



Technische
Hochschule
Nürnberg

Fakultät Informatik

**Vergleichsanalyse von multimodalen Large
Language Models und einer
OCR/YOLO-Pipeline zur
Dokumentenklassifikation für die
Bundesagentur für Arbeit**

Bachelorarbeit im Studiengang Informatik

vorgelegt von

Lukas Müller

Matrikelnummer 3698673

Erstgutachter: Prof. Dr. Natalie Kiesler

Zweitgutachter: Prof. Dr. Korbinian Riedhammer

© 2026

Dieses Werk einschließlich seiner Teile ist **urheberrechtlich geschützt**. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung des Autors unzulässig und strafbar. Das gilt insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen sowie die Einspeicherung und Verarbeitung in elektronischen Systemen.

Kurzdarstellung

Kurze Zusammenfassung der Arbeit, höchstens halbe Seite. Deutsche Fassung auch nötig, wenn die Arbeit auf Englisch angefertigt wird.

Inhaltsverzeichnis

1 Einleitung	1
1.1 Motivation	1
1.2 Problemstellung	1
1.3 Zielsetzung	2
2 Theoretische Grundlagen	3
2.1 Dokumentenarten	3
2.1.1 KG5b	3
2.1.2 Ausbildungsvertrag	3
2.1.3 Sonstige Dokumente	4
2.2 Fachliche Grundlagen	4
2.2.1 Der Ist-Zustand	4
2.2.2 Rahmenbedingungen und Infrastruktur	6
2.3 Vision Language Models (VLMs)	7
2.4 Vorstellung der Modelle	8
2.4.1 Pixtral-12B	8
2.4.2 Qwen-2.5-VL	9
2.4.3 Benchmarks	9
2.5 Parameter-Efficient Fine-Tuning	9
2.6 Verwandte Arbeiten	10
3 Methodik und Implementierung	11
3.1 Modellierung des Dokumentenbestands als Grundlage der Evaluation	12
3.1.1 Preprocessing und Darstellung der Dokumentenarten als JSON-Objekte	12
3.1.2 Test- und Validierungsdatensatz	13
3.2 Evaluation der generierten JSON-Objekte	14
3.2.1 Validierung des VLM-Ouputs	14
3.2.2 Vergleichslogik der unterschiedlichen JSON-Felder	14
3.2.3 Bewertung der Klassifikation und der Information Extraction	15
3.2.4 Messung von Latenz, Energieverbrauch und VRAM-Nutzung	15
3.3 Auswahl des Basismodells	16
3.3.1 Prompt-Design	16
3.3.2 Modellauswahl	16

3.4 Fine-Tuning des Basismodells	17
3.4.1 Trainingsdatensatz	17
3.4.2 Optimierung des Fine-Tuning	17
4 Ergebnisse	19
4.1 Ergebnisse der YOLO/OCR-Pipeline gegenüber der Basismodelle	19
4.2 Ergebnisse des weitertrainierten Modells gegenüber dem größeren Basismodell . .	22
4.3 Ressourcenverbrauch der VLMs gegenüber der YOLO/OCR-Pipeline	22
5 Diskussion	25
6 Zusammenfassung	27
Abbildungsverzeichnis	29
Tabellenverzeichnis	31
List of Listings	33
Literaturverzeichnis	35
Glossar	37

Kapitel 1

Einleitung

1.1 Motivation

Die öffentliche Verwaltung in Deutschland steht vor einer großen Herausforderung. Durch den demografischen Wandel verliert der öffentliche Sektor eine hohe Anzahl an erfahrenen Sachbearbeitern, während der Anspruch der Bürger steigt. Dieser Wandel wird ohne Digitalisierung und Automatisierung der Prozesse kaum abzufangen sein.

Auch in der Bundesagentur für Arbeit (BA) macht sich dieser Wandel, hin zu mehr Digitalisierung, bemerkbar. Ein Beispiel hierfür ist die Bearbeitung von Kindergeldanträgen. Mit einem jährlichen Aufkommen von mehreren Millionen Anträgen[Arbe 24] und saisonalen Spitzen, kommen hier die Sachbearbeiter an ihre Grenzen.

Um sich dieser Herausforderung zu stellen, wurde bereits im Oktober 2025 ein teilautomatisiertes System produktiv gesetzt. Hierbei wird sich auf die Kindergeldanträge von volljährigen Auszubildenden konzentriert. Es werden Dokumente klassifiziert und mithilfe von YOLO- und OCR-Modellen Inhalte aus hochgeladenen Dokumenten erkannt und dem Sachbearbeiter Vorschläge angezeigt. Diese aktuelle YOLO/OCR-Pipeline wertet die Bearbeitung der Kindergeldanträge für die Sachbearbeiter bereits auf, stößt jedoch an ihre Grenzen.

Aufgrund der rasanten Entwicklung im Bereich der Vision Language Models (VLMs) werden diese zunehmend interessanter. In der vorliegenden Arbeit soll evaluiert werden, inwiefern ein solches Modell Potenzial hat, die YOLO/OCR-Pipeline zu ersetzen.

1.2 Problemstellung

Seitdem die aktuelle Pipeline im operativen Einsatz ist, zeigen sich verschiedene Herausforderungen.

Das Hauptproblem ist die hohe Varianz der hochgeladenen Dokumente, insbesondere der Ausbildungsverträge. Die verschiedenen Firmen und Kammern nutzen alle unterschiedliche

Layouts, wodurch das Training der Modelle komplex ist. Es ist schwierig, die Varianz in den Trainingsdaten abzubilden, zudem es zeitaufwendig ist, eine solche Menge an Trainingsdaten bereitzustellen.

Ein weiteres Problem ist die Qualität der hochgeladenen Dokumente. Die Dokumente, die relevant für den Kindergeldantrag bei volljährigen Auszubildenden sind, benötigen in allen Fällen eine Unterschrift. Dadurch werden die Dokumente zwingenderweise ausgedruckt, ausgefüllt und unterschrieben, um schlussendlich wieder eingescannt zu werden. Dieser Prozess, den man auch Medienbruch nennt, hat eine schlechte Qualität zur Folge.

Angesichts der Verwendung von vielen unterschiedlichen Modellen ist die Wartung des Systems aufwändig. Eine Anpassung an neue Gegebenheiten ist ein zeitaufwendiger Prozess. Schon kleine Änderungen an der Laufzeitumgebung, wie zum Beispiel eine neue Python-Version, erfordern meistens ein erneutes Training der Modelle.

1.3 Zielsetzung

Das Ziel dieser Bachelorarbeit ist die Entwicklung und Evaluation eines prototypischen Systems zur teilautomatisierten Klassifikation und Informationsextraktion von Dokumenten. Angesichts der hohen Anzahl manuell bearbeiteter Anträge soll untersucht werden, inwieweit VLMs diesen Prozess effizienter gestalten können.

Es wird eine Pipeline implementiert, die in der Lage ist, gescannte Dokumente als Bilddaten zu verarbeiten. Das System soll den Dokumententyp eigenständig erkennen und definierte Inhalte, darunter handschriftliche Merkmale wie Unterschriften oder Stempel, in ein standardisiertes Format überführen.

Ein weiteres Ziel ist der Vergleich unterschiedlicher Modellansätze hinsichtlich ihrer Extraktionsperformance und Effizienz. Hierbei wird ein kleineres, domänenpezifisch nachtrainiertes Modell gegen leistungsstärkere Modelle mit höherer Parameteranzahl antreten. Es soll ermittelt werden, ob durch Fine-Tuning mit einem begrenzten Datensatz vergleichbare oder bessere Ergebnisse erzielt werden können als durch den Einsatz größerer Basismodelle.

Diese Arbeit konzentriert sich auf die Machbarkeit und die Evaluation der Modelle anhand eines Testdatensatzes. Die Entwicklung zielt auf einen funktionsfähigen Prototyp ab, der lokal betrieben wird. Eine vollständige Integration in das bestehende operative Fachverfahren ist nicht Gegenstand dieser Arbeit.

Kapitel 2

Theoretische Grundlagen

2.1 Dokumentarten

Im Rahmen der Kindergeldbeantragung für Auszubildende sind verschiedene Nachweise gültig. Zu den anerkannten Dokumententypen zählen der offizielle Vordruck der Bundesagentur für Arbeit (KG5b)[Arbe22] sowie Ausbildungsverträge. Die relevanten Informationen aus diesen Dokumenten sind von der fachlichen Seite vorgegeben. Zusätzlich laden Kunden häufig weitere Unterlagen, wie beispielsweise Schulbescheinigungen, im Portal hoch. Da diese für den Kindergeldantrag nicht im Fokus stehen, werden sie im Folgenden unter der Kategorie „Sonstiges“ zusammengefasst.

2.1.1 KG5b

Das Formular KG5b ist, wie bereits erwähnt, ein offizielles Dokument der Bundesagentur für Arbeit, welches als Bescheinigung der Ausbildungsstätte dient. Volljährige Kinder weisen damit gegenüber der Familienkasse den Status ihrer Ausbildung nach, was die Voraussetzung für den weiteren Kindergeldbezug ist.

Abbildung 2.1 zeigt ein exemplarisch ausgefülltes KG5b-Formular mit Markierung der für den Kindergeldantrag relevanten Felder.

2.1.2 Ausbildungsvertrag

Im Gegensatz zu den standardisierten KG5b-Formularen weisen Ausbildungsverträge eine deutlich höhere Varianz auf. Dies ist auf die Vielzahl unterschiedlicher zuständiger Stellen (z. B. Industrie- und Handelskammern, Handwerkskammern, Ärztekammern) und Firmen zurückzuführen, die jeweils ein eigenes Layout definieren. Die Vielfalt der Dokumentenstruktur reicht dabei von formularbasierten Layouts bis hin zu unstrukturierten Fließtexten.

In der folgenden Abbildung 2.2 ist ein synthetischer Ausbildungsvertrag der Industrie- und Handelskammer[Hand25] abgebildet, in dem die relevanten Felder markiert sind.

Erklärung zum Ausbildungsverhältnis

Angaben zum Kind

Angaben zum Ausbildungsverhältnis

Hinweis an den Kindergeldberechtigten: Bitte füllen Sie Punkt e) erst aus, nachdem der Ausbildungsbetrieb die Angaben zum Ausbildungsverhältnis bestätigt hat!

Wir versichern, dass unsere Angaben vollständig sind und der Wahrheit entsprechen. Uns ist bekannt, dass wir alle Daten, die für den Anspruch auf Kindergeld von Bedeutung sind, unverzüglich der Familienkasse mitteilen haben. Den Inhalt des Merkblattes Kindergeld (zu finden unter www.bzg.de oder www.familienkasse.de) haben wir zur Kenntnis genommen.

Hinweise zum Datenschutz: Ihre Daten werden gemäß der §§ 31, 92 bis 70 Einkommensteuergesetz und der Regelungen der Abgabenordnung bzw. aufgrund des Bundeskindergeldgesetzes und des Sozialgesetzbuches verarbeitet. Zweck der Verarbeitung der Daten ist die Prüfung Ihres Anspruchs auf Kindergeld. Nähere Informationen über die Verarbeitung Ihrer Daten durch die Familienkasse und zu Ihren Rechten nach Artikel 13 bis 22 der Datenschutz-Grundverordnung erhalten Sie im Internet auf der Seite [www.familienkasse.de](#) (zu finden unter [www.arbeitsagentur.de/datenbehandlung](#)). Weitere Informationen über die Kontaktstellen der Kreisdatenschutzbeauftragten befinden sich auf der Internetseite [www.kreisdatenschutz.de](#). Die Daten der Kindergeldanträge werden in der Regel nach dem Ende der Kindergeldzahlung noch für 6 Jahre aufbewahrt.

Unterschrift der kindergeldberechtigten Person bzw. der gesetzlichen Vertretung

Unterschrift des vorjährigen Kindes

All Eingaben löschen **Drucken** **Speichern**

(a) Erste Seite

(b) Zweite Seite

Abbildung 2.1: KG5b-Formular

2.1.3 Sonstige Dokumente

Die Kategorie **Sonstiges** steht als Auffangklasse für alle restlichen Dokumente bereit. Darin befinden sich zum Beispiel Schulbescheinigungen, Anträge auf Eintragung bei der Handelskammer oder Studienbescheinigungen. Da diese Dokumente keine Relevanz für die Weiterbeantragung des Kindergeldes bei volljährigen Auszubildenden haben, werden keine Informationen aus ihnen benötigt.

2.2 Fachliche Grundlagen

2.2.1 Der Ist-Zustand

Der schematische Ablauf der YOLO/OCR-Pipeline ist in Abbildung 2.3 dargestellt.

Der Prozess der Dokumentenverarbeitung lässt sich in zwei Stufen einteilen.

In der ersten Stufe werden die hochgeladenen Dokumente des Kunden entgegengenommen. Wenn ein Dokument als PDF vorliegt, wird mithilfe des Textlayers und eines OCR-Modells

Berufsausbildungsvertrag
§§ 10, 11 des Berufsbildungsgesetzes – BBG

Zwischen dem Ausbildenden (Ausbildungsbetrieb) und dem Auszubildenden wird nachstehender Berufsausbildungsvertrag zur Ausbildung im Ausbildungsbetrieb

Apotheker
(wenn einschlägig, bitte einschließlich Fachrichtung, Schwerpunkt, Wahlqualifikationen) und/oder Einsatzgeber nach der Ausbildungsvorordnung bestimmt

nach Maßgabe der Ausbildungsordnung¹ geschlossen.

Beispielschule
Zuständige Berufskammer
Änderungen an den vorliegenden Vertragshälften und von Aushändigen unterschlagbar. Erfüllung in den Verträgen der Betriebsvereinbarungen gilt für beide Vertragshälften. Die vertraglichen Regelungen des Ausbildungsbetriebes (Ausbildungsvertrag) sowie die Regelungen weiterer Steuerabkommen sind bestehend dieses Vertrages

Angaben zum Ausbildenden

Apotheker am Markt
Name des Ausbildenden (Ausbildungsbetrieb):
Marktstraße 1
Straße, Haus-Nr.:
PLZ Ort
E-Mail-Adresse (Angabe freiwillig)
0800 123482394
Telefonnummer
Beispiel, Petta
Name, Vorname verantwortlicher Ausbilderin
Angaben zum/zu gesetzlichen Vertreter(n)
keiner Eltern Mutter Vater Vormund
Name, Vorname
Straße, Haus-Nr.:
PLZ Ort
Name, Vorname
Straße, Haus-Nr.:
PLZ Ort

Angaben zum Auszubildenden

Mustermann Max
Name Vorname
Musterrstraße 4
Straße, Haus-Nr.:
PLZ Ort
01.03.2002 Geburtsdatum
E-Mail-Adresse (Angabe freiwillig)
0800 23834570
Mobile/Telefonnummer (Angabe freiwillig)

§ 1 – Dauer der Ausbildung

1. Dauer
Die Ausbildungsdauer beträgt nach der Ausbildungsvorordnung
36 Monate.
Auf die Ausbildungsdauer wird die Berufsausbildung zurück²
bzw. eine berufliche Fortbildung in
mit Monaten angeholt.³

Die Berufsausbildung wird in
keiner Eltern Mutter Vater Vormund
Vollzeit Teiltzeit (%) der Ausbildungsdauer in Vollzeit durchgeführt.
Die Ausbildungsdauer verlängert sich aufgrund der Teilzeit um Monate.
Die Ausbildungsdauer verkürzt sich vorbehaltlich der Entscheidung der zuständigen Stelle aufgrund
um Monate.
Soviel keine gesetzlich erlaubte Formulierung gewählt wird, steht dies allein der Vereinbarung der Leistungsfähigkeit. Auch darf werden alle Menschen angesprochen – unabhängig von ihrem Geschlecht (siehe).

§ 2 – siehe S. 3 des Berufsausbildungsvertrages

§ 3 – Ausbildungsstätte
Die Ausbildung findet vorbehaltlich der Regelungen nach § 4 Nr. 12 dieses Vertrages in

Name/Ort der Ausbildungsstätte
und den mit dem Betrieb für die Ausbildung (bitteweise zusammenhängende Bau-, Montage- und sonstigen Arbeitsstellen) statt

§ 4 – Pflichten des Ausbildenden
Ausbildungsmethoden aufzuhalten der Ausbildungsstätte) und für den folgenden Zeitraum in den/deren (späteren) Ausbildungsstätte(n) vorgesehen (hierzu zählen auch Auslandsaufenthalte)

§ 5 – Pflichten des/dieses Auszubildenden
Führung von schriftlichen oder elektronischen Ausbildungsnachweisen
folgt geführt.
Der Ausbildungsnachweis wird schriftlich elektronisch

§ 6 – Bestandteile der Vergütung und sonstige Leistungen
Höhe und Fälligkeit
Das Ausbildungsvorhaben fällt in den Geltungsbereich des folgenden Tarifvertrages:
Das Ausbildungsvorhaben fällt nicht in den Geltungsbereich eines gültigen Tarifvertrages.

§ 13 – Vertragssignatur
Der Auszubildende verzögert die Berufsausbildung und deren gewährte Vermehrung und Vertreterin die Vertragssignatur verzögern noch dem Erreichung auszuführen. Bei elektronischer Ausführung ist die Vertragssignatur so zu überprüfen, dass die Signatur den tatsächlichen Willen des Auszubildenden ausdrückt. Diese Auszubildende verpflichtet sich, den Empfang der elektronischen Vertragssignatur selbst zu überprüfen. Der Auszubildende verzögert die Vertragssignatur und der Empfangswiederholung und der Entgegennahme wird von den beiden Auszubildenden nach Abschluss der Berufsausbildung bestätigt. Der Auszubildende verzögert die Vertragssignatur und der Empfangswiederholung und der Entgegennahme bestätigt, die Jahre automatisch.

Vorliegende Vertrag ist
gem. § 13 Satz 1 zweifel (bei Mündlich –fach) ausgestellt
und von Vertragsschließenden ehrlichst unterschrieben worden.
gemäß § 13 Satz 2 elektronisch abgeschlossen und übermittelt worden.

Anlage gemäß § 4 Nr. 1 des Berufsausbildungsvertrages⁴
date_document

Musterrat, 20.09.2022
Ort, Datum
Max Mustermann
Unterschrift des Auszubildenden
Mustermann
Unterschriften des gesetzlichen Vertreters
stamp_company signature_company signature_legal_guardian

(a) Erste Seite

(b) Zweite Seite

Abbildung 2.2: Ausbildungsvertrag der Industrie- und Handelskammer

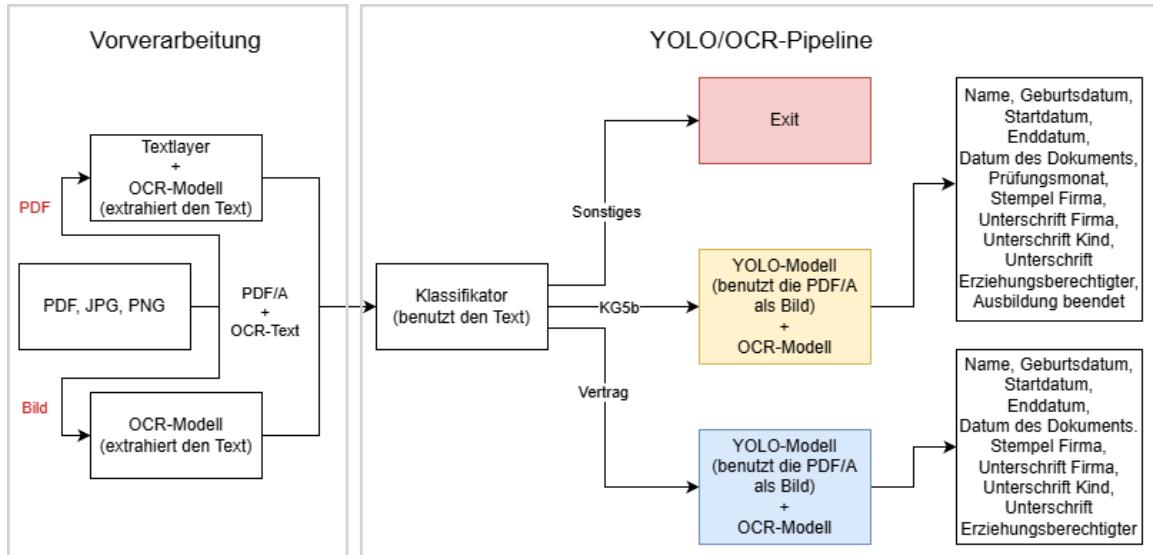


Abbildung 2.3: Schematischer Workflow der aktuellen AuBe-Pipeline

der Text extrahiert, während bei einer Bilddatei lediglich das OCR-Modell zum Einsatz kommt. Als Ergebnis der ersten Stufe werden dann ein PDF/A und der extrahierte Text zurückgeliefert.

Auf Basis des extrahierten Textes wird in der nächsten Stufe zu Beginn eine Klassifikation durchgeführt. Hierbei unterscheidet der Klassifikator zwischen den bereits vorgestellten Dokumententypen: KG5b, Vertrag und Sonstiges. Wurde das Dokument als **Sonstiges** klassifiziert, endet an dieser Stelle die Bearbeitung. Handelt es sich hingegen um ein KG5b oder einen Vertrag, wird je nach Dokumententyp eine Erkennung mit einem YOLO-Modell gestartet. Innerhalb der Bounding-Boxen wird der Text extrahiert und dem jeweiligen Label zugeordnet.

Schlussendlich stehen die erkannten Informationen zur weiteren Verarbeitung bereit.

2.2.2 Rahmenbedingungen und Infrastruktur

Die Entwicklung und Evaluation der Modelle erfolgt unter datenschutzrechtlichen Auflagen. Da im Rahmen dieser Arbeit personenbezogene Echtdaten verarbeitet werden, wird eine isolierte On-Premises-Infrastruktur verwendet.

Die technische Architektur ist in Abbildung 2.4 schematisch dargestellt.

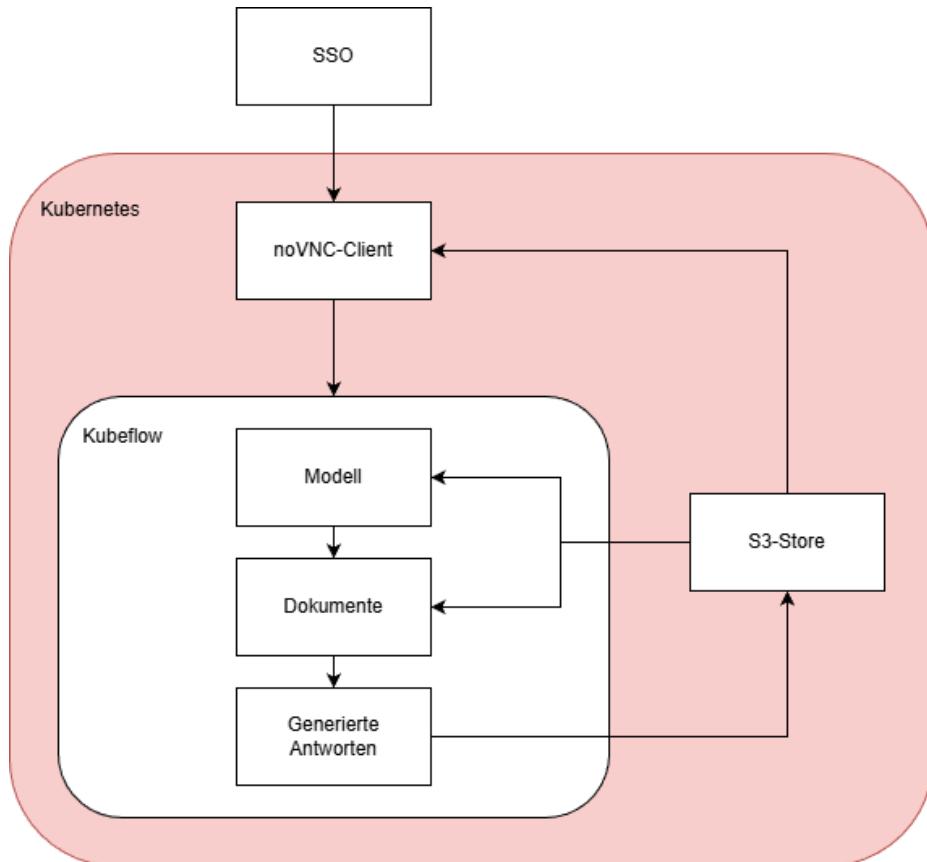


Abbildung 2.4: Schematische Darstellung der Trainings- und Inferenzinfrastruktur

Das System basiert auf einem abgeschotteten Kubernetes-Cluster. Für die rechenintensiven Aufgaben, insbesondere das Fine-Tuning und die Inferenz der VLMs, stehen innerhalb des Clusters zwei NVIDIA A40 GPUs mit jeweils 48 GB VRAM[NVID 22] zur Verfügung. Um die Bereitstellung der VLMs zu erleichtern, wird auf dem Cluster Kubeflow verwendet. Kubeflow ist eine Open-Source-Plattform, die speziell für das Entwickeln, Trainieren und Deployen von Machine-Learning-Modellen konzipiert wurde.

Der Zugang zum Cluster sowie zur Integrated Development Environment (IDE) erfolgt über eine noVNC-Schnittstelle (browserbasierter Remote-Desktop).

Um einen Test- oder Trainingslauf durchzuführen, wird der folgende Prozess durchlaufen:

1. **Initialisierung:** Ein Python-Skript in der IDE startet den Kubeflow-Job.
2. **Datenbereitstellung:** Die Pipeline lädt sowohl das VLM als auch die zu klassifizierenden Dokumentenbilder aus dem S3-Store in den GPU-Cluster.
3. **Verarbeitung:** Die Inferenz oder das Training findet im Kubernetes-Cluster statt.
4. **Persistierung:** Die Ergebnisse werden zurück in den S3-Store geladen.

2.3 Vision Language Models (VLMs)

Ein VLM besteht aus drei Komponenten: einem Image-Encoder, einem Adapter und einem Large Language Model (LLM). In Abbildung 2.5 wird der Aufbau dargestellt.



Abbildung 2.5: Architektur eines Vision Language Models

Der Image-Encoder verarbeitet die Bildeingabe und extrahiert visuelle Merkmale (Features). Zu diesem Zweck kommen häufig vortrainierte Modelle wie Vision Transformer (ViT) oder CLIP zum Einsatz. Diese Modelle zerlegen ein Bild in kleinere Patches, die ähnlich wie Token in Sprachmodellen behandelt werden. Jeder Patch wird in einen Vektor überführt, der die Eigenschaften dieses Bildausschnitts repräsentiert. Das Ergebnis ist eine Sequenz von Bildvektoren, die die visuellen Informationen des gesamten Bildes beinhaltet.

Der Adapter verbindet den Image-Encoder mit dem Sprachmodell. Diese Schicht transformiert die Ausgabe des Image-Encoders in ein Format, das mit den Textvektoren des Sprachmodells kompatibel ist. In vielen Modellen besteht der Adapter aus einer linearen Schicht oder einem kleinen neuronalen Netz. Die Bildvektoren des Image-Encoders werden in denselben Vektorraum wie die Textvektoren des LLM projiziert.

Die Hauptkomponente bildet das LLM, welches für die eigentliche Verarbeitung und Generierung zuständig ist. Das LLM erhält die Textvektoren zusammen mit den Bildvektoren und fusioniert diese. Mithilfe des Attention-Mechanismus werden Beziehungen zwischen visuellen und textuellen Elementen verstanden. So ist es möglich, Fragen zu Bildinhalten zu beantworten oder Bildbeschreibungen zu generieren.

Die größte Herausforderung dieser modernen Technologie ist der hohe VRAM-Verbrauch. Neben der statischen VRAM-Nutzung kommt mit jeder Anfrage ein dynamischer Verbrauch dazu. Dabei setzt sich der statische Verbrauch aus den Gewichten der Modelle zusammen, während sich der dynamische Verbrauch aus den Key-Value-Caches und den Aktivierungen bildet.

2.4 Vorstellung der Modelle

Für die Klassifikation und die Extraktion der Informationen aus den Dokumenten wurden drei verschiedene VLMs evaluiert.

2.4.1 Pixtral-12B

Das Pixtral-12B-2409 [[Pixt 24](#)] wurde im Jahr 2024 von Mistral AI veröffentlicht. Es basiert auf einem 12 Milliarden Parameter großen Text-Decoder mit einem zusätzlichen 400 Millionen Parameter umfassenden Vision-Encoder. Es wurde speziell auf das Verständnis von Bildern und Dokumenten trainiert, weshalb es einen optimalen Kandidaten für die vorliegende Arbeit darstellt. Mit einem theoretischen Kontextfenster von 128.000 Token ermöglicht das Modell die gleichzeitige Verarbeitung mehrerer Bilder [[Agra 24](#)].

2.4.2 Qwen-2.5-VL

Das Qwen2.5-VL-7B-Instruct[Qwen 25b] sowie das Qwen2.5-VL-32B-Instruct[Qwen 25a] wurden im Jahr 2025 von Alibaba Cloud veröffentlicht. Die beiden Modelle basieren auf dem gleichen Image-Encoder mit 600 Millionen Parametern. Lediglich die Größe des Text-Decoders ist mit 7 Milliarden beziehungsweise 32 Milliarden Parametern unterschiedlich. Eine Besonderheit dieser Modelle ist, dass sie speziell auf das Verarbeiten von Dokumenten trainiert wurden[Bai 24].

2.4.3 Benchmarks

Im Folgenden sind die Benchmarks der einzelnen Modelle gelistet. Besonders das Ergebnis des DocVQA-Benchmarks ist von Interesse, da hier ein Visual Question Answering (VQA) auf Dokumentenbildern durchgeführt wird[Mine 21].

Modell	Pixtral-12B	Qwen2.5-VL-7B	Qwen2.5-VL-32B
DocVQA	90,7	95,7	94,8
MMMU	52,0	58,6	70,0

Tabelle 2.1: Benchmark-Ergebnisse der evaluierten Modelle[Bai 24, Qwen 25a, Agra 24]

2.5 Parameter-Efficient Fine-Tuning

Das Anpassen der Gewichte eines bereits trainierten Modells an eine spezifische Domäne setzt eine Infrastruktur mit hoher Rechenleistung voraus. Mithilfe von Parameter-Efficient Fine-Tuning (PEFT) kann die Anzahl der zu trainierenden oder neuen Parameter signifikant gesenkt werden. Infolgedessen verringert sich auch der Rechenaufwand für das Fine-Tuning der Modelle erheblich[Mang 22].

Eine der bekanntesten Methoden im Bereich PEFT ist die Low-Rank Adaptation (LoRA). Anstatt das gesamte Modell neu zu trainieren, friert LoRA die ursprünglichen Gewichte ein und fügt stattdessen trainierbare Matrizen mit einem niedrigen Rang r in jede Schicht der Transformer-Architektur ein. Um Rechenaufwand und Speicherplatz zu sparen, werden ausschließlich diese kleineren Matrizen trainiert[Hu 21].

Ergänzend zu LoRA wird die Variante Rank-Stabilized LoRA (rsLoRA) betrachtet. Bei der Standard-LoRA führt ein steigender Rang r oft nicht zu einer besseren Performance, da der verwendete Faktor (α/r) das Lernen bei höheren Rängen verlangsamen oder hemmen kann. rsLoRA stabilisiert diesen Prozess, indem die Adapter durch die Quadratwurzel des Rangs (α/\sqrt{r}) geteilt werden[Kala 23].

2.6 Verwandte Arbeiten

Ein Feld gilt hierbei als True Positive (TP), wenn sowohl der Key des Feldes vorhanden ist als auch der Value des Feldes mit dem der Ground Truth übereinstimmt[Huan 21].

Kapitel 3

Methodik und Implementierung

Im Rahmen dieses Proof of Concepts werden verschiedene VLMs vergleichend gegenüber der bestehenden YOLO/OCR-Pipeline evaluiert. Für die Entscheidung, ob eines der VLMs die YOLO/OCR-Pipeline ersetzt, sollen die folgenden Forschungsfragen beantwortet werden:

1. Inwiefern ist ein nicht domänen spezifisch angepasstes VLM hinsichtlich der Klassifikations- und Extraktionsgenauigkeit der trainierten YOLO/OCR-Pipeline überlegen, und welche Vorteile bietet ein einzelnes generalistisches Modell gegenüber der Verwendung spezialisierter Einzelsysteme?
2. Wie verhält sich ein kleineres, domänen spezifisch trainiertes Modell gegenüber einem leistungs stärkeren Basismodell in Bezug auf Performanz und Effizienz?
3. In welchem Verhältnis steht der Ressourcenverbrauch der VLMs zu dem der YOLO/OCR-Pipeline?

In Abbildung 3.1 ist ein Überblick der gesamten Methodik abgebildet.

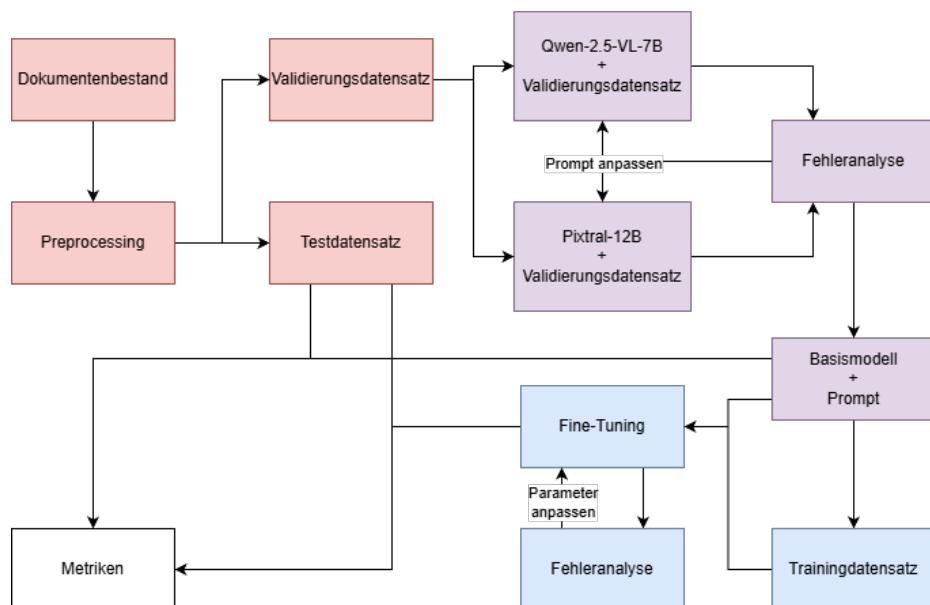


Abbildung 3.1: Überblick der Methodik.

3.1 Modellierung des Dokumentenbestands als Grundlage der Evaluation

3.1.1 Preprocessing und Darstellung der Dokumentarten als JSON-Objekte

Durch Datenlieferungen der Familienkasse stehen mehrere Tausend Dokumente als PDF oder Bilddatei zur Verfügung. Um eine einheitliche Basis zu schaffen und die Klassifikations- und Extraktionsgüte zu verbessern, müssen die Daten vorverarbeitet werden. Im Fokus stehen dabei die Korrektur der Bildausrichtung sowie die Konvertierung der PDF-Dokumente in Bilder. Zur Korrektur der Rotation werden die Exchangeable Image File Format (EXIF)-Metadaten ausgelesen, um die ursprüngliche Ausrichtung wiederherzustellen. Während ältere VLMs Bilder oft auf fixe Größe skalieren, unterstützen die in dieser Arbeit evaluierten Modelle eine dynamische Skalierung. Um jedoch die maximale Token-Anzahl, werden die Bilder auf eine maximale Pixelanzahl skaliert, wobei das ursprüngliche Seitenverhältnis beibehalten wird.

Für die standardisierte Weiterverarbeitung der Modellausgaben ist eine feste Struktur essenziell. Da moderne VLMs darauf trainiert sind, Antworten im JavaScript Object Notation (JSON)-Format zu liefern, werden die extrahierten Informationen in ein vordefiniertes JSON-Schema überführt. Ein wesentlicher Vorteil gegenüber der herkömmlichen Pipeline ist die gleichzeitige Durchführung der Klassifikation sowie der IE in einem einzigen Inferenzschritt. Das Modell klassifiziert die Dokumentenart, die im Feld `type` abgebildet wird. Während das Feld `type` in allen JSON-Schemata konsistent vorhanden ist, variieren die Felder der IE je nach Dokumentenart.

Bei den binären, booleschen Feldern markiert ein `true` das Vorhandensein eines Merkmals, während ein `false` dessen Fehlen oder das Nicht-Erkennen repräsentiert. Um eine strukturelle Konsistenz zu erzwingen, werden Felder, bei denen die dazugehörige Information im Dokument fehlt, bei booleschen Typen mit `false` und bei Textfeldern mit einer leeren Zeichenkette belegt.

Da die Dokumente in den meisten Fällen aus mehreren Seiten bestehen, repräsentiert ein JSON-Objekt das mehrseitige Dokument. Sollten Dokumente verschiedener Arten vermischt vorliegen, generiert das Modell für jede erkannte Dokumentenart ein separates JSON-Objekt.

Die JSON-Schemata für die relevanten Dokumentenarten sind in den Abbildungen 3.2 und 3.3 dargestellt. Bei dem JSON-Schema der sonstigen Dokumente 3.4, findet nur die Klassifikation statt, da hier keine Informationen relevant sind.

```

"file_name": {
    "type": "kg5b",
    "name_child": "name, vorname",
    "birthday_child": "DD.MM.YYYY",
    "start_date_apprenticeship": "DD.MM.YYYY",
    "end_date_apprenticeship": "DD.MM.YYYY",
    "date_document": "DD.MM.YYYY",
    "stamp_company": true,
    "signature_company": true,
    "signature_child": true,
    "signature_legal_guardian": true,
    "apprenticeship_finished": true,
    "exam_month": "MM"
}

```

Abbildung 3.2: JSON-Schema des Dokumententyps KG5b

```

"file_name": {
    "type": "vertrag",
    "name_child": "name, vorname",
    "birthday_child": "DD.MM.YYYY",
    "start_date_apprenticeship": "DD.MM.YYYY",
    "end_date_apprenticeship": "DD.MM.YYYY",
    "date_document": "DD.MM.YYYY",
    "stamp_company": true,
    "signature_company": true,
    "signature_child": true,
    "signature_legal_guardian": true
}

```

Abbildung 3.3: JSON-Schema des Dokumententyps Ausbildungsvertrag

```

"file_name": {
    "type": "sonstiges"
}

```

Abbildung 3.4: JSON-Schema des Dokumententyps Sonstiges

3.1.2 Test- und Validierungsdatensatz

Nach der Vorbereitung der Dokumente wird ein Test- und Validierungsdatensatz erstellt. Der Testdatensatz als auch der Validierungsdatensatz umfasst insgesamt 60 Dokumente, aufgeteilt in jeweils 30 Beispiele Dokumentenarten. Um die reale Datenverteilung bestmöglich abzubilden, werden die Dokumente manuell ausgewählt.

Die Datensätze deckt dabei verschiedene Herausforderungen ab. Neben Dokumenten mit geringer Qualität, vorrangig schlechte Scans und Fotos, befinden sich fehlerhaft ausgefüllte Dokumente sowie Dokumente mit fehlenden Unterschriften und Stempeln in den Datensätzen. Um die hohe Varianz der Verträge und sonstigen Dokumente abzubilden, werden in den Datensätzen viele unterschiedliche Layouts gewählt. Die Exemplare für die Verträge beinhalten sowohl starre Layouts als auch Fließtexte unterschiedlicher Firmen und Kammern. Die sonstigen Dokumente beinhalten keine kontextfremden Dokumente, sondern Schulbescheinigungen, Studienbescheinigungen und Eintragungen bei der Handwerkskammer, um eine zusätzliche Herausforderung zu stellen.

Zur Erstellung der Wahrheitswerte (Ground Truth) werden alle Dokumente manuell annotiert und in das entsprechende JSON-Format überführt.

3.2 Evaluation der generierten JSON-Objekte

3.2.1 Validierung des VLM-Ouputs

Um den JSON-Output der VLMs zu limitieren, existieren verschiedene Ansätze. Während komplexere Lösungen nur Tokens zulassen die in JSON-Objekten enthalten sein könnten, wird in dieser Evaluation ein einfacherer Ansatz gewählt. Hierbei werden aus der generierten Antwort JSONs gefiltert und geparsst. Sobald das Parsen der JSON fehlschlägt, wird diese zurück an das VLM geliefert, mit der Anweisung diese zu korrigieren. Das Modell bekommt insgesamt drei Versuche eine valide JSON zu generieren, bevor das Dokument als ungültig eingestuft wird.

3.2.2 Vergleichslogik der unterschiedlichen JSON-Felder

Mit dem Ziel, ein realistisches Ergebnis zu erhalten, müssen unterschiedliche Feldtypen spezifisch verglichen werden. Die Menge an distinkten Typen beinhaltet: Namen, Booleans, Zeichenketten, Daten und Monate.

Namen werden mit der Levenshtein-Similarity verglichen, um kleinere Fehler oder verschiedene Varianten zu tolerieren. Beispielsweise wird ein Name, der ue statt ü enthält, nicht als Fehler erkannt. Der Schwellenwert, der zwischen korrekt und nicht korrekt entscheidet, liegt bei 0,8. Des Weiteren werden minimale OCR-Fehler gefiltert. Die YOLO/OCR-Pipeline verwendet ebenso diesen Vergleich mit den gleichen Schwellenwerten, somit wird das Ergebnis vergleichbar.

Zeichenketten und Booleans werden exakt verglichen. Hierzu zählen die Felder `type`, `stamp_company`, `signature_company`, `signature_child`, `signature_legal_guardian` und `apprenticeship_finished`.

Lediglich die booleschen Werte werden zu `true` normalisiert, da hier eine potenzielle Fehlerquelle liegt.

Daten werden einheitlich in das Format DD.MM.YYYY gebracht, obwohl dies nicht dem internationalen Standard entspricht. Das ist darauf zurückzuführen, dass so die Erkennung des VLMs am robustesten ist, da die deutschen Dokumente auch mit diesem Format ausgefüllt werden.

Monate werden in das Format MM normalisiert. Hierbei werden ausgeschriebene Monatsnamen sowie vollständige Datumsangaben angepasst.

3.2.3 Bewertung der Klassifikation und der Information Extraction

Um die generierten JSON-Objekte des VLMs schlussendlich bewerten zu können, werden die Klassifikation und IE unabhängig voneinander bewertet.

Die Bewertung der Klassifikation ist ein Multiclass-Problem, da hier mehr als zwei Klassen vorliegen (KG5b, Ausbildungsvertrag, Sonstiges). Für die Bestimmung der Güte der Modelle, wird eine Confusion-Matrix erstellt, wobei der Fokus auf dem F1-Score liegt.

Die Bewertung der Information Extraction ist komplexer als die der Klassifikation. Um den Entity-Level F1-Score, der die Hauptmetrik für den Vergleich darstellt, zu berechnen werden einige Metadaten für jede generierte JSON bestimmt. Dazu gehören die Status der Felder, also ob diese vorhanden, nicht vorhanden oder halluziniert sind. Sobald ein Feld als vorhanden markiert wird, wird mit den verschiedenen Vergleichslogiken die Richtigkeit der Felder bestimmt und den Metadaten hinzugefügt.

Auf Basis dieser Metadaten werden für den gesamten Dokumentenkorpus die True Positive (TP), False Positive (FP) und False Negative (FN) gezählt. Ein Feld zählt als TP, wenn es vorhanden sowie korrekt ist. Zu den FN gehören Felder, die entweder fehlen oder vorhanden, aber falsch sind. Ein Feld, das vorhanden aber falsch ist, zählt zusätzlich zu den FP zusammen mit den Feldern die halluziniert wurden. Diese doppelte Bestrafung stellt eine sehr strenge Bewertung für inhaltliche Fehler dar. True Negatives (TN) sind nichtzählbar, da die Menge an nicht gefundenen, leeren Feldern, theoretisch unendlich groß ist.

3.2.4 Messung von Latenz, Energieverbrauch und VRAM-Nutzung

Im Hinblick auf die Bewertung der Modelle werden neben der Güte der Klassifikation und IE die Latenz, die VRAM-Nutzung sowie der Energieverbrauch gemessen.

Die Latenz der Modelle stellt lediglich das Delta zwischen Start- und Endzeit der Inferenz dar. Hierbei wird der Zeitpunkt gemessen, an dem die eigentliche Inferenz startet und wenn

die Antwort bereitsteht. Das Laden des Modells wird nicht betrachtet, da diese keinen Einfluss auf die Antwortzeit in der produktiven Umgebung hat und folglich nicht relevant für die Evaluation ist.

Des Weiteren wird der Energieverbrauch während der Inferenz in Joule erfasst. Die Messung der Leistung in Watt ist nicht zielführend, da sie die Zeit nicht berücksichtigt und Modelle mit geringerer Latenz bei vergleichbarem Energieverbrauch benachteiligt werden.

Wie auch der Energieverbrauch wird die VRAM-Nutzung während der Inferenz gemessen. Der Messzeitraum für den Energieverbrauch als auch für die VRAM-Nutzung ist gleich der Latenz der Modelle. Mit der Initialisierung des Modells wird einmalig der statische Verbrauch gemessen.

3.3 Auswahl des Basismodells

3.3.1 Prompt-Design

Die Gestaltung der Prompts für die Modellauswahl folgt der R-K-F-Formel. Zur Strukturierung werden die Abschnitte durch Markdown-Überschriften getrennt, um dem Modell die Zuordnung zu erleichtern. Während der Kontext die vorliegenden Dokumentarten detailliert beschreibt, definiert das Format die JSON-Schemata. Zum Einsparen von Tokens wird das Modell strikt angewiesen lediglich das JSON-Objekt zu generieren.

Um den Ressourcenverbrauch zu minimieren wird ein Zero-Shot Ansatz gewählt. Während sowohl bei One-Shot als auch Few-Shot Prompts das Kontextfenster durch mitgegebene Bilder und Antworten belegt wird, ist bei Zero-Shot Prompts die Leistung des Modells rein auf die Instruktion zurückzuführen. Dies minimiert die Kosten pro Inferenz als auch die Latenz.

VLMs werden primär auf englischen Texten trainiert, jedoch werden auch deutschsprachige Prompts untersucht, um zu evaluieren, ob die Sprache der Prompts einen Einfluss auf die Klassifikation und IE bei deutschen Dokumenten hat.

3.3.2 Modellauswahl

Für das Fine-Tuning muss ein geeignetes Basismodell gefunden werden. Wie in den Lila markierten Felder in Abbildung 3.1 dargestellt, erfolgt die Auswahl in einem iterativen Prozess.

In jeder Iteration wird das Pixtral-12B sowie das Qwen-2.5-VL-7B mit dem Validierungsdatensatz evaluiert. Daraufhin werden die Fehler der Klassifikation und IE analysiert und

auf dessen Grundlage der Prompt angepasst. Nach der Anpassung wird erneut mit dem Validierungsdatensatz jedes Modell erprobt.

Dieser Prozess wird so lange fortgesetzt, bis durch die Anpassung des Prompts keine Steigerung der Metriken des Validierungsdatensatzes erzielt wird. Schlussendlich wird das finale Modell mit dem verfeinerten Prompt mit dem Testdatensatz getestet. Somit wird in ein Data Leakage ausgeschlossen und die Metriken werden Vergleichbar.

3.4 Fine-Tuning des Basismodells

3.4.1 Trainingsdatensatz

Um ein qualitativ hochwertiges Fine-Tuning zu ermöglichen, ist ein deutlich umfangreicherer Datensatz als für die Modellauswahl erforderlich. Der finale Trainingsdatensatz umfasst 610 Dokumente, die sich aus 227 Verträgen, 165 KG5b-Formularen und 218 sonstigen Dokumenten zusammensetzen.

Die Klassen sind bewusst ungleich verteilt. KG5b-Formulare haben ein starres Layout, wodurch schon mit einer geringeren Anzahl an Trainingsdaten ein gutes Ergebnis erwartet wird. Infolge der hohen Varianz der Verträge und der sonstigen Dokumente erhalten diese im Training eine höhere Gewichtung. Anders als beim Testdatensatz werden hier die Exemplare nicht manuell, sondern mithilfe einer Stichprobe ausgewählt. Dabei werden die Daten aus einem einwöchigen Zeitraum betrachtet, wodurch ohne manuelle Auswahl eine genaue Repräsentation der realen Daten entsteht. Des Weiteren wird so der Selektionsbias verhindert.

Zur effizienten Erstellung der Ground Truth wird ein Model-Assisted Labeling eingesetzt. Hierbei generiert das im Prozess der Modellauswahl [3.3.2](#) gewonnene Modell, zusammen mit dem angepassten Prompt, für jedes Dokument das jeweilige JSON-Objekt. Im Anschluss werden diese JSON-Objekte manuell kontrolliert und korrigiert. Durch den Einsatz des Modells zur Ermittlung der Ground Truth konnte der Aufwand erheblich gesenkt werden.

3.4.2 Optimierung des Fine-Tuning

Das Basismodell aus [3.3.2](#) wurde mit mehreren Methoden des PEFT und dem Framework Unslloth trainiert. Im Laufe des Trainings wurde verschiedene Strategien angewandt, die sowohl die Veränderung der Parameter als auch des Trainingsdatensatzes [3.4.1](#) beinhalten.

Ein wesentlicher Aspekt der Optimierung lag in der Augmentation des Datensatzes. Vor jedem erneuten Trainingslauf wurde der Datensatz vollständig randomisiert, um eine zufällige Reihenfolge der Trainingsdaten zu erhalten. Des Weiteren wurden in einigen der

Trainingsläufe die Seiten der einzelnen Dokumente getauscht, um dem Modell zu lernen die Informationen, ohne Position zu extrahieren.

Die Optimierung der Parameter erfolgte iterativ wie im blau markierten Bereich in Abbildung 3.1 zu erkennen. Darunter zählten die Parameter der Ranggröße, des Optimizers

Kapitel 4

Ergebnisse

Alle der aufgelisteten Ergebnisse wurden mit dem Testdatensatz [3.1.2](#) evaluiert.

4.1 Ergebnisse der YOLO/OCR-Pipeline gegenüber der Basismodelle

Die Evaluation der IE in Abbildung [4.1](#) zeigt deutliche Unterschiede zwischen den modernen VLMs und der YOLO/OCR-Pipeline.

Schwächen der YOLO/OCR-Pipeline zeigten sich bei der Verarbeitung von Ausbildungsverträgen. Hier erreichte die Pipeline einen F1-Score von 0,51. Im Vergleich schnitten insbesondere die beiden Qwen-Modelle deutlich besser ab. Das Qwen-2.5-VL-7B erkannte mit einem F1-Score von 0,83 und das Qwen-2.5-VL-32B mit einem F1-Score von 0,94 einen Großteil der Informationen. Das Pixtral-12B befindet sich mit einem F1-Score von 0,50 am unteren Ende der IE.

Bei den KG5b-Formularen liegt die Pipeline mit 0,75 nur knapp unter den Qwen-Modellen. Während das Qwen-2.5-VL-7B mit 0,8 und das Qwen-2.5-VL-32B mit 0,87 die stärksten Kontrahenten darstellen, erreichte auch hier das Pixtral-12B gerade einmal einen F1-Score von 0,46.

Die Klassifikation in Abbildung [4.2](#) zeigt ein einheitlicheres Ergebnis. Während das Qwen-2.5-VL-32B alle Dokumentarten am besten klassifiziert, befinden sich die restlichen Modelle mit einer maximalen Differenz von 0,03 in allen Dokumentarten auf einem Level. Bei Betrachtung der in der Abbildung [4.3](#) beinhalteten Confusion Matrices fallen jedoch qualitative Unterschiede auf. Das Qwen-2.5-VL-32B klassifiziert die Verträge als auch KG5b-Formulare fehlerfrei. Im Gegensatz dazu erkennt die YOLO/OCR-Pipeline die sonstigen Dokumente perfekt, klassifiziert aber einige der Verträge und KG5b-Formulare als sonstige Dokumente. Das Pixtral-12B weist die größten Unsicherheiten auf und klassifizierte insgesamt 25 Dokumente als Vertrag.

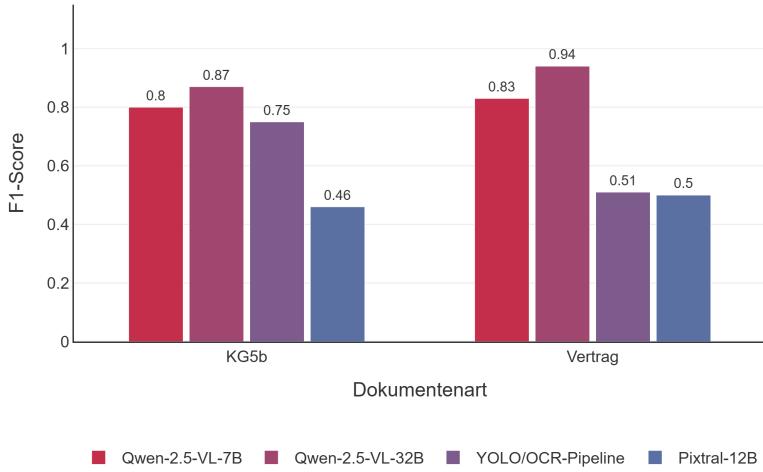


Abbildung 4.1: Ergebnisse der Information Extraction

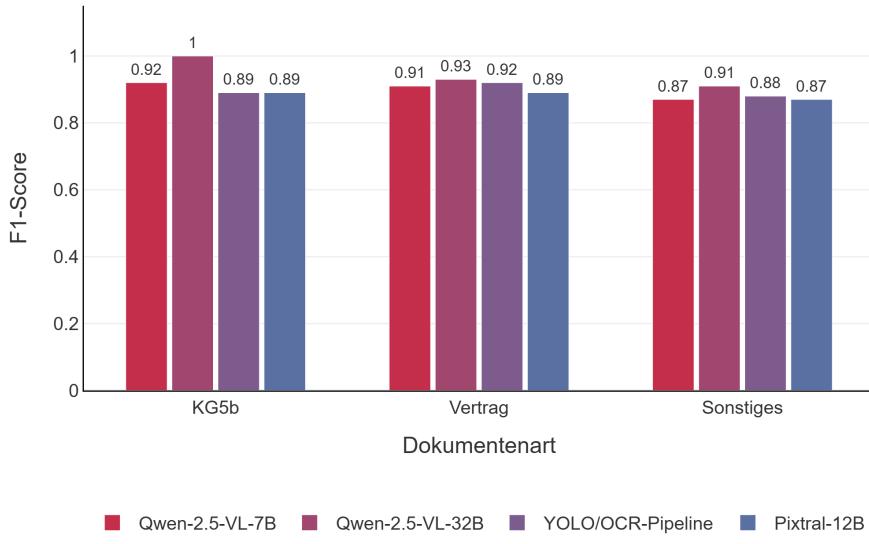


Abbildung 4.2: Ergebnisse der Klassifikation

Hinsichtlich der Latenz kehrt sich das Bild um. Die YOLO/OCR-Pipeline antwortet mit einer durchschnittlichen Latenz von 0,58 Sekunden nahezu sofort. Das kleinste VLM, das Qwen-2.5-VL-7B, benötigt mit durchschnittlich 7,43 Sekunden fast 13-mal länger als die Pipeline. Während das Pixtral mit durchschnittlich 10,28 Sekunden nur knapp über dem Qwen-2.5-VL-7B liegt, braucht das größte Modell im Durchschnitt 64,16 Sekunden

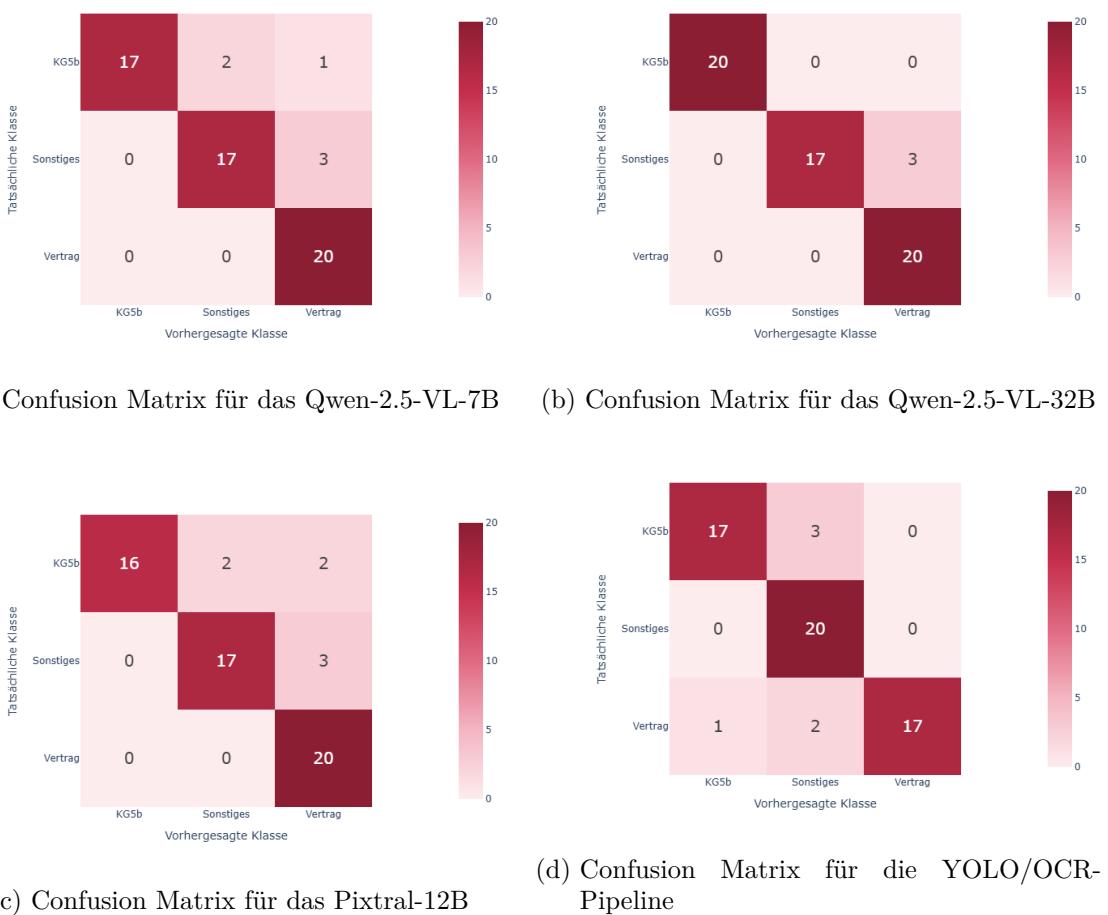


Abbildung 4.3: Confusion Matrices aller Modelle

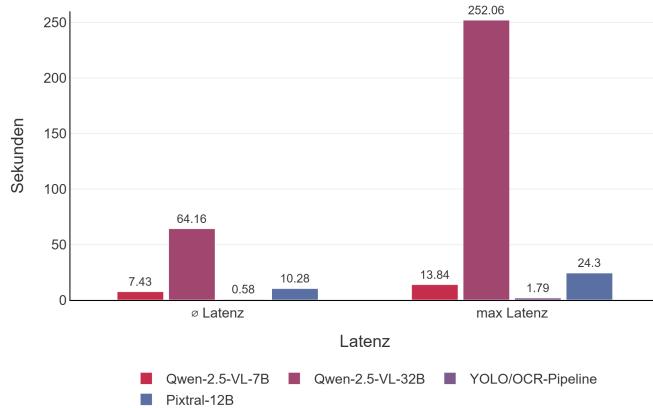


Abbildung 4.4: Ergebnisse der Latenzmessung

4.2 Ergebnisse des weitertrainierten Modells gegenüber dem größeren Basismodell

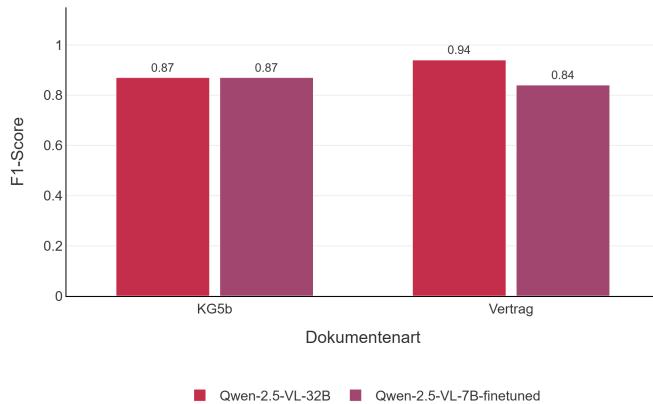


Abbildung 4.5: Ergebnisse der Information Extraction

4.3 Ressourcenverbrauch der VLMs gegenüber der YOLO/OCR-Pipeline

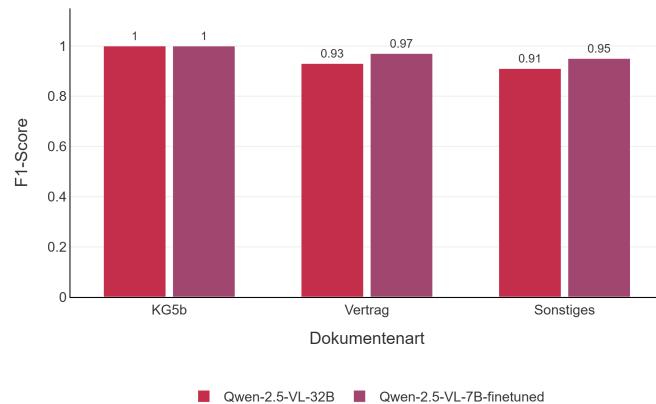


Abbildung 4.6: Ergebnisse der Klassifikation

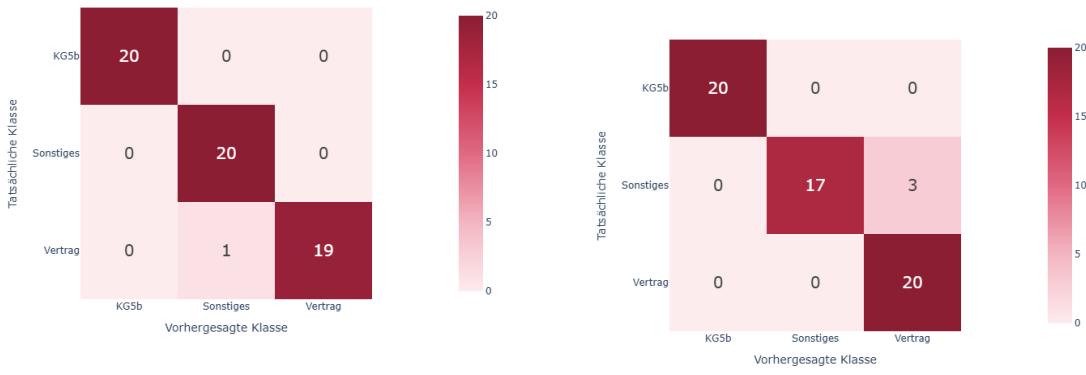


Abbildung 4.7: Confusion Matrices aller Modelle

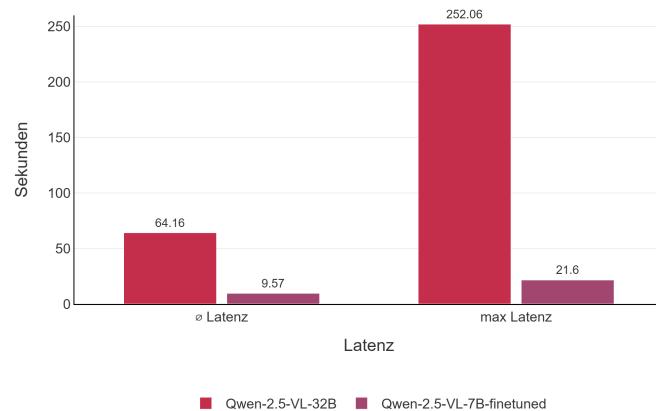


Abbildung 4.8: Ergebnisse der Latenzmessung

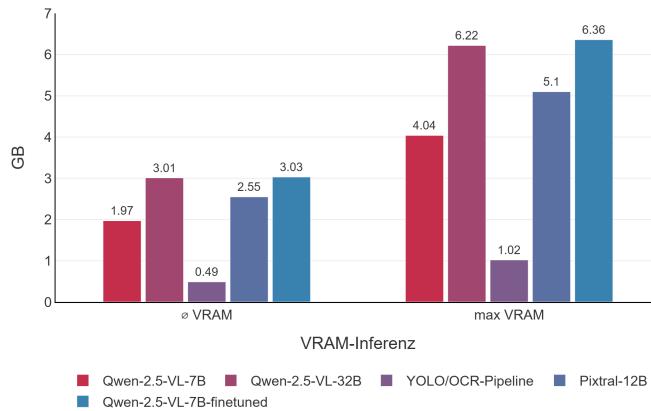


Abbildung 4.9: Ergebnisse der VRAM-Nutzung während der Inferenz

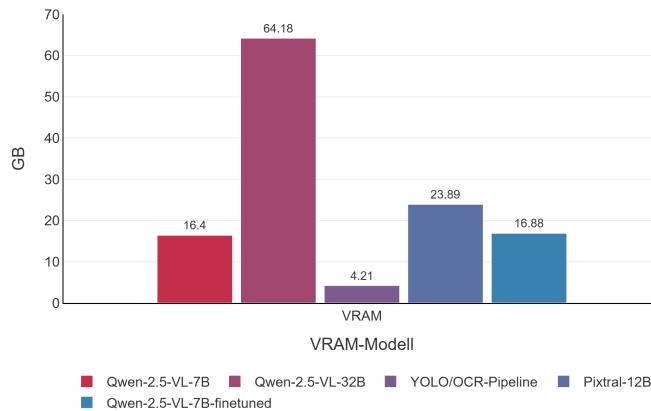


Abbildung 4.10: Ergebnisse der VRAM-Nutzung des geladenen Modells

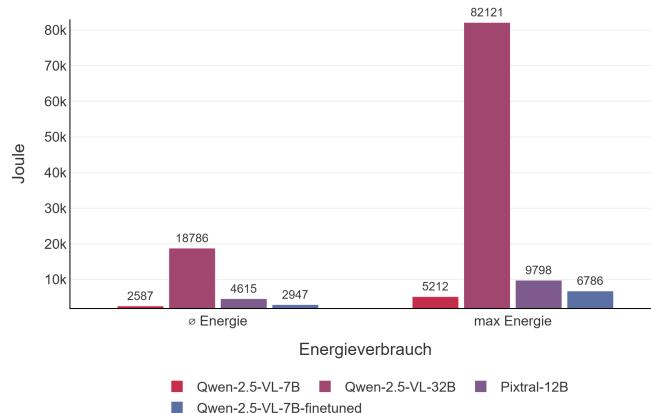


Abbildung 4.11: Ergebnisse des Energieverbrauchs

Kapitel 5

Diskussion

Eine besondere Herausforderung bei der Verarbeitung stellen die binären Informationen dar, wie etwa die Checkbox für das Feld `apprenticeship-ended`. Hier muss das Modell visuell unterscheiden, ob ein Kasten leer, angekreuzt oder durchgestrichen ist. Ebenso kritisch ist die Detektion von `signature_company` und `stamp_company`. Im Gegensatz zu Textfeldern ist hier nicht der textliche Inhalt des Stempels relevant, sondern lediglich dessen Vorhandensein. Die Stempel sind oft blass oder eingefärbt, was besonders bei Schwarz-Weiß Scans zu einer schlechten Qualität führt. Zudem liegen die Felder `date_document` und `signature_company` nah beieinander. Unterschriften sind regelmäßig größer als ihr vorgesehener Platz, wodurch das Feld `date_document` überdeckt wird. Zusätzlich werden die Stempel häufig zusammen mit der Unterschrift auf der linken Seite platziert.

- yolo ocr schneidet am besten ab wegen pipeline design bei sonstigen dokumenten - fehlerhafte klassifikation zieht sich durch - confusion matrix yolo ocr viele als sonstige -> große Fehler

Kapitel 6

Zusammenfassung

Abbildungsverzeichnis

2.1	KG5b-Formular	4
2.2	Ausbildungsvertrag der Industrie- und Handelskammer	5
2.3	Schematischer Workflow der aktuellen AuBe-Pipeline	5
2.4	Schematische Darstellung der Trainings- und Inferenzinfrastruktur	6
2.5	Architektur eines Vision Language Models	7
3.1	Überblick der Methodik.	11
3.2	JSON-Schema des Dokumententyps KG5b	13
3.3	JSON-Schema des Dokumententyps Ausbildungsvertrag	13
3.4	JSON-Schema des Dokumententyps Sonstiges	13
4.1	Ergebnisse der Information Extraction	20
4.2	Ergebnisse der Klassifikation	20
4.3	Confusion Matrices aller Modelle	21
4.4	Ergebnisse der Latenzmessung	22
4.5	Ergebnisse der Information Extraction	22
4.6	Ergebnisse der Klassifikation	23
4.7	Confusion Matrices aller Modelle	23
4.8	Ergebnisse der Latenzmessung	23
4.9	Ergebnisse der VRAM-Nutzung während der Inferenz	24
4.10	Ergebnisse der VRAM-Nutzung des geladenen Modells	24
4.11	Ergebnisse des Energieverbrauchs	24

Tabellenverzeichnis

2.1 Benchmark-Ergebnisse der evaluierten Modelle[Bai 24, Qwen 25a, Agra 24] 9

List of Listings

Literaturverzeichnis

- [Agra 24] P. Agrawal *et al.* “Pixtral 12B”. 2024.
- [Arbe 22] B. für Arbeit. “Erklärung zum Ausbildungsverhältnis (KG 5b)”. https://www.arbeitsagentur.de/datei/dok_ba031910.pdf, 2022.
- [Arbe 24] B. für Arbeit. “Kindergeld / Kinderzuschlag Jahreszahlen 2024”. https://statistik.arbeitsagentur.de/Statistikdaten/Detail/202412/famka/famka-jz/famka-jz-d-0-202412-pdf.pdf?__blob=publicationFile&v=3, 2024.
- [Bai 24] S. Bai *et al.* “Qwen2.5-VL Technical Report”. 2024.
- [Hand 25] D. I. und Handelskammer. “Berufsausbildungsvertrag”. <https://www.dihk.de/resource/blob/140492/9f4fa2617d668aa62f5d3caad77cdbaa/bildung-musterausbildungsvertrag-data.pdf>, 2025.
- [Hu 21] E. J. Hu *et al.* “LoRA: Low-Rank Adaptation of Large Language Models”. 2021.
- [Huan 21] Z. Huang *et al.* “ICDAR2019 Competition on Scanned Receipt OCR and Information Extraction”. 2021.
- [Kala 23] D. Kalajdzievski. “A Rank Stabilization Scaling Factor for Fine-Tuning with LoRA”. 2023.
- [Mang 22] S. Mangrulkar *et al.* “PEFT: State-of-the-art Parameter-Efficient Fine-Tuning methods”. <https://github.com/huggingface/peft>, 2022.
- [Mine 21] M. Minesh, K. Dimosthenis, and J. C.V. “DocVQA: A Dataset for VQA on Document Images”. 2021.
- [NVID 22] NVIDIA. “NVIDIA A40”. <https://www.nvidia.com/de-de/data-center/a40/>, 2022.
- [Pixt 24] Pixtral. “Pixtral-12B-2409”. <https://huggingface.co/mistralai/Pixtral-12B-2409>, 2024.
- [Qwen 25a] Qwen. “Qwen2.5VL-32B-instruct”. <https://huggingface.co/Qwen/Qwen2.5-VL-32B-Instruct>, 2025.

[Qwen 25b] Qwen. “Qwen2.5VL-7B-instruct”. <https://huggingface.co/Qwen/Qwen2.5-VL-7B-Instruct>, 2025.

Glossar

Accuracy Metrik, die den Anteil der insgesamt korrekten Vorhersagen an der Gesamtzahl der getroffenen Vorhersagen beschreibt.. i

AuBe Ausbildungsbescheinigungen. i

BA Bundesagentur für Arbeit. i, 1

EXIF Exchangeable Image File Format. i, 12

F1-Score Das harmonische Mittel aus Precision und Recall.. i

FamKa Familienkasse der Bundesagentur für Arbeit. i

Few-Shot Prompting Ein Ansatz, bei dem dem Modell mehrere Beispiele im Prompt übergeben werden.. i

FN False Negative. i, 15

FP False Positive. i, 15

IDE Integrated Development Environment. i, 7

IE Information Extraction (IE) umfasst nach Jurafsky und Martin die Aufgabe, Ereignisse oder Situationen in Dokumenten zu identifizieren und die entsprechenden Felder eines vorgegebenen Templates zu füllen[?]. Ziel hierbei ist es, vordefinierte Entitäten aus einem Dokumentenbild zu extrahieren und diese in ein maschinenlesbares Schema zu überführen.. i, 12, 15, 16, 19, 37

JSON JavaScript Object Notation. i, 12, 14, 15, 16, 17

Latenz Die Zeitdauer, die das System für die Inferenz einer einzigen Antwort benötigt.. i, 20

Levenshtein-Similarity Die Levenshtein-Similarity ist ein aus der Levenshtein-Distanz abgeleiteter, normalisierter Ähnlichkeitswert zwischen zwei Zeichenketten. Formal basiert sie auf der Levenshtein-Distanz, also der minimalen Anzahl von Einfügungen, Löschungen oder Ersetzungen, die nötig sind, um eine Zeichenkette in eine andere zu überführen[?]. i

LLM Large Language Model. i, 7, 8

LoRA Low-Rank Adaptation. i, 9

MLLM Multimodal Large Language Model. i

OCR Optical Character Recognition. i, 1, 4, 5, 11, 14

One-Shot Prompting Eine Technik, bei der dem Modell genau ein Beispiel für die Lösung im Prompt bereitgestellt wird.. i

PEFT Parameter-Efficient Fine-Tuning. i, 9, 17

Precision Gibt den Anteil der tatsächlich positiven Fälle an allen vom Modell als positiv klassifizierten Fällen an.. i

Prompt Engineering Ist ein Prozess, um die effektivste Anweisung für eine spezifische Aufgabe zu finden[?]. i

R-K-F-Formel Strategie im Prompt Engineering zur strukturierten Anweisung eines LLMs. Dabei steht R für Rolle (Rolle, die das Modell einnehmen soll), K für Kontext (relevante Hintergrundinformationen) und F für Format (Vorgabe der gewünschten Ausgabestruktur).. i

Recall Gibt den Anteil der korrekt als positiv erkannten Fälle an allen tatsächlich vorhandenen positiven Fällen an.. i

rsLoRA Rank-Stabilized Low-Rank Adaptation. i, 9

TN True Negative. i, 15

TP True Positive. i, 10, 15

Unsloth Open-Source-Framework zum beschleunigten und speichereffizienten Fine-Tuning von Large Models, das speziell für Methoden des PEFT optimiert ist.. i

ViT Vision Transformer. i, 8

VLM Vision Language Model. i, 1, 2, 7, 8, 11, 12, 14, 15, 16, 19, 20

VQA Visual Question Answering. i, 9

VRAM Video Random Access Memory. i, 7, 8, 15, 16

VRAM-Auslastung Der maximale Bedarf an Grafikspeicher der GPU, der während der Ausführung eines Modells gemessen wird.. i

YOLO You Only Look Once. i, 1, 4, 6, 11, 14

Zero-Shot Prompting Eine Methode, bei der dem Modell eine Aufgabe gestellt wird, ohne dass zusätzliche Beispiele im Prompt enthalten sind.. i