert oder wie man die Trainingsinfrastruktur verwaltet.
- *Produktionsdeployment:* Themen wie Modell-Serving, API-Entwicklung, Skalierung für hohen Traffic, Monitoring und Kostenoptimierung werden *nicht* behandelt. Die Codebeispiele konzentrieren sich darauf, die Konzepte zu vermitteln, und eignen sich nicht für die Produktion.
- *Enterprise-Anwendungen:* Dieses Buch führt Sie *nicht* durch das Erstellen kommerzieller LLM-Anwendungen, den Umgang mit Benutzerdaten oder die Integration in bestehende Systeme.

Wenn Sie daran interessiert sind, die mathematischen Grundlagen von Sprachmodellen zu erlernen, ihre Funktionsweise zu verstehen, die Kernkomponenten in eigener Regie zu implementieren oder zu lernen, mit LLMs effektiv zu arbeiten,

ist dieses Buch für Sie das richtige. Möchten Sie jedoch in erster Linie Modelle in der Produktion bereitstellen oder skalierbare Anwendungen erstellen, sollten Sie zusätzlich zu diesem Buch andere Ressourcen heranziehen.

Wie dieses Buch aufgebaut ist

Um dieses Buch interessant zu gestalten und das Verständnis zu vertiefen, habe ich mich dafür entschieden, die Sprachmodellierung als Ganzes zu diskutieren, einschließlich der Ansätze, die in der modernen Literatur oftmals übersehen werden. Während Transformer-basierte LLMs im Rampenlicht stehen, bleiben jedoch frühere Ansätze wie zählbasierte Methoden und rekurrente neuronale Netze (RNNs) für einige Aufgaben weiterhin effektiv.

Die Mathematik der Transformer-Architektur von Grund auf zu lernen, mag für einen Neueinsteiger überwältigend erscheinen. Indem ich diese grundlegenden Methoden wieder aufgreife, möchte ich die Intuition und das mathematische Verständnis der Lesenden schrittweise aufbauen, sodass sich der Übergang zu modernen Transformer-Architekturen eher wie eine natürliche Entwicklung anfühlt als wie ein einschüchternder Sprung.

Das Buch ist in sechs Kapitel gegliedert, die von den Grundlagen bis zu fortgeschrittenen Themen reichen:

- **Kapitel 1** behandelt die Grundlagen des Machine Learning, darunter Schlüsselkonzepte wie KI, Modelle, neuronale Netze und Gradientenabstieg. Auch wenn Sie mit diesen Themen bereits vertraut sind, bietet das Kapitel wichtiges Basiswissen für das Verständnis von Sprachmodellen.
- **Kapitel 2** führt in die Grundlagen der Sprachmodellierung ein und untersucht Methoden der Textdarstellung, etwa Bag-of-Words und Word-Embeddings, sowie zählbasierte Sprachmodelle und Bewertungstechniken.
- **Kapitel 3** konzentriert sich auf rekurrente neuronale Netze (RNNs) und beschäftigt sich mit Implementierung, Training und Anwendung als Sprachmodelle.
- **Kapitel 4** untersucht detailliert die Transformer-Architektur, einschließlich Schlüsselkomponenten wie Self-Attention, Positions-Embeddings und praktischer Implementierung.
- **Kapitel 5** befasst sich mit großen Sprachmodellen (Large Language Models, LLMs), erörtert die Bedeutung des Datenumfangs, beschreibt Techniken zum Finetuning, nennt praktische Anwendungen und stellt wichtige Überlegungen zu Halluzinationen, Copyright und Ethik an.
- **Kapitel 6** schließt mit weiterführender Lektüre zu fortgeschrittenen Themen wie Mixture of Experts (MoE), Modellkomprimierung, präferenzbasierte Ausrichtung sowie Vision Language Models und gibt Hinweise zum weiteren Lernen.

Die meisten Kapitel enthalten funktionsfähige Codebeispiele, die Sie unverändert ausführen und auch modifizieren können. Im Buch selbst werden zwar nur die we-

sentlichen Teile des Codes wiedergegeben, doch steht Ihnen der vollständige Code als Jupyter Notebook auf der Website des Buchs (unter *http://thelmbook.com/wiki*) zur Verfügung, wobei in den jeweiligen Abschnitten auf die Notebooks verwiesen wird. Der gesamte Code in den Notebooks bleibt mit den neuesten stabilen Versionen von Python, PyTorch und anderen Bibliotheken kompatibel.

Die Notebooks laufen auf Google Colab, das derzeit kostenlosen Zugang zu Rechenressourcen wie GPUs und TPUs bietet. Diese Ressourcen stehen jedoch nicht garantiert zur Verfügung und haben Nutzungsgrenzen, die variieren können. Einige Beispiele benötigen erweiterten GPU-Zugriff, was eventuell bis zu seiner Verfügbarkeit mit Wartezeiten verbunden ist. Sollte sich die kostenlose Version als zu einschränkend erweisen, können Sie mit der Pay-as-you-go-Option von Colab Rechenguthaben für einen zuverlässigen GPU-Zugang erwerben. Für nordamerikanische Verhältnisse sind diese Guthaben relativ günstig, doch können die Kosten je nach Ihrem Standort deutlich höher ausfallen.

Wenn Sie mit der Linux-Befehlszeile vertraut sind, bieten die GPU-Cloud-Dienste eine weitere Option in Form von virtuellen Computern, die auf Zeitbasis bezahlt werden. Das Wiki zum Buch enthält aktuelle Informationen über kostenlose und kostenpflichtige Notebook- und GPU-Mietdienste.

Begriffe und Blöcke in `Schreibmaschinenschrift` bezeichnen Code, Codefragmente oder Ergebnisse der Codeausführung. In *Kursivschrift* angegebene Begriffe heben Fachbegriffe sowie gelegentlich Schritte in Algorithmen hervor.

In diesem Buch verwenden wir `pip3`, um sicherzustellen, dass die Pakete für Python 3 installiert sind. Auf den meisten modernen Systemen können Sie stattdessen `pip` verwenden, wenn das System bereits für Python 3 eingerichtet ist.

Beispiele und Wiki zum Buch

Die Codebeispiele finden Sie zum Herunterladen unter *https://github.com/aburkov/theLMbook*.

Zusätzliche englischsprachige Informationen, Tutorials und Errata finden Sie im Wiki zum Buch unter *http://thelmbook.com/wiki*.

Verwenden von Codebeispielen

bei Ihrer Arbeit helfen. Ganz allgemein gilt: Wenn in diesem Buch Beispielcode angeboten wird, können Sie ihn in Ihren Programmen und Dokumentationen verwenden. Sie müssen sich dafür nicht unsere Erlaubnis einholen, es sei denn, Sie reproduzieren einen großen Teil des Codes. Schreiben Sie zum Beispiel ein Programm, das mehrere Teile des Codes aus diesem Buch benutzt, brauchen Sie keine Erlaubnis. Verkaufen oder vertreiben Sie Beispiele aus O'Reilly-Büchern, brauchen Sie eine Erlaubnis. Beantworten Sie eine Frage, indem Sie dieses Buch und Beispielcode daraus zitieren, brauchen Sie keine Erlaubnis. Binden Sie einen

großen Anteil des Beispielcodes aus diesem Buch in die Dokumentation Ihres Produkts ein, brauchen Sie eine Erlaubnis.

Erwähnung, verlangen sie aber nicht. Eine Erwähnung enthält üblicherweise Titel, Autor, Verlag und ISBN, zum Beispiel: »Language Models kompakt von Andriy Burkov, O'Reilly 2025, ISBN 978-3-96009-274-2.«

Falls Sie befürchten, zu viele Codebeispiele zu verwenden oder die oben genannten Befugnisse zu überschreiten, kontaktieren Sie uns unter *kommentar@oreilly.de.*

Danksagungen

Die hohe Qualität dieses Buchs wäre ohne ehrenamtliche Redakteure nicht möglich. Insbesondere danke ich Erman Sert, Viet Hoang Tran Duong, Alex Sherstinsky, Kelvin Sundli und Mladen Korunoski für ihre systematischen Beiträge.

Mein Dank gilt auch Alireza Bayat Makou, Taras Shalaiko, Domenico Siciliani, Preethi Raju, Srikumar Sundareshwar, Mathieu Nayrolles, Abhijit Kumar, Giorgio Mantovani, Abhinav Jain, Steven Finkelstein, Ryan Gaughan, Ankita Guha, Harmanan Kohli, Daniel Gross, Kea Kohv, Marcus Oliveira, Tracey Mercier, Prabin Kumar Nayak, Saptarshi Datta, Gurgen R. Hayrapetyan, Sina Abdidizaji, Federico Raimondi Cominesi, Santos Salinas, Anshul Kumar, Arash Mirbagheri, Roman Stanek, Jeremy Nguyen, Efim Shuf, Pablo Llopis, Marco Celeri, Tiago Pedro und Manoj Pillai für ihre Hilfe.

Wenn Sie sich zum ersten Mal mit Sprachmodellen beschäftigen, beneide ich Sie ein wenig – es ist wirklich magisch, zu entdecken, wie Maschinen lernen, die Welt durch natürliche Sprache zu verstehen.

Ich hoffe, dass Sie dieses Buch genauso gerne lesen, wie ich es geschrieben habe. Nehmen Sie jetzt Ihren Tee oder Kaffee und lassen Sie uns beginnen!