

# Steigerung der GPT-Leistung für Named Entity Recognition in öffentlichen Ausschreibungen durch gezielte Anpassung und Feinabstimmung von Prompts

## Bachelorarbeit

zur Erlangung des Grades *Bachelor of Science (B.Sc.)*  
im Studiengang Informatik

vorgelegt von  
Andy Schmidt

Erstgutachter: Dr. Kai Sauerwald  
Artificial Intelligence Group, FernUniversität in Hagen

Betreuer: Prof. Dr. Fatih Gedikli  
Institut Informatik, Hochschule Ruhr West



# Erklärung

Ich erkläre, dass ich die Bachelorarbeit selbstständig und ohne unzulässige Inanspruchnahme Dritter verfasst habe. Ich habe dabei nur die angegebenen Quellen und Hilfsmittel verwendet und die aus diesen wörtlich oder sinngemäß entnommenen Stellen als solche kenntlich gemacht. Die Versicherung selbstständiger Arbeit gilt auch für enthaltene Zeichnungen, Skizzen oder graphische Darstellungen. Die Bachelorarbeit wurde bisher in gleicher oder ähnlicher Form weder derselben noch einer anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht. Mit der Abgabe der elektronischen Fassung der endgültigen Version der Bachelorarbeit nehme ich zur Kenntnis, dass diese mit Hilfe eines Plagiatserkennungsdienstes auf enthaltene Plagiate geprüft werden kann und ausschließlich für Prüfungszwecke gespeichert wird.

Der Veröffentlichung dieser Arbeit auf der Webseite des Lehrgebiets Künstliche Intelligenz und damit dem freien Zugang zu dieser Arbeit stimme ich ausdrücklich zu.

Für diese Arbeit erstellte Software wurde quelloffen verfügbar gemacht, ein entsprechender Link zu den Quellen ist in dieser Arbeit enthalten. Gleiches gilt für angefallene Forschungsdaten.

Düsseldorf, 10.05.2025

.....  
(Ort, Datum)



.....  
(Unterschrift)



## Zusammenfassung

Öffentliche Ausschreibungen sind ein essenzielles Instrument zur Sicherstellung von Transparenz, Fairness und Wettbewerb in der öffentlichen Beschaffung. Aufgrund der Heterogenität und Komplexität der Online-Vergabeportale gestaltet sich die manuelle Extraktion relevanter Entitäten, wie Auftraggeber oder Fristen, als herausfordernd. Um diesen Prozess zu automatisieren, kann Named Entity Recognition (NER) aus dem Bereich des Natural Language Processing (NLP) eingesetzt werden. Die Feinabstimmung von Large Language Models (LLMs) wie Bidirectional Encoder Representations from Transformers (BERT) ist jedoch ressourcenintensiv, insbesondere da die Datenmenge in öffentlichen Ausschreibungen vergleichsweise gering ist. Neuere Forschungen zeigen, dass durch gezielte Gestaltung von Prompts für reine Decoder-Modelle wie Generative Pre-trained Transformers (GPT) eine hohe Modellleistung erzielt werden kann, ohne kostspieliges Fine-Tuning durchführen zu müssen.

Diese Arbeit untersucht Strategien zur Optimierung von Prompts für die NER in öffentlichen Ausschreibungen mit GPT. Es werden theoretische Grundlagen zu Sprachmodellen, Prompt Engineering und In-Context-Learning (ICL) dargestellt sowie domänenübergreifende Anwendungen von NER analysiert. In einem experimentellen Setup werden verschiedene Prompt-Strategien getestet und ihre Effizienz evaluiert. Die Ergebnisse liefern Erkenntnisse darüber, wie GPT durch gezielte Anpassung der Prompts eine effiziente und kostengünstige Extraktion relevanter Entitäten in öffentlichen Ausschreibungen ermöglichen kann.

## Abstract

Public tenders are an essential instrument for ensuring transparency, fairness, and competition in public procurement. Due to the heterogeneity and complexity of online procurement platforms, manual extraction of relevant entities such as contracting authorities or deadlines can be challenging. To automate this process, Named Entity Recognition (NER) from the field of Natural Language Processing (NLP) can be utilized. Fine-tuning large language models (LLMs) like Bidirectional Encoder Representations from Transformers (BERT) is resource-intensive, especially given the relatively small dataset in public tenders. Recent research indicates that high model performance can be achieved through targeted prompt design for pure decoder models like Generative Pre-trained Transformers (GPT), without the need for costly fine-tuning.

This study explores strategies for optimizing prompts for NER in public tenders using GPT. Theoretical foundations on language models, prompt engineering, and in-context learning (ICL) are presented, alongside cross-domain applications of NER. Various prompt strategies are tested in an experimental setup, evaluating their efficiency. The results provide insights into how GPT, through targeted prompt adjustments, can enable efficient and cost-effective extraction of relevant entities in public tenders.



# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
<b>2</b>	<b>Theoretische Grundlagen</b>	<b>4</b>
2.1	Sprachmodelle . . . . .	4
2.1.1	Funktionsweise reiner Decoder-Modelle . . . . .	4
2.2	Prompt Engineering . . . . .	6
2.2.1	Prompt-Strategien . . . . .	6
2.2.2	Kontextuelles Lernen . . . . .	8
2.2.3	Prompt Design . . . . .	9
2.3	Wirtschaftliche Relevanz öffentlicher Ausschreibungen . . . . .	11
2.3.1	Elektronische Ausschreibungsportale . . . . .	12
<b>3</b>	<b>Stand der Wissenschaft</b>	<b>16</b>
3.1	Ansätze in der Named Entity Recognition . . . . .	17
3.2	Domänenübergreifender Einsatz von Few-Shot Learning für Named Entity Recognition . . . . .	19
3.3	Reine Decoder Modelle und Few-Shot Prompting für Named Entity Recognition . . . . .	21
3.4	Aufbau einer Bekanntmachung . . . . .	22
3.5	Named Entity Recognition im Bereich öffentlicher Ausschreibungen	24
<b>4</b>	<b>Methode</b>	<b>28</b>
4.1	Vorgehen . . . . .	28
4.2	Technischer Aufbau . . . . .	28
4.3	Aufbereitung der Testdaten . . . . .	30
4.4	Aufbau und Ausführung des Prompts . . . . .	31
4.5	Erhobene Metriken . . . . .	34
<b>5</b>	<b>Ergebnisse</b>	<b>35</b>
5.1	Andere Befunde . . . . .	37
<b>6</b>	<b>Diskussion</b>	<b>38</b>
<b>7</b>	<b>Fazit</b>	<b>42</b>
7.1	Limitation und zukünftige Forschung . . . . .	42
<b>8</b>	<b>Anhang</b>	<b>52</b>



# 1 Einleitung

Eine öffentliche Ausschreibung ist ein Verfahren, bei dem staatliche oder öffentlich-rechtliche Stellen Aufträge öffentlich bekannt machen. Sie verfolgen das Ziel im Rahmen eines transparenten und wettbewerblichen Verfahrens Angebote von mehreren Unternehmen einzuholen. Öffentliche Ausschreibungen sind ein zentraler Bestandteil und aus mehreren Gründen ein wichtiges Vorgehen in der öffentlichen Beschaffung.

So wird sichergestellt, dass alle potenziellen Anbieter über geplante Aufträge gleichermaßen informiert werden. Das reduziert das Risiko von Korruption, Bereicherung und Misswirtschaft, weil sich der gesamte Prozess öffentlich und nachvollziehbar gestaltet. Durch die öffentliche Bekanntmachung von Aufträgen haben alle qualifizierten Unternehmen, unabhängig von ihrer Größe oder ihrem Standort, die Möglichkeit, sich zu beteiligen. Das gewährt einen fairen Marktzugang und sorgt für maximale Wirtschaftlichkeit. Je mehr Unternehmen ein Angebot abgeben, desto höher ist die Wahrscheinlichkeit, dass ein optimales Preis-Leistungs-Verhältnis erzielt wird. Der Staat oder die öffentlichen Institutionen erhalten die bestmögliche Leistung für die eingesetzten Mittel, was zu einem effizienten Einsatz öffentlicher Gelder führt.

Öffentliche Ausschreibungen können als Katalysator für Innovationen wirken. Durch die Förderung eines breiten Wettbewerbs und die Öffnung der Märkte für neue Anbieter wird ein Anreiz geschaffen, innovative Lösungen zu entwickeln und anzubieten. Langfristig kann das dazu beitragen, die Qualität der öffentlichen Dienstleistungen zu verbessern und auf neue Standards zu setzen. Insgesamt sind öffentliche Ausschreibungen also ein wichtiges Instrument, um Transparenz, Effizienz, Fairness und Innovation im öffentlichen Sektor zu fördern [5].

In Deutschland gibt es zahlreiche Online-Vergabeportale für öffentliche Ausschreibungen, die von Bund, Ländern, Kommunen und verschiedenen Institutionen genutzt werden. Insgesamt existieren in Deutschland mindestens 20 solcher Portale [14]. Jedes Portal hat seine eigene Struktur und Nutzeroberfläche in Bezug auf Inhalt, Design und Integration mit anderen Systemen, wodurch die Darstellung sogar innerhalb eines Vergabeportals variieren kann. Das macht es für Interessenten schwierig wichtige Entitäten wie den Auftraggeber, die Vergabestelle oder diverse Ausschreibungsfristen schnell zu ermitteln. Hinzu kommen Lose, welche den Gesamtauftrag in mehrere kleinere, in sich abgeschlossene Einheiten oder Abschnitte gliedern und individuell betrachtet werden müssen.

Um Entitäten nicht manuell extrahieren zu müssen, bietet es sich an, Ansätze aus dem Bereich des NLP mit künstlicher Intelligenz zu verwenden. NER ist eine Methode aus dem Bereich des NLP, die bereits in Domänen wie der Biomedizin [71] oder Astronomie [49] Anwendung findet. Häufig werden dazu LLMs wie BERT feinabgestimmt. Das ist kostspielig und aufwändig [11]. Zudem ist die verfügbare Datenmenge, anhand der ein Modell trainiert werden könnte, im Bereich öffentlicher Ausschreibungen wesentlich geringer.

Neuere Ansätze weisen darauf hin, dass eine hohe Modellleistung bzw. State-of-the-Art (SOTA) nahe Leistung auch rein durch das Design eines geeigneten Prompts erreicht werden kann, wodurch das Modell alle zur Lösung notwendigen Elemente im Rahmen des Prompts erhält. Reine Decoder-Modelle wie GPT sind in einigen Szenarien kostengünstiger zu verwenden als Modelle, die sowohl einen Encoder als auch einen Decoder nutzen (oder reine Encoder-Modelle wie BERT), da sie mit minimalem Fine-tuning und Anpassungen in der Lage sind eine breite Palette von Aufgaben zu bewältigen [9, 37].

Diese Modelle berücksichtigen den gesamten Eingabetext, um sinnvolle Antworten oder Fortsetzungen zu erzeugen. Das Modell analysiert nicht nur einzelne Wörter oder Phrasen isoliert, es achtet auch darauf, wie jedes Wort im Kontext zu anderen Wörtern steht – sogenannte *Self-Attention* (Selbst-Aufmerksamkeit).

Um NER im Bereich öffentlicher Ausschreibungen in Zukunft mit Hilfe von GPT zuverlässig und wirtschaftlich durchzuführen, ist folgende Frage zu beantworten: Wie lässt sich die Leistung von GPT für die NER in öffentlichen Ausschreibungen durch gezielte Anpassung und Feinabstimmung des Prompts verbessern?

In dieser Arbeit werden verschiedene Ansätze zur optimierten Gestaltung von Prompts untersucht, die zu einer Effizienzsteigerung der NER in öffentlichen Ausschreibungen durch das Modell gpt-4o-mini von OpenAI führen können.

In Kapitel 2 gehen wir auf die theoretischen Grundlagen ein. Dazu zählt der generelle Einsatz von Language Models (LMs) und die Funktionsweise reiner Decoder-Modelle wie GPT, sowie Grundlagen aus dem Bereich des Prompt Engineering, welche diverse Strategien und das Design eines Prompts umfassen. Zudem gehen wir in diesem Kapitel auf kontextuelles Lernen ein.

In Kapitel 3 wird eine Übersicht des aktuellen Stands der Wissenschaft in Bezug auf NER und dessen Einsatz aus evolutionärer Sicht gegeben. Wir betrachten den domänenübergreifenden Einsatz von Few-Shot Ansätzen für NER in verschiedenen Domänen. Hier wird außerdem auf den semantischen und syntaktischen Aufbau öffentlicher Ausschreibungen auf Online-Vergabeportalen eingegangen.

In Kapitel 4 folgt eine Beschreibung des Versuchsaufbaus und diverser zu untersuchender Prompts, die auf Grundlage vorangegangener Forschung und ermitteltem Wissen designed wurden.

Die Ergebnisse, die beim Prompten des OpenAI Modells gpt-4o-mini zur Ermittlung relevanter Entitäten in öffentlichen Ausschreibungen erzielt wurden, betrachten wir näher in Kapitel 5, werten diese aus und setzen sie in Kapitel 6 mit dem ermittelten Stand der Wissenschaft in Bezug.

In Kapitel 7 fassen wir die Ergebnisse aus der vorangegangenen Diskussion sowie die wichtigsten Erkenntnisse zur Optimierung von Prompts zur Effizienzsteuerung von NER in öffentlichen Ausschreibungen durch das Modell gpt-4o-mini von OpenAI zusammen. Zuletzt werden Limitationen und zukünftige Forschungsmöglichkeiten in diesem Bereich aufgezeigt.

## 2 Theoretische Grundlagen

Im Folgenden wird auf Begrifflichkeiten und Konzepte eingegangen, die als Grundlage für die durchgeführte Untersuchung relevant sind. Neben einem Grundverständnis für LMs, wird auf das Thema Prompt Engineering sowie die wirtschaftliche Relevanz öffentlicher Ausschreibungen eingegangen.

### 2.1 Sprachmodelle

Sprache spielt eine fundamentale Rolle in der Kommunikation zwischen Menschen und ihrer Interaktion mit Maschinen. Es besteht zunehmend der Bedarf Modelle zu entwickeln, die komplexe Aufgaben wie Übersetzung, Zusammenfassung oder Extraktion von Wissen aus Texten in natürlicher Sprache übernehmen. LMs sind Computermodelle, die Texte verarbeiten und erzeugen können. Dies wird als NLP bezeichnet.

Während man mit statistischer Sprachmodellierung begann, stehen heute neuronale Ansätze im Vordergrund der Forschung. Zudem entwickelten sich die Modelle von vortrainierten Sprachmodellen (Pre-trained Language Models (PLMs)) bis hin zu LLMs [32]. LLMs, wie GPT-3 oder PaLM, übertreffen einige PLMs, wie BERT oder T5, insbesondere in der Anzahl genutzter Modellparameter (dutzende bis hunderte von Milliarden) sowie enorme Trainingsdatensätze von vielen Giga Bytes bis Terra Bytes [39].

Bei konventionellen Ansätzen werden aufgabenspezifische Modelle mit *supervised* (kontrollierten) Einstellungen trainiert, während PLMs selbstüberwacht und mit großen Textkörpern trainiert werden. Ihr Ziel ist es eine generische Repräsentation zu erlernen, die anschließend für verschiedene Aufgaben im Bereich des NLP verwendet werden können [32]. Diese Entwicklung von allgemeinem Wissen ermöglicht es dem Modell Text zu „verstehen“. Es kann von einfachem Wissen (zum Beispiel die Schreibweise oder Bedeutung von Wörtern) bis hin zu einem hohen Niveau von Wissen (zum Beispiel dass eine Tuba zu groß ist, um in die meisten Rucksäcke zu passen) reichen [40]. Durch Fine-tuning können Modelle nachgelagert auf bestimmte Aufgaben angepasst werden [32].

#### 2.1.1 Funktionsweise reiner Decoder-Modelle

Mehrere Aufgaben zur Generierung natürlicher Sprache, wie die Übersetzung oder Zusammenfassung von Texten in natürlicher Sprache fallen unter die Kategorie der seq2seq-Aufgaben. Das Seq2Seq-Modell (Sequence-to-Sequence-Modell) ist eine Architektur des Deep Learning, die zur Umwandlung einer Eingabesequenz in eine Ausgabesequenz genutzt wird. Sie basiert auf einem Encoder-Decoder-Ansatz, wobei der Encoder die Eingabesequenz in eine abstrakte Repräsentation überführt und der Decoder diese Repräsentation zur Generierung der Ausgabesequenz verwendet. Ein klassisches Beispiel ist die neuronale maschinelle Übersetzung, bei der ein

Seq2Seq-Modell einen Satz in einer Sprache als Eingabe erhält, ihn im Encoder verarbeitet und dann im Decoder in eine andere Sprache übersetzt. Durch die Integration von Mechanismen, wie der Aufmerksamkeitsfunktion (Attention), fokussiert das Modell relevante Teile der Eingabesequenz. Oft basieren seq2seq-Frameworks auf einer Encoder-Decoder Architektur (ED-Frameworks), bei der ein Encoder für die Kodierung der Daten und der Decoder für die Generierung des Ziel-Ausgabetexts verantwortlich ist [3]. Dennoch werden häufig reine Decoder-Modelle, sogenannte auto-regressive Transformer wie GPT-2, GPT-3 oder Llama zur Lösung von seq2seq-Aufgaben eingesetzt. Ein autoregressives Decoder-Modell arbeitet, indem es jedes neue Token, basierend auf den zuvor generierten Tokens, vorhersagt. So wird schrittweise zusammenhängender Text erzeugt, ohne dass zwingend ein Encoder zur Kodierung notwendig ist.

Angenommen, wir wollen den Satz „Es war einmal...“ ergänzen, dann würde der Encoder diesen Satz vollständig analysieren und eine interne Repräsentation erzeugen. Erst dann erzeugt der Decoder eine Ausgabe. Ein reines Decoder-Modell arbeitet ohne den Encoder, indem es den bisherigen Kontext zwar berücksichtigt, ohne jedoch zuvor eine vollständige Repräsentation der Eingabe zu erzeugen. Ein Token kann ein vollständiges Wort, ein Wortteil oder auch nur ein einzelnes Zeichen sein. Die Vorhersage jedes neuen Tokens erfolgt dabei durch Wahrscheinlichkeitsberechnungen, die aus den Trainingsdaten gewonnen wurden.

Während des Trainings lernen die Decoder-Modelle das nächste Token in der Sequenz aus den vorherigen Token vorherzusagen, indem sie eine Maximum Likelihood Estimation (MLE) verwenden. Dieser Prozess ermöglicht es dem Modell, Text zu generieren, indem es während der Inferenz aus der gelernten Verteilung der Token-Wahrscheinlichkeiten Stichproben zieht [37]. Beispielsweise könnte ein Sprachmodell beim Satzanfang „Die Sonne...“ verschiedene Fortsetzungen generieren (zum Beispiel „geht bald auf.“ oder „geht unter.“), indem es Wahrscheinlichkeiten für mögliche Folgewörter berechnet und sie auswählt.

Kern von Decoder-Modellen ist die *Self-Attention* (Selbstaufmerksamkeit), die es dem Modell ermöglicht, bei der Vorhersage des nächsten Tokens die Bedeutung vorgegangener Token in der Sequenz abzuwägen. Decoder-Modelle sind in der Regel aus mehreren Schichten aufgebaut, die Selbstaufmerksamkeitsmechanismen und Feed-Forward-Netzwerke umfassen. Jede Schicht hilft, die Vorhersagen des Modells zu verfeinern, indem sie die beobachteten Informationen und erlernten Muster verarbeitet [58].

Reine Decoder-Modelle können mit unbeschrifteten Textdaten vortrainiert werden. Die Trainingsdaten sind so leichter zu beschaffen und zu warten [17]. Gelabelte Daten sind Daten, die mit zusätzlichen Informationen versehen sind. Das kann eine Rezension sein, die als positiv oder negativ gelabelt ist oder auch ein Zeitungsartikel mit Kennzeichnung als Sportartikel.

## 2.2 Prompt Engineering

Ein Prompt ist eine Texteingabe, die einem Sprachmodell zur Verfügung gestellt wird, um eine Antwort zu initiieren [9]. Er gibt den Kontext oder die Richtung für die Ausgabe des Modells vor [38]. Prompts sind eine praktikable Strategie, um Modelle in die Lage zu versetzen, sich schnell an neue Domänen und Aufgaben anzupassen [49].

Prompt Engineering ist der Prozess der Entwicklung und Verfeinerung dieser. Reynold und McDonell, 2021, beschreiben Prompt Engineering als „programming in natural language“ [41]. Aufgabe eines Prompt Engineers ist es, Design und Inhalt des Prompts zu gestalten [26].

Das Designen geeigneter Prompts ist entscheidend für die Verbesserung der Leistung von Modellen im Bereich der künstlichen Intelligenz (KI), insbesondere bei der Verarbeitung natürlicher Sprache. Ein Prompt Engineer verfolgt das Ziel Prompts zu erstellen, die das Modell anleiten, nützliche, relevante und genaue Ergebnisse zu erzeugen [9]. Durch Trial and Error Prinzipien werden Prompts angepasst, um bessere Ergebnisse zu erzielen. Dazu kann es gehören, den Wortlaut, die Struktur oder den Kontext der Aufforderung iterativ anzupassen [41].

Gut konzipierte Prompts können die Qualität der Antworten des Modells verbessern und es für bestimmte Aufgaben nützlicher machen [18]. Beispielsweise kann das Einbeziehen eines Kontexts bei der Eingabe eines Prompts dem Modell helfen die Aufgabe besser zu verstehen und so zu präziseren Antworten führen [26].

### 2.2.1 Prompt-Strategien

Es wird zwischen Zero-Shot, One-Shot und Few-Shot unterschieden. Von Zero-Shot Transfer wird oft auch gesprochen, wenn keine Aktualisierung von Gradienten erfolgt und das Modell die Aufgabe mit Hilfe von gegebenen Beispielen aus dem Prompt löst [9]. Wir bezeichnen Shots als Übungsbeispiele, die im Prompt enthalten sind:

- **Zero-Shot:** Dem Prompt werden keine Beispiele beigefügt.
- **One-Shot:** Dem Prompt wird genau ein Beispiel beigefügt.
- **Few-Shot:** Dem Prompt werden mehr als ein Beispiel beigefügt.

Eine Zugabe von Übungsbeispielen eignet sich insbesondere dann, wenn eine Aufgabe besser durch Beispiele beschrieben werden kann, als es durch eine Umschreibung der Aufgabe möglich wäre, oder wenn ohnehin viele gute Lehrbeispiele zur Verfügung stehen [41].

Instructional Prompts definieren klar die Aufgabe wie beispielsweise „fasse diesen Absatz zusammen“, wodurch keine weiteren Informationen notwendig sind, um die Aufgabe zu lösen [46]. Durch einen „Signifier“ wird ein Muster beschrieben, dass das beabsichtigte Verhalten beschreibt und implizit eine Funktion, wie beispielsweise „übersetze diesen Text (translate)“ oder „sortiere der Größe nach

(sort)“, aufruft. Es werden weder Beispiele noch der Lösungsweg mitgegeben, da angenommen wird, dass das Modell die aufgerufene Funktion bereits erlernt hat [41].

Hinzu kommen diverse spezielle Prompt-Strategien wie das Chain of Thought (CoT), bei dem das Modell dazu aufgefordert wird Zwischenschritte beim Lösungsprozess auszugeben und die Aufgabe so in kleine logische Schritte aufzubrechen [62]. Diese Strategie eignet sich bei Aufgaben, die in mehreren Schritten gelöst werden müssen, wie unter anderem mathematische Berechnungen [62] oder logische Herleitungen [24].

Weitere Strategien sind das Self-Consistency Prompting, bei dem das Modell mehrere Antworten auf die selbe Aufgabe erzeugt und anschließend evaluiert, welche Antwort die konsistenteste ist [60].

Des Weiteren können mit Hilfe von kontextueller Kalibration laut Zhao et al., 2021, Biases in der Antwort reduziert werden [69]. Beispielsweise könnte man ein Modell bei der Erstellung einer Ausschreibung für Bewerber anweisen explizit geschlechterneutrale Sprache zu verwenden.

Zusätzlich haben Lewis et al., 2020, Retrieval-Augmented Generation (RAG) für Aufgaben mit hoher Informationsdichte vorgestellt. Bei dieser Strategie wird ein Retriever verwendet, um einen großen Satz an Dokumenten nach relevanten Informationen zu durchsuchen und diese anschließend einem Generator zu übergeben, der die Daten synthetisiert, um genaue und kontextbezogene Antworten zu erzeugen [25].

Beim Plan and Solve Prompting wird das Modell zuerst angewiesen einen detaillierten Plan zur Lösung der Aufgabe vorzulegen (Planning Prompt). In einem folgenden Solving Prompt wird das Modell angewiesen diese nach vorgeneriertem Plan zu lösen [62].

Des Weiteren nutzen wir in der menschlichen Kommunikation Analogien, in denen ein memetisches Konzept, beispielsweise ein Charakter, als Stellvertreter für eine Absicht dient. Laut Reynolds und McDonell, 2021, hat GPT-3 die Fähigkeit bekannte Figuren wie Mahatma Gandhi zu simulieren, was beispielsweise bei moralischen Fragen Zugang zu verschiedenen Biases und kulturellen Informationen bietet.

Zudem stellen Reynolds und McDonell, 2021, Metaprompt Programming vor, durch welches Modelle mit Hilfe von sogenannten Metaprompts befähigt werden, selbstständig weitere nützliche Prompts zur weiteren Verwendung für die Lösung der jeweiligen Aufgabe zu generieren. Die Nutzung von Metaprompts kann Modelle befähigen durch eigene aufgabenspezifische Instruktionen ganz ohne Übungsbeispiele effektiv Probleme zu lösen [41]. Beispiele für einen Metaprompt wären „Lösen wir das Problem, indem wir es in mehrere Schritte aufteilen.“, „Liste die Vor- und Nachteile auf, bevor du eine Entscheidung triffst.“ oder „Stelle Fragen zum Thema, bevor du versuchst, die Frage zu beantworten“.

Bei der Take A Deep Breath Strategie wird das Modell angewiesen nach der Generierung einer Antwort eine Pause einzulegen, zu reflektieren und die initiale Ant-

wort zu verfeinern, bevor es die finale Antwort ausgibt [33].

2024 zeigten Yugeswardeenoo et al. die Leistungsfähigkeit von Question-Analysis Prompting (QAP) für diverse Argumentationsaufgaben für GPT-3.5 und GPT-4 von OpenAI Turbo. Kern von QAP ist das Zerlegen in Komponenten und Analysieren der Fragestellung, wodurch das Modell angeleitet wird jede einzelne Komponente der initialen Fragestellung bei der Antwort zu berücksichtigen [67].

Verschiedene Prompt-Strategien können bei verschiedenen Modellen zu mehr oder weniger guten Leistungen führen. Es wurden die nach SOTA relevantesten Prompting-Strategien zusammengefasst, jedoch gibt es noch weitaus mehr Ansätze für erfolgsversprechende Prompt-Strategien.

## 2.2.2 Kontextuelles Lernen

Menschen schaffen es neue Sprachaufgaben mit einer kleinen Anzahl von Beispielen und einfachen Anweisungen zu bewältigen. Im Gegensatz dazu brauchen vortrainierte NLP-Systeme zur Bewältigung solcher Aufgaben eine Feinabstimmung auf eine konkrete Aufgabe mit mehreren tausend Beispielen [9]. Seit geraumer Zeit werden vortrainierte Transformer LLMs direkt feinabgestimmt, sodass eine aufgabenspezifische Architektur nicht mehr erforderlich ist. Damit sind sie aufgabengnostisch. Das heißt, sie sind in der Lage diverse Aufgaben zu lösen, die ihnen aufgegeben werden. Erhält ein Modell eine NLP-Aufgabe sowie gegebenenfalls zusätzlichen Kontext und Trainingsbeispiele ausschließlich durch einen Prompt und löst diese mit den gegebenen Informationen, spricht man von ICL beziehungsweise Prompt-basiertem Lernen. Eine zentrale Komponente beim ICL ist der Prompt [9].

ICL ist eine Form des Few-Shot-Lernens. Bei dieser Form des Lernens werden Modell-Parameter nicht nachträglich angepasst. Das Modell muss flexibel mit vielen verschiedenen NLP-Aufgaben umgehen, ohne vorher intensiv darauf trainiert zu werden, was enorme Einsparungen bei der Datenannotation und den technischen Betrieb bringen kann [21]. Die Feinabstimmung eines Modells bedarf häufig hunderter von Grafikprozessoren, was zeitaufwändig und teuer ist, sowie der Speicherung von Modellprüfpunkten, Zustände des Modells zu verschiedenen Zeitpunkten des Trainings [26].

Menschen benötigen oft nur wenige Beispiele für eine Art von Aufgabe, um eine neue Aufgabe mit angemessener Kompetenz zu lösen [9]. Lu et al. konnten 2021 zeigen, dass auch GPT-2XL mit nur 1,5B zu über 90 Prozent zutreffende Ergebnisse mit nur vier gegebenen Übungsbeispielen in einem Few-shot Prompt liefert. Sie liefern teilweise vergleichbare Ergebnisse zu überwachten Modellen, die mit über 60.000 Übungsbeispielen trainiert wurden. Dennoch bedeuten starke Varianzen in der Leistung verschiedener Konfigurationen eines Modells ein großes Problem, da insbesondere „kleinere“ Modelle nicht zuverlässig sind [28]. Perez et al., 2022, haben bei größeren Modellen wie GPT-3 (6,7B) schlechtere bis gleichbleibende Ergebnisse beim Einsatz von Few-Shot Prompts mit der maximalen Menge an Beispielen aufzeigen können und von einer Notwendigkeit mehrerer tausend Beispiele gespro-

chen, um überdurchschnittliche Ergebnisse zu erzielen [46]. Einige Forscher argumentieren, dass die Leistung des Modells stark vom Design des Prompts abhängt und bei entsprechend passenden Übungsbeispielen in der richtigen Reihenfolge sogar zu besseren Ergebnissen führen kann [70, 28].

Ein Prompt kann entweder so gestaltet werden, dass er die maximale Leistung eines konkreten Modells gewährleistet oder das Modell wird darauf trainiert flexibler auf verschiedene Prompts zu reagieren, sogenannte Prompt-basierte Feinabstimmung [46]. Bei ersterem hat die Struktur des gewählten Templates für den Prompt einen signifikanten Einfluss auf das Ergebnis [46].

### 2.2.3 Prompt Design

Gut gestaltete Prompts können die Performanz eines Modells maßgeblich beeinflussen [70]. Leistungsstarke Prompts sind nicht modellübergreifend gleichbleibend performant. Diverse Prompts schneiden im Einsatz bei verschiedenen Modellen unterschiedlich gut ab [28]. Die zentrale Herausforderung bei der Entwicklung von performanten Prompts besteht darin, Trainingsdaten, insofern vorhanden, in eine Textsequenz umzuwandeln.

Handgefertigte Templates erfordern erheblichen Arbeitsaufwand und führen nicht immer zu optimalen Leistungen, wodurch die automatisierte Erstellung von Prompts [46, 28] mittels separater Modelle [41] an Aufmerksamkeit gewonnen hat. Gao et al., 2021, benutzen beispielsweise ein zusätzliches Modell zur Generierung von Templates [18], während Shin et al., 2020, ein selbstentwickeltes Tool namens AutoPrompt auf Grundlage gradientengeführter Suche zur automatischen Erstellung von Templates für diverse NLP Aufgaben für die maximale Leistung eines Modells einsetzen [50].

Um den performantesten Prompt aus einer Menge von alternativen Prompts zu erkennen, werden, wie Lu et al., 2022, beschreiben, verschiedene Strategien zur Prompt-Selektion beziehungsweise zum Ranking diverser Prompts, wie unter anderem Entropy-based Probing, angewandt. Bei Entropy-based Probing wird die Entropie der Antwortverteilung für jeden alternativen Prompt berechnet. Entropie misst die Unsicherheit in der Antwortverteilung eines Sprachmodells, wobei eine höhere Entropie auf größere Variabilität und eine niedrigere Entropie auf konsistentere Antworten hinweist. Prompts mit moderater Entropie sind oft am nützlichsten, da sie informative und konsistente Antworten hervorrufen, ohne zu restriktiv zu sein. Der Prozess zur Selektion des performantesten Prompts wird häufig automatisiert durchgeführt [28].

Während bei vortrainierten Modellen mit mehreren tausend Beispielen und großen Datensätzen die Varianz keine so große Rolle mehr spielt und die Trainingsdaten bereits im Prozess bereinigt werden, ist bei der Erstellung handgefertigter Prompts ein präzises Design ausschlaggebend, um das Modell zu maximaler Leistung zu befähigen.

In den letzten Jahren stieg die Kapazität von Transformern von 100 Millionen 2018

[37] auf 530 Milliarden Parameter (Megatron-Turing NLG von NVIDIA Microsoft) [55] an. Durch eine höhere Anzahl von Parametern und ein breiteres Spektrum an Wissen und Beispielen, die den Modellen in der initialen Trainingsphase zur Verfügung stehen, gewinnen Modelle an Flexibilität bei der Anpassung an verschiedene Kontexte und Eingabearten. Daher geht man mit dem Anstieg an Parametern auch von einem Anstieg der Leistungsfähigkeit beim kontextuellen Lernen aus [9].

Um einen performanten Prompt für ICL zu entwerfen bedarf es Fachwissen über die Domäne sowie Verständnis für die innere Funktionsweise des verwendeten Sprachmodells [18]. Das Erstellen eines Prompts erfordert Textstandardisierung mit Hilfe von Vorlagen und die Integration von Übungsbeispielen [9]. Unter `www.promptingguide.ai` [10] werden Bausteine zusammengefasst, die ein Prompt enthalten kann.

Unter einer **Instruktion** versteht man einen klaren und prägnanten Befehl, der dem Modell sagt, was es tun soll (zum Beispiel „Übersetze folgenden Text ins Spanische“). Der **Kontext** liefert zusätzliche Hintergrundinformationen oder Details, wie beispielsweise zusätzliche Informationen zu einzelnen Entitäten bei der NER.

**Übungsbeispiele** sind spezifische Input-Output-Paare, die das gewünschte Ergebnis veranschaulichen. Brown et al. konnten 2021 zeigen, dass die Genauigkeit des Ergebnisses bei verschiedenen NLP Aufgaben durch ICL beim Einsatz eines Prompts in natürlicher Sprache mit der gegebenen Anzahl an Übungsbeispielen steigt. Das untersuchte Modell war GPT-3 und es wurde keine vorherige Feinabstimmung vorgenommen [9]. 2021 untersuchten Lu et al. inwiefern die Anordnung von Übungsbeispielen eine Rolle bei der Performanz spielt, indem sie in einem Prompt mit vier Übungsbeispielen die Reihenfolge dieser variierten, wodurch 24 potentielle Prompts entstanden. Sie konnten zeigen, dass die Reihenfolge, in der die Beispiele übergeben wurden, eine Auswirkung auf die Leistung des Modells in Bezug auf mehrere Aufgabenarten aus SuperGLUE (General Language Understanding Evaluation) hat [28]. Die SuperGLUE Benchmark ist eine Sammlung anspruchsvoller Aufgaben aus dem Bereich des NLP [53]. Daher bleibt beim Einsatz von Übungsbeispielen zu beachten, dass diese innerhalb eines Prompts im Gegensatz zum Einsatz von Beispielen bei der Feinabstimmung als Ganzes verarbeitet werden und nicht grundsätzlich als parallel und voneinander unabhängig interpretiert werden [41].

**Aufgabenschwerpunkte** enthalten detaillierte und beschreibende Aufforderungen, die das Modell genauer anleiten (zum Beispiel „Schreibe eine Stellenausschreibung und achte darauf, dass der Ton förmlich ist.“).

Um zwischen einzelnen Sektionen in einem Prompt zu unterscheiden, können **Separatoren** (zum Beispiel „### Sektion 1 ###“) verwendet werden.

Es kann ein **Ausgabeindikator** angegeben werden, der das erwartete Format oder die Struktur der Antwort enthält (zum Beispiel Eingabe: „Orte: [Liste der Orte]“, erwarteter Output: „Orte: Paris, London, Düsseldorf“).

Hinzu kommen Aspekte die beim Design eines Prompts beachtet werden sollten. Prompts können eine maximale Anzahl von Tokens enthalten. GPT-basierte Model-

le können nur eine Anzahl von Beispielen entgegen nehmen [36]. 100 Tokens entsprechen circa 75 Wörtern in der englischen Sprache. Abhängig vom GPT-Modell können hier bis zu 128.000 Tokens für einen Prompt und eine entsprechende Antwort verteilt werden [35]. Reynolds und McDonell haben 2021 Prompts aus der Perspektive von natürlicher Sprache und der Formulierung von Aufforderungen in dieser evaluiert. Bei dem Design eines Prompts müssen dieselben Überlegungen in Bezug auf unter anderem Tonfall, Implikation, Plausibilität, Stil und Mehrdeutigkeit angestellt werden, wie bei Aufforderungen an andere Menschen, da Modelle wie GPT-3 auf Grundlage von natürlicher Sprache trainiert werden.

### **2.3 Wirtschaftliche Relevanz öffentlicher Ausschreibungen**

Öffentliche Ausschreibungen sind ein wichtiges Instrument für öffentliche Einrichtungen, um Wettbewerb zu fördern, Innovationen anzuregen und Transparenz zu schaffen. Sie haben weitreichende Auswirkungen auf das Wirtschaftswachstum und die Entwicklung einzelner Branchen. Besonders für kleine und mittlere Unternehmen bieten öffentliche Ausschreibungen eine wichtige Möglichkeit Aufträge zu gewinnen. Dies trägt zum Wachstum dieser Unternehmen bei und steigert so zukünftigen Wettbewerb. Die Dominanz von großen Unternehmen kann im Rahmen gehalten werden [2, 12].

Durch öffentliche Ausschreibungen können öffentliche Institute gezielt in verschiedene Projekte, insbesondere im Bereich der Infrastruktur, investieren und so das Wirtschaftswachstum fördern. Eine starke Wirtschaft fördert private Investitionen und schafft Arbeitsplätze. Sie können auch als Instrument zur Förderung von Innovation genutzt werden. Dabei werden Unternehmen gezielt zur Entwicklung von spezifischen technologischen Lösungen oder innovativer Ansätze beauftragt [6, 13]. Öffentliche Ausschreibungen sorgen für mehr Transparenz im Ausschreibungsprozess und wirken Korruption entgegen. Ein transparenter Ausschreibungsprozess fördert das Vertrauen in öffentliche Institutionen und sorgt für Rechtssicherheit bei allen Beteiligten [34, 42].

In Branchen wie dem Baugewerbe oder der Infrastrukturentwicklung sind Ausschreibungen oft der zentrale Mechanismus für die Vergabe großer Projekte. Sie beeinflussen die wirtschaftliche Entwicklung ganzer Regionen, indem sie die Grundlage für den Ausbau von Verkehrsinfrastruktur, Energieversorgung und öffentlichen Gebäuden legen. In der Technologie und IT-Branche haben öffentliche Ausschreibungen eine wesentliche Rolle und dienen der Beschaffung von Technologien, Software und Dienstleistungen für verschiedene öffentlich-rechtliche Einrichtungen. Im Gesundheitswesen dienen öffentliche Ausschreibungen oft zur Beschaffung von Medizintechnik, Arzneimitteln und Dienstleistungen. Die Qualität und Verfügbarkeit der Gesundheitsversorgung werden durch öffentliche Ausschreibungen direkt beeinflusst [30, 20]. Ein transparenter und effizienter Ablauf ist daher unabdingbar.

### 2.3.1 Elektronische Ausschreibungsportale

In diesem Abschnitt werden verschiedene elektronische Vergabeportale vorgestellt, einschließlich der verantwortlichen Organisationen, des jeweiligen Fokus sowie der spezifischen Funktionen im Zusammenhang mit dem Ausschreibungsmanagement. Öffentliche Auftraggeber, darunter Kommunen, Städte und staatliche Einrichtungen, betreiben in der Regel eigene Vergabeportale (Vergabemarktplätze), um die Effizienz und Transparenz des Beschaffungsprozesses zu erhöhen. Diese Portale bieten eine zentrale Anlaufstelle für die Ausschreibung und Vergabe öffentlicher Aufträge und tragen maßgeblich zur Digitalisierung des öffentlichen Sektors bei. Eine Übersicht einiger Portale und ihrer Funktionen ist in Tabelle 1 zu finden.

Organisation	Link	Beschreibung
Tenders Electronic Daily (TED)	<a href="https://ted.europa.eu">https://ted.europa.eu</a>	TED ist das offizielle Ausschreibungsportal der Europäischen Union, auf dem öffentliche Ausschreibungen aus allen EU-Mitgliedsstaaten veröffentlicht werden.
Bundesanzeiger	<a href="https://www.bundesanzeiger.de">https://www.bundesanzeiger.de</a>	Der Bundesanzeiger ist das offizielle Verlautbarungsorgan der deutschen Bundesregierung, in dem auch Ausschreibungen veröffentlicht werden.
Deutsches Vergabeportal (DTVP)	<a href="https://dtvp.de">https://dtvp.de</a>	Das Deutsche Vergabeportal bietet eine zentrale Plattform für öffentliche Ausschreibungen in Deutschland, insbesondere für öffentliche Aufträge und Vergabeverfahren.

<b>Organisation</b>	<b>Link</b>	<b>Beschreibung</b>
Subreport	<a href="https://www.subreport.de">https://www.subreport.de</a>	Subreport ist ein deutsche Online-Portal für Ausschreibungen und Vergabeverfahren, die sowohl öffentliche als auch private Aufträge abdeckt.
eVergabe.de	<a href="https://www.evergabe.de">https://www.evergabe.de</a>	eVergabe.de ist das zentrale Portal für elektronische Ausschreibungen und die Abwicklung von Vergabeverfahren in Deutschland.
Vergabe24	<a href="https://www.vergabe24.de">https://www.vergabe24.de</a>	Vergabe24 bietet eine umfassende Datenbank von öffentlichen Ausschreibungen in Deutschland und ist eine wichtige Quelle für Bieter.
evergabe-online.de	<a href="https://www.evergabe-online.de">https://www.evergabe-online.de</a>	evergabe-online.de ist ein Portal, das ermöglicht, an öffentlichen Ausschreibungen teilzunehmen und Angebote online einzureichen.
service.bund.de	<a href="https://www.service.bund.de">https://www.service.bund.de</a>	Das Portal service.bund.de bietet eine Übersicht über verschiedene Verwaltungs- und Dienstleistungsangebote der deutschen Bundesregierung, einschließlich Ausschreibungen.

<b>Organisation</b>	<b>Link</b>	<b>Beschreibung</b>
Deutsche Bahn AG	<a href="https://bieterportal.noncd.db.de">https://bieterportal.noncd.db.de</a>	Das Vergabeportal der Deutschen Bahn AG ermöglicht die Teilnahme an Ausschreibungen im Bereich Verkehrsinfrastruktur und Dienstleistungen im Eisenbahnsektor.
Vergabemarktplatz Bergisch-Rheinischer Wasserverband / Wupperverband	<a href="https://vergabe.brw-haan.de">https://vergabe.brw-haan.de</a>	Der Vergabemarktplatz des Bergisch-Rheinischen Wasserverbands und des Wupperverbands dient zur Ausschreibung von Aufträgen im Bereich Wasserwirtschaft.
Vergabemarktplatz Bremen	<a href="https://vergabe.bremen.de">https://vergabe.bremen.de</a>	Der Vergabemarktplatz Bremen ist ein Portal für die Ausschreibung öffentlicher Aufträge in Bremen, das von der Stadtverwaltung genutzt wird.
Vergabemarktplatz Baden-Württemberg	<a href="https://ausschreibungen.landbw.de">https://ausschreibungen.landbw.de</a>	Der Vergabemarktplatz Baden-Württemberg ist das zentrale Portal für öffentliche Ausschreibungen in diesem Bundesland.
Vergabemarktplatz Köln	<a href="https://vergabe.stadt-koeln.de">https://vergabe.stadt-koeln.de</a>	Auf dem Vergabemarktplatz Köln werden Ausschreibungen der Stadt Köln und öffentlicher Stellen in der Region veröffentlicht.

<b>Organisation</b>	<b>Link</b>	<b>Beschreibung</b>
Vergabemarktplatz Rheinland	<a href="https://vmp-rheinland.de">https://vmp-rheinland.de</a>	Der Vergabemarktplatz Rheinland stellt öffentliche Ausschreibungen für die Region Rheinland zur Verfügung.
Vergabemarktplatz Mecklenburg-Vorpommern	<a href="https://vergabe.mv-regierung.de">https://vergabe.mv-regierung.de</a>	Der Vergabemarktplatz Mecklenburg-Vorpommern ist das offizielle Portal für Ausschreibungen im Bundesland Mecklenburg-Vorpommern.
GIZ Vergabemarktplatz	<a href="https://ausschreibungen.giz.de">https://ausschreibungen.giz.de</a>	Der GIZ Vergabemarktplatz ist das Ausschreibungsportal der Deutschen Gesellschaft für Internationale Zusammenarbeit (GIZ), die internationale Entwicklungsprojekte umfasst.
Vergabemarktplatz NRW	<a href="https://www.evergabe.nrw.de">https://www.evergabe.nrw.de</a>	Der Vergabemarktplatz NRW bietet einen Zugang zu öffentlichen Ausschreibungen aus Nordrhein-Westfalen.

Tabelle 1: Übersicht einiger elektronischer Ausschreibungsportale

### 3 Stand der Wissenschaft

Neben der Beantwortung von Fragen, der Zusammenfassung von Texten und der Erkennung von Fehlinformationen, ist eine wichtige Aufgabe im Bereich des NLP die Beziehungsextraktion. Dabei geht es darum hochwertige und semantische Beziehungen zwischen Entitäten in freien Texten zu erkennen [45]. Die manuelle Aufbereitung von Informationen aus der wissenschaftlichen Literatur, Datensätzen wie der elektronischen Gesundheitsakte [71], Zeitungsartikeln [68] und freien Texten ist zeitaufwändig, kostspielig und nicht ausreichend, um mit dem raschen Wachstum der Literatur Schritt zu halten [47, 45]. Traditionelle Methoden zur Extraktion von Wissensseinheiten aus Texten bedeuten einen hohen manuellen Aufwand, schlechtere Generalisierung und kostspieligere Wartung. Die Identifikation von Entitäten ist entscheidend für die Literatursuche, den Aufbau von Wissensgraphen [49] sowie den erfolgreichen Einsatz von Ansätzen im Bereich des Deep Learnings, deren Erfolg stark von großen Datensätzen mit präzisen Annotationen abhängt [68]. Daher gibt es ein wachsendes Interesse an der Verwendung von Techniken im Bereich des NLP für die automatische Extraktion von Beziehungen (Relation Extraktion (RE)) von und zwischen Entitäten aus Texten [71]. RE ist die Klassifizierung von Beziehungen von Entitäten innerhalb eines Satzes oder zwischen Sätzen [66].

Das Verknüpfen von Entitäten ist ein notwendiger Schritt, der auf der Erkennung und Klassifizierung von Entitäten - NER - aufbaut. Hierbei handelt es sich um den Prozess der automatischen Verknüpfung von Erwähnungen in Texten mit einer standardisierten Liste von Entitäten in einer Wissensdatenbank [71]. Das Benennen und Klassifizieren bekannter Entitäten ist relevant zur Kosten- und Zeitersparnis in fast jeder Domäne [47]. NER, auch bekannt als Entity Identification oder Entity Extraction, ist eine Teilaufgabe in der Informationsextraktion.

Ziel ist es atomare Elemente in Texten ausfindig zu machen und in vordefinierte Kategorien, wie Namen von Personen, Zeitangaben oder Orte, einzuordnen [51]. NER ist unter anderem relevant für die Zusammenfassung von Dokumenten, da relevante Entitäten und zugehörige Informationen als Kernaspekte identifiziert werden oder auch für das Social Media Monitoring in Bezug auf die Erkennung von Markennamen und Personen, was beim Krisenmanagement oder der Marktanalyse von Vorteil sein kann [23].

Obwohl es im allgemeinen Bereich zahlreiche Ansätze zur Verknüpfung von Entitäten gibt, können diese aufgrund der großen Unterschiede in den Sprachmerkmalen und Wissensbasen nicht direkt auf eine beliebige Domäne übertragen werden. Zwei große Probleme sind dabei die Vielfalt und Mehrdeutigkeit im Sprachgebrauch [71]. Während es bereits vor über zehn Jahren mit beispielsweise TaggerOne die ersten Werkzeuge zur Erkennung und Verknüpfung von Entitäten gab, waren diese vor allem regelbasierte Ansätze, die einer großen Diskrepanz zwischen Morphologien und Konzepten nicht standhalten können. Anschließend folgten verschiedene Ansätze aus dem Bereich des maschinellen Lernens [71]. Im Folgenden gehen wir auf diverse Ansätze beim Einsatz von NER in verschiedenen Domänen ein.

### **3.1 Ansätze in der Named Entity Recognition**

Die Extraktion und Klassifikation von benannten Entitäten findet bereits in vielen Domänen Anwendung. Erste systematische Ansätze für NER sind lexikon- und regelbasierte Methoden. Ein regelbasierter Ansatz für NER beginnt häufig mit einer lexikonbasierten Suche, bei der das System über eine Liste bekannter Ortsnamen wie „Frankfurt“, „Bremen“ oder „Chicago“ verfügt. Bei der Textanalyse wird jedes Wort mit der Liste abgeglichen, um mögliche Entitäten zu identifizieren. Ist ein Begriff nicht in der Liste enthalten, greifen regelbasierte Muster. Eine solche Regel könnte vorgeben, dass ein Wort nach Präpositionen wie „aus“ oder „gelaufen nach“ wahrscheinlich der Name eines Orts ist. Regelbasierte Systeme können komplex und mit zunehmender Anzahl von Regeln schwer wartbar werden, so dass sie nur mit erhöhtem Aufwand an neue Bereiche oder Situationen angepasst werden können [19].

Vor über zehn Jahren folgten Ansätze für NER im Bereich des Machine Learning beziehungsweise statistischen Lernens [8] wie das Hidden Markov Model (HMM), Maximum Entropy Markov Model (MEMM) und Conditional Random Fields (CRF). Sobhana et al. untersuchten 2010 NER in Bezug auf geologische Texte. Zu extrahierende geologische Entitäten waren in dieser Studie unter anderem Berge, Flüsse, Inseln und Mineralien. Während hier insbesondere Mehrdeutigkeit durch die mehrfache Verwendung gleicher Namen für beispielsweise Städte sowie multiple Namen für den gleichen Ort eine besondere Herausforderung darstellten. Sie verglichen Machine Learning-basierte Ansätze - hier CRF - mit regelbasierten NER-Systemen. Der Einsatz von Machine Learning Ansätzen wurde von den Forschern besser als regelbasierte NER Systeme unter Verwendung von Ortsverzeichnissen und lexikalisierte Grammatik bewertet, da sie keine Regeln qualifizierter Linguisten erfordern, nicht das Maß an Erfahrung über die Grammatik der jeweiligen Sprache haben und wesentlich einfacher an andere Domänen und Sprachen angepasst werden können. Im Gegensatz nutzt das System die verschiedenen kontextuellen Informationen der Wörter zusammen mit einer Vielzahl von Merkmalen, die bei der Vorhersage der verschiedenen Klassen von benannten Entitäten hilfreich sind [51].

2013 untersuchten Segura-Bedmar et al. Support Vector Machines (SVMs) zur Extraktion pharmakologischer Substanzen aus Texten. Extrahierte Entitäten waren hier Arzneimittel (generische Arzneimittelname), Marke (Markenarzneimittelname) und Gruppe (Arzneimittelgruppenname). Die ForscherInnen konnten zwar positive Ergebnisse durch den Einsatz von SVMs zeigen, sprachen aber auch von Verbesserungspotenzial [45].

Spätestens ab 2015 erfolgte ein Übergang zu Deep Learning Ansätzen. Hier haben sich insbesondere Methoden im Bereich der Recurrent Neural Networks (RNNs) und später Transformer-basierte Methoden bei den meisten NER-Aufgaben durchgesetzt [8]. Zhu et al., 2022, identifizieren als zwei große Probleme in NER die Vielfalt und Mehrdeutigkeit im Sprachgebrauch im Bereich der Biomedizin. So gibt es beispielsweise 79.609 Synonyme im Unified Medical Language System (UMLS), die auf zusätzliche Informationen angewiesen sind, um zwischen verschiedenen Entitäten zu unterscheiden. Ein Versagen bei der Disambiguierung kann zu einer Fehlinterpretation des gesamten Kontexts führen. Um die Ambiguität aufzulösen, untersuchen sie einen zweistufigen Entitätsverknüpfungsalgorithmus, um die Entitätsrepräsentationen auf der Grundlage von Prompt-Lernen zu verbessern. Mit Hilfe eines Bi-Encoder werden zuerst alle Kandidaten aus dem Repräsentationsraum abgerufen und daraufhin auf Grundlage des gegebenen Kontexts und weiteren Entitätsinformationen eingestuft, um die Verknüpfungsgenauigkeit zu verbessern und Mehrdeutigkeiten zu beseitigen [71]. In diesem Zusammenhang sind auch Kontextkodierungstechniken relevant, welche sich auf die Darstellung von Wörtern in ihrem umgebenden Kontext konzentrieren, um die Abhängigkeiten und Beziehungen zwischen benachbarten Wörtern zu berücksichtigen [23].

In einer Studie von Shao et al., 2023, wurden repräsentativ die vier LLMs Llama-2-70B, GPT-3.5, GPT-4 und Claude auf die Erkennung von Entitäten im Bereich der Astronomie untersucht. Sie untersuchten das Potential der LLMs die Bezeichnungen von Himmelsobjekten und Teleskopnamen durch Anwendung einer selbstentwickelten Prompt-KEE (Knowledge Entity Extraction)-Strategie aus astrophysikalischen Zeitschriftenartikeln zu extrahieren. Während auf Grundlage von Googles BERT bereits ein domänenspezifisches Modell namens astroBERT auf Basis von 395.499 astronomischen Forschungsartikeln, welches für NER in Astrophysics Data System (ADS) genutzt wird, wollten Shao et al. zeigen, dass vortrainierte Decoder-Modelle wie GPT-3.5 auf Dauer geeigneter seien. Sie begründen dies damit, dass solche Modelle mit einer entsprechenden Prompt-KEE-Strategie ebenso zuverlässige Ergebnisse liefern und es aufgrund der zunehmenden Spezialisierung und Vielfalt der astronomischen Wissensentitäten ineffizient und nicht nachhaltig ist, funktionale Extraktionsmodelle wie astroBERT mit Hilfe von zahlreichen Beispielen für jeden Entitätstyp zu trainieren. Sie konnten in ihrer Studie sowohl die Effektivität der Prompt-KEE-Strategie sowie das Potential vortrainierter Decoder-Modelle für KEE in astrophysikalischen Fachartikeln aufzeigen [49]. Bei der Prompt-KEE-Strategie setzt sich der Prompt aus einer Aufgabenbeschreibung und einer Beschreibung der zu labelnden Entitäten zusammen. Als dritter Bestandteil des Prompts kommt die

Beschreibung von Aufgabenschwerpunkten hinzu (zum Beispiel „Please ensure all entities from the paragraph have been extracted“). Darauf folgen Übungsbeispiele. Im Falle von Shao et al., 2023, sind das ein astrophysikalischer Fachartikel und zwei Listen, jeweils mit den extrahierten Teleskopnamen und Himmelskörpern. Als letzten Abschnitt des Prompts, der dem Modell nachträglich als separate Eingabe gegeben wird, dient eine Methode zur Selbstverifizierung, in dem das Modell angewiesen wird, das Ergebnis auf bestimmte Kriterien zu prüfen.

### 3.2 Domänenübergreifender Einsatz von Few-Shot Learning für Named Entity Recognition

Bei NER handelt es sich stets um dieselbe Art der Aufgabe: das Erkennen und Klassifizieren von Entitäten in Texten. Während es in einigen Domänen (zum Beispiel Nachrichten) zahlreiche verfügbare Annotationen gibt, ist es in Fachdomänen wie unter anderem der Biomedizin äußerst kostspielig NER-Labels manuell zu annotieren [68]. Die manuelle Annotation von Daten ist zudem zeitaufwändig und fehleranfällig. Daher ist der Gedanke Modelle, die bereits in einer Domäne beispielsweise der Biomedizin auf NER spezialisiert sind, auch in einer anderen Domäne wie beispielsweise den Rechtswissenschaften einzusetzen naheliegend. Alternativ könnte ein allgemein trainiertes Modell durch Ansätze wie dem kontextuellen Lernen die Aufgabe der NER für diverse Domänen erlernen.

2022 haben Ge et al. in einer Metastudie acht Studien aufgezeigt, die untersuchten, wie man mit nur wenigen Übungsbeispielen - Few-Shot-Learning - eine gute Modellleistung bei NER erreichen kann. Dabei wählten die Studien verschiedene Modelle und Ansätze, um NER mit nur wenigen Übungsbeispielen domänenübergreifend zu verbessern. Sie weisen damit vor allem das vermehrte Interesse am Forschungsthema seit 2020 aus [19].

2023 haben Zhang et al. den Einsatz von domänen-übergreifender Few-Shot-NER untersucht. Sie identifizierten als größte Herausforderungen die domänenübergreifenden Unterschiede der zu erkennenden Entitäten. Dazu zählt die diverse Struktur von Entitäten. Als Beispiel nennen sie das „Lincoln Memorial“, welches als Entität einen Ort repräsentiert, jedoch anderweitig „Lincoln“ enthält und damit zusätzlich als Entität der Person verstanden werden kann. Dabei handelt es sich um eine geschachtelte Entität. Entitäten sind grundsätzlich flach, verschachtelt oder diskontinuierlich [68]. Letzteres bedeutet, dass Start und/oder Ende einer Entität nicht eindeutig identifizierbar sind, beziehungsweise sich die Entitäten überschneiden [61] (zum Beispiel „Have much muscle (*E1*, *E2*) pain (*E1*) and fatigue (*E2*)“) [65].

Zum anderen weisen sie auf die Herausforderungen sich unterscheidender Annotationen von Entitäten in diversen Domänen hin, da es je nach Domäne vordefinierte Entitätstypen gibt [68]. Textbasierte Daten können Mehrdeutigkeiten und Konnotationen enthalten, die eine Verallgemeinerung erschweren. Bereichsspezifische Terminologien, Ausdrücke und Assoziationen schaffen weitere Herausforderungen für FSL [22]. Beispielsweise könnte Java in der Welt der Software-Entwicklung als

Programmiersprache annotiert werden, in der Geografie jedoch als Insel.

Ein FSL-Ansatz bedarf innovativer Mechanismen zur Einbindung von Vorwissen, das dem lernenden Modell vor dem Lernen zur Verfügung gestellt wird, wodurch eine kleine Anzahl von Übungsbeispielen ausreicht, um das Modell auf eine bestimmte Domäne fein abzustimmen. Untersuchte Methoden zur Einbeziehung von Vorwissen sind unter anderem prototypische Netzwerke, Meta-Lernen und Prompt-basierte Lernmethoden wie beispielsweise das kontextuelle Lernen [19].

Dass die Leistung von LLMs in Bezug auf NER gemischt ausfällt, liegt laut Wang et al., 2023, an der Kluft, die zwischen der Aufgabe und dem eigentlichen Fokus des Modells besteht. NER ist von Natur aus eine Sequenzetikettierungsaufgabe, bei der das Modell den Text analysieren und jedem Token innerhalb eines Satzes ein Entity-Typ-Label zuweisen muss. Es handelt sich um eine Klassifizierungsaufgabe auf Tokenebene, während reine Decoder als Textgenerierungsmodelle darauf trainiert sind, das nächste Token in einem Text vorherzusagen.

Sie schlagen daher GPT-NER vor, wodurch NER-Aufgaben in Textgenerierungsaufgaben umgewandelt werden, die von decoderbasierten Modellen leichter gelöst werden können [59]. Zudem soll mit der vorgestellten Methode das Problem haluzinierender Modelle gelöst werden [59], in dem das Modell Informationen und Entitäten erzeugt, die plausibel klingen, aber faktisch falsch, unsinnig oder frei erfunden sind, da das Modell in ihrem Ansatz selbst keine Entitäten mehr „aufschreiben“ muss.

Ihr Prompt zur Extraktion von Standorten besteht aus einer Aufgabenbeschreibung, die dem Modell aufträgt linguistisches Wissen einzubeziehen („*I am an excellent linguist*“), und dem Auftrag inklusive zu extrahierender Entitäten („*The task is to label [Entity Type] entities in the given sentence*“), gefolgt von dem Hinweis auf angehängte Beispiele („*Below are some examples*“). Anschließend werden Beispiele als Input-Sequenz („*China says Taiwan spoils atmosphere for talks.*“) und Output-Sequenz beispielsweise der Input als gelabelter Satz („*@@China## says @@Taiwan## spoils atmosphere for talks.*“) bereitgestellt. Um die Lücke zwischen der Sequenzbeschriftungsaufgabe und der Texterzeugungsaufgabe zu schließen, weisen sie das Modell an, eine beschriftete Sequenz zu erzeugen, indem das Modell ermittelte Entitäten mit speziellen Token (@@, ##) umgibt. Die Schwierigkeit der Aufgabe wird für das Modell reduziert, indem das Modell nur Positionen markieren muss und den Rest des Textes schlichtweg kopiert. Das heißt, die Aufgabe der Extraktion und Transformation in ein anderes Format fällt weg.

Beispiele zur Verwendung im Prompt können per Random Retrieval aus einer Reihe verfügbarer Beispiele zufällig ausgewählt werden, was keine Sicherheit in Bezug auf die Nähe des Beispiels zum Test-Input der Aufgabe gibt, das heißt, dass die mitgegebenen Übungsbeispiele wirklich repräsentativ für die Aufgabe sind. Um die Nähe zu gewährleisten, nutzen Wang et al. kNN-based Retrieval, bei dem zunächst Repräsentationen für alle Trainingsbeispiele berechnet werden, auf deren Grundlage sie anschließend die *k nearest neighbor* für eine eingegebene Testsequenz erhalten. kNNs können basierend auf Satz- oder Entitäts-Ebene bestimmt werden, wobei letz-

teres letztendlich zu einer höheren Leistung führt.

Zudem nutzen sie eine Strategie zur Selbstverifizierung, um dem Halluzinationsproblem vorzubeugen, indem sie das Modell über eine einfache „yes“ oder „no“ Frage verifizieren lassen, ob es sich bei der extrahierten Entität tatsächlich um einen Standort handelt („*The task is to verify whether the word is a location entity extracted from the given sentence*“).

Sowohl bei flachen (CoNLL2003, OntoNotes5.0) als auch geschachtelten (ACE2004, ACE2005 und GENIA) NER-Datensätzen konnten Wang et al. zeigen, dass sie mit nur wenigen Übungsbeispielen in Kombination mit kNN-Verfahren, insbesondere auf Entitäts-Level, zur Ermittlung geeigneter Übungsbeispiele und einer Strategie zur Selbstverifikation, Leistungen nah an denen von Modellen wie BERT-Tagger oder BERT-MRC, die durch überwachtes Training unter anderem speziell auf NER-Aufgaben fein abgestimmt wurden und damit SOTA Baselines vorgeben, erzielen können [59]. Geschachtelt bedeutet in diesem Zusammenhang, dass die Entitäten im Datensatz miteinander in einer Beziehung, beispielsweise einer hierarchischen Beziehung, stehen. Ein Beispiel wäre eine Entität wie UN Security Council, wobei diese als Untergruppe der Entität United Nations betrachtet wird.

Insbesondere diese Studie zeigt wie Übungsbeispiele, die ähnlich zur Aufgabenstellung sind, die Modellleistung verbessern können. Wang et al. untersuchen GPT-NER am Modell GPT-3 und weisen darauf hin, dass insbesondere GPT-3 in seiner Leistung durch das gegebene Token-Limit kein Leistungsplateau erreichen kann.

### **3.3 Reine Decoder Modelle und Few-Shot Prompting für Named Entity Recognition**

Laut Brown et al., 2020, kann zum Beispiel GPT-3 durch kontextuelles Lernen in vielen NLP-Aufgaben mit nur wenigen anschaulichen Beispielen in einem Few-Shot Prompt für Lösungen von Aufgaben eine konkurrenzfähige Leistung im Vergleich zu vollständig überwachten Ansätzen erzielen [9]. Gutiérrez et al. konnten in einer darauf aufbauenden Untersuchung jedoch zeigen, dass zumindest GPT-3 bei Aufgaben im Bereich der Informationsextraktion basierend auf NER und RE durch kontextuelles Lernen schlechter abschneidet, als ein großes feinabgestimmtes LLM wie RoBERTa-large, das mit domänenspezifischen Datensätzen trainiert wurde. Jedoch weisen sie darauf hin, dass insbesondere die Auswahl des Prompts maßgeblich für die Qualität des Ergebnisses sei [21]. Shao et al. haben 2023 eine selbst entwickelte Few-Shot Prompt-Strategie beim Einsatz verschiedener Decoder-Modelle für NER in Bezug auf astrophysikalische Fachartikel untersucht. Hier zeigte vor allem GPT-4 eine gute Performance hinsichtlich Precision, Recall und den F1-Score [49]. Höhere Versionen eines GPT ohne Feinabstimmung scheinen beim Einsatz von Few-Shot Prompting bessere Ergebnisse zu liefern, weswegen sich in dieser Arbeit für gpt-4o-mini von OpenAI als Testmodell entschieden wurde.

### 3.4 Aufbau einer Bekanntmachung

Öffentliche Ausschreibungen auf Online-Portalen liegen dort meistens im HTML (Hypertext Markup Language)-Format vor, ein standardisiertes System zur Kennzeichnung von Textdateien, um Schrift-, Farb-, Grafik- und Hyperlink-Effekte auf World Wide Web-Seiten zu erzielen, da der Fokus bei der Formatierung des Web auf leichter Lesbarkeit statt leichter Extraktion von Informationen liegt [7]. Paragraphen in Hypertext sind oft kürzer und verwenden häufiger Listen- und Tabellen-Formate.

HTML-Bestandteile enthalten Layoutinformationen, die für NER relevant sein können, da beispielsweise über das Wissen über Zusammenhänge von diversen Zellen in einer Tabelle weitere Informationen zu einzelnen Zellen (Header-Zeile, Content-Zeile) in Erfahrung gebracht werden können. Zudem kann ein Beibehalten von Layoutinformationen sinnvoll sein, um ein Dokument in Interessensbereiche einzuteilen (Titel, Absätze, Tabellen, Listen, Bilder) [16]. In der Regel werden HTML-Tags bei der Erkennung von Entitäten in freien Texten auf Webseiten jedoch entfernt [7].

2020 untersuchten C. Ashby und D. Weir die Auswirkungen von HTML-Tags auf die Performanz bei der Erkennung von Entitäten auf Webseiten, indem sie Text-Tag-Kombinationen mit der reinen Text-Repräsentation verglichen und ermittelten eine Leistungserhöhung des harmonischen Mittelwerts (F1-Score) in Bezug auf insgesamt erkannte und korrekt erkannte Entitäten zwischen 0,9 und 13,2 Prozent [7]. Textelemente in Webseiten liegen entweder als strukturierte (Listen oder Tabellen) oder unstrukturierte freie Texte (`<p>`) vor, wobei letztere häufiger vorkommen.

Um HTML-Tags mit in den NER-Prozess einzubeziehen, kann es sinnvoll sein die in HTML zusätzlichen Attribute, Scripts, Kommentar-Blöcke und Styling-Informationen zu entfernen und die bereinigten Tags als einzelne Token zu betrachten oder visuelle Informationen sogar mit einzubeziehen [7]. So konnten E. Apostolova und N. Tomuro zeigen, dass visuelle Informationen wie Position oder Textgröße bei der Entitätserkennung in Immobilien-Marketing-Flyern die Performanz bei NER verbessern kann [4]. Während Ashby und Weir, 2020, mit bereinigten Tags als Tokens zeigen konnten, dass HTML-Merkmale helfen können Mehrdeutigkeiten aufzulösen und die Genauigkeit der Entitätsextraktion zu verbessern. Die Ergebnisse der Studie legen nahe, dass webbasierte NER-Aufgaben erheblich von der Nutzung von HTML-Tags in Kombination mit traditionellen NER-Techniken profitieren können.

In Bezug auf NER durch Web Crawling - ein Prozess, bei dem automatisierte Programme systematisch Inhalte von Webseiten durchsuchen, um deren Inhalte zu sammeln und zu extrahieren - müssen Banner, Werbung, inkonsistente HTML-Tags sowie weitere Elemente berücksichtigt werden. Zudem können HTML-Meta-Tags, wie der Meta-Titel-Tag oder der Meta-Beschreibungs-Tag, nützliche Zusatzinformationen liefern [43]. Speichert man eine Webseite als HTML-Datei, erhält aber nur ein jQuery-Skript oder einen minimalen Inhalt, ist die Webseite wahrscheinlich stark auf JavaScript angewiesen, um den Inhalt dynamisch zu laden. Dies ist häufig bei modernen Webanwendungen der Fall, die mit Frameworks wie React, Angular oder

Vue.js erstellt wurden. Hier ist der anfängliche HTML-Code möglicherweise minimal und der eigentliche Inhalt wird von JavaScript geladen oder gerendert, nachdem die Seite geladen wurde. Bei dem Vergabeportal der Deutsche Bahn AG ist das der Fall und beim Abruf einer Ausschreibung ist der Inhalt nicht geladen. Im Gegensatz dazu erhält man bei der e-Vergabe, das Vergabeportal des Bundes, eine gut gegliederte HTML, inklusive Styles und Skripts, die alle notwendigen Informationen in Bezug auf die Ausschreibung bereit hält.

Es gibt verschiedene Lösungen, um durch crawlen der Webseiten an relevante Inhalte zu gelangen. Der Crawler kann einen Headless-Browser verwenden, um die Seite zu laden, auf die Ausführung von JavaScript zu warten und dann das vollständig gerenderte HTML oder bestimmte Inhalte zu extrahieren. Headless-Browser sind zum Beispiel Puppeteer (für Chrome/Chromium) und Playwright (für unter anderem Chrome, Firefox und WebKit). Einige Crawler oder Scraping-Tools haben zudem die Fähigkeit JavaScript-Code auf einer Seite auszuführen, um Inhalte dynamisch zu laden, so wie es der Browser eines Benutzers tun würde. Viele dynamische Webseiten laden Inhalte über AJAX oder andere XHR-Aufrufe. Durch Network Interception können Crawler diese Netzwerkanfragen abfangen, um direkt auf die vom Server zurückgegebenen JSON- oder andere Datenformate zuzugreifen und so die Notwendigkeit zu umgehen, die gesamte Seite zu rendern. Dadurch können jedoch relevante Informationen aus dem originalen HTML verloren gehen. Mit Tools wie Puppeteer oder Playwright kann ein Crawler den Netzwerkverkehr überwachen und Daten aus den Antworten extrahieren.

Einige Webseiten bieten APIs an, die den Inhalt direkt zurückgeben, sodass Daten durch den direkten Zugriff auf die API beschafft werden können. Wenn eine Webseite über eine API verfügt, kann es effizienter sein, diese zu nutzen, anstatt den HTML-Code zu crawlen. APIs stellen häufig strukturierte Daten, beispielsweise im JSON-Format, zur Verfügung, die sich leichter analysieren und verwenden lassen. Jedoch gehen auch hier gegebenenfalls statische Informationen aus dem HTML verloren.

Bei Webseiten, die Client-Side Rendering (CSR) verwenden und bei denen der meiste Inhalt erst nach der ersten HTML-Anfrage geladen wird, ist Headless Browsing oder die Ausführung des erforderlichen JavaScript eine Option, um vollständige Inhalte auszulesen.

Um die extrahierte HTML-Struktur näher zu untersuchen wurden zwei exemplarische Ausschreibungen, von den Vergabeportalen e-Vergabe und Deutsche Bahn AG, näher untersucht. Bei der Ausschreibung der e-Vergabe „Lieferung eines Ionenchromatographen gemäß Leistungsbeschreibung inkl. Installation, Inbetriebnahme und Einweisung/Schulung“, zu sehen in den Screenshots aus Abbildung 1 und Abbildung 2, handelt es sich um eine einfache Ausschreibung ohne Lose.

Der extrahierte HTML-Quellcode 1 von dem e-Vergabe Online-Portal umfasst 24.420 Zeichen, wovon der tatsächliche Inhalt in Bezug auf die Ausschreibung circa 5.894 Zeichen beträgt. Der `div`-Container, in welchem sich der Inhalt befindet, ist mit der ID „content“ markiert. Durch die Überschriften `h1` bis `h4` ist eine ge-

## Ausschreibungsdetails

Lieferung eines Ionenchromatographen gemäß Leistungsbeschreibung inkl. Installation, Inbetriebnahme & Einweisung/Schulung ⓘ

Veröffentlichungsdatum:	21.08.2024
Teilnahme aktivieren bis:	16.09.2024 09:00
Abgabefrist Angebot:	16.09.2024 09:00
Geschäftszeichen:	102-10125345
Vergabestelle:	Bundesanstalt für Geowissenschaften und Rohstoffe
Letzte Änderung:	21.08.2024 15:31



[AUSSCHREIBUNGSUNTERLAGEN EINSEHEN](#)

Abbildung 1: Screenshot mit Ausschreibungsdetails einer Ausschreibung aus dem e-Vergabe Portal

schachtelte Informationsstruktur abgebildet. Diverse Sektionen sind jeweils durch `div`-Container mit den Klassen `section` gekennzeichnet und enthalten mehrere Kapitel (`class=„chapter“`). Innerhalb eines Kapitels befindet sich jeweils eine Überschrift (`<h4>`) mit einer Kapitelnummerierung (`class=„chapter-number“`) und einem Titel (`class=„chapter-headline“`) und dem Inhalt (`class=„chapter-content“`). Abbildung 3 stellt den Aufbau der HTML als Ausschnitt dar. Die Metadaten aus der HTML geben keine nutzbaren Zusatzinformationen über den Inhalt der Ausschreibung.

Der extrahierte HTML-Quellcode 2 von dem Vergabeportal der Deutschen Bahn AG zeigt, dass der Aufbau der HTML, im Vergleich zum HTML-Quellcode von dem Vergabeportal der e-Vergabe 1, unterschiedlich ist. Ein wesentlicher Unterschied ist, dass das HTML vom Vergabeportal der Deutschen Bahn AG mit Angular Material Design Elementen und `div`-Containern strukturiert ist. Auch hier geben die Metadaten aus der HTML keine nutzbaren Zusatzinformationen über den Inhalt der Ausschreibung.

### 3.5 Named Entity Recognition im Bereich öffentlicher Ausschreibungen

Aus den vorangegangenen Kapiteln geht hervor, dass sich sowohl Ansätze aus dem Bereich des kontextuellen Lernens mit mehreren Übungsbeispielen (Few-Shot) als auch reine Decoder-Modelle zur NER eignen können. Dieser Ansatz bedarf einer geringen Menge gelabelter Trainingsdaten, einer geringen Trainingszeit und verringerter Komplexität des Modells durch Eliminierung des Encoders. Dennoch sind im Bereich öffentlicher Ausschreibungen neben den bekannten Herausforderungen bei

## Auftragsbekanntmachung

Öffentliche Ausschreibung nach UVgO

Liefer- / Dienstleistungsauftrag

1. Öffentlicher Auftraggeber (Vergabestelle)	
a) Hauptauftraggeber (zur Angebotsabgabe auffordernde Stelle)	
Name:	Bundesanstalt für Geowissenschaften und Rohstoffe (BGR)
Straße, Hausnummer:	Stilleweg 2
Postleitzahl (PLZ):	30655
Ort:	Hannover
E-Mail:	luca.magiera@bgr.de
b) Zuschlag erteilende Stelle	
Wie Hauptauftraggeber siehe a)	

Abbildung 2: Screenshot mit Informationen zur Vergabestelle einer Ausschreibung aus dem e-Vergabe Portal

NER - unter anderem geschachtelte, mehrdeutige und diskontinuierliche Entitäten - folgende zu nennen:

### **Fehlende NLP-Benchmarks und -Datensätze**

Es gibt mehrere NLP-Benchmarks und -Datensätze, die NER als eine Kernaufgabe beinhalten, um wissenschaftliche Standards zu setzen, Modelle in Bezug auf NER zu validieren, trainieren und zu testen. Darunter, unter anderem das Automatic Content Extraction (ACE) Programm der University of Pennsylvania [57], Datensätze von Hugging Face [15] oder branchenspezifische Benchmarks wie FiNER-Open Research Dataset (FiNER-ORD) [48] für NER in Finanzdokumenten. Bekannt in Bezug auf NER ist zudem CoNLL, welche sich auf eine Reihe gemeinsamer Aufgaben und zugehöriger Datensätze, die im Rahmen der Conference on Natural Language Learning, einer bekannten Konferenz in der NLP-Gemeinschaft, organisiert wurden [44]. 2024 stellten Mayhew et al. Universal NER (UNER) vor, ein offenes, gemeinschaftsgetriebenes Projekt zur Entwicklung von NER-Benchmarks nach dem Gold-Standard in vielen Sprachen. Ziel von UNER ist es, sprachübergreifend konsistente Annotationen qualitativ hochwertig bereitzustellen, um die mehrsprachige NER-Forschung zu erleichtern und zu standardisieren [29].

In Bezug auf NER im Bereich der Auftragsvergabe, des Ausschreibungs- und Beschaffungswesens wurden keine spezialisierten Benchmarks oder Datensätze in jeglicher Sprache ermitteln können. Ebenso konnte keine wissenschaftliche Literatur in direktem Bezug auf NER in dieser Domäne insbesondere für deutschsprachige Texte gefunden werden. Die Erstellung von Datensätzen und Benchmarks für spezifische Anwendungen wie NER in öffentlichen Ausschreibungen kann aufwändig und teuer sein [52], wäre jedoch ein wichtiger Schritt um ein Modell gezielt auf die Aufgabe und Domäne zu trainieren und zu testen.

```

Sektion
<div class="section">
  <div class="chapter">
    <h4>
      <div class="chapter-number">1.</div>
      <div class="chapter-headline">öffentlicher Auftraggeber (Vergabestelle)</div>
    </h4>
    <div class="chapter-content">
      <div class="chapter">
        <h4>
          <div class="chapter-number">a)</div>
          <div class="chapter-headline">Hauptauftraggeber (zur Angebotsabgabe auffordernde Stelle)</div>
        </h4>
        <div class="chapter-content">
          <div><span class="content-left">Name:</span> <span class="content-right">Bundesanstalt für
            Geowissenschaften und Rohstoffe (BGR)</span>
          </div>
          <div><span class="content-left">Straße, Hausnummer:</span> <span
            class="content-right">Silleweg 2</span></div>
          <div><span class="content-left">Postleitzahl (PLZ):</span> <span class="content-right">30655</span>
          </div>
          <div><span class="content-left">Ort:</span> <span class="content-right">Hannover</span></div>
          <div><span class="content-left">E-Mail:</span> <span class="content-right"><a
            href="mailto:luca.magiera@bgr.de">luca.magiera@bgr.de</a> </span></div>
        </div>
      </div>
    </div>
  </div>
  <div class="chapter">
    <h4>
      <div class="chapter-number">b)</div>
      <div class="chapter-headline">Zuschlag erteilende Stelle</div>
    </h4>
    <div class="chapter-content"><span>wie Hauptauftraggeber siehe&nbsp;a)</span></div>
  </div>
</div>

```

Abbildung 3: Ausschnitt extrahierte HTML einer Ausschreibung aus dem e-Vergabe Online-Portal

### Vergleichsweise kleine Datenmenge

Die Menge der zur Verfügung stehenden öffentlichen Ausschreibungen für die Feinabstimmung oder das Training eines Modells ist im Vergleich zu anderen Domänen wie beispielsweise der Medizin gering. Auf dem Deutschen Vergabeportal, das einen Großteil der öffentlichen Ausschreibungen auflistet, befinden sich aktuell 14.660 Ausschreibungen (Stand 26.04.2025). Inwiefern es Zugriffsmöglichkeiten und Datenbanken mit vergangenen Ausschreibungen gibt, ist uns nicht bekannt. Lu et al., 2021, weisen als Gegenbeispiel in ihrer Studie auf die zentrale biomedizinische Literaturdatenbank PubMed [27] hin, die über 35 Millionen Zitate aus der biomedizinischen Literatur umfasst (Stand: 2023) [54].

## **Heterogenes Format öffentlicher Ausschreibungen**

Öffentliche Ausschreibungen liegen häufig im HTML-Format mit Layoutinformationen in Cascading Style Sheets vor. Dies kann je nach Vergabeportal sehr unterschiedlich aussehen.

## **Diversität der Entitäten**

Sowohl die Benennung beziehungsweise die Anzahl von Synonymen als auch die Darstellung von Entitäten ist divers. Für die Entität „Geschäftszeichen“ gibt es beispielsweise mindestens fünf Synonyme (Vergabenummer, Identifikationsnummer, Vergabentr., Projektnummer, Interne Kennung). Die Vergabestelle kann außerdem nur der Name einer Organisation sein bis hin zu einer vollständigen Adressangabe inklusive Webauftritt und Kontaktinformationen.

Im Weiteren wird der Einsatz eines nicht feinabgestimmten GPT-4o-Modells von OpenAI im Einsatz zur NER in öffentlichen Ausschreibungen und welche Faktoren bei der Prompt-Gestaltung die Effizienz steigern können untersucht. Es wird angenommen, dass ein rein decoderbasiertes Sprachmodell wie GPT durch gezieltes Prompting in der Lage ist, eine gute Leistung bei der NER in öffentlichen Ausschreibungen zu erzielen. Darüber hinaus wird erwartet, dass ein in den Prompt integriertes Übungsbeispiel (ICL) die Leistung weiter verbessert, wobei eine höhere Ähnlichkeit zwischen Übungsbeispiel und Aufgabenstellung zu einer zusätzlichen Leistungssteigerung führt.

## 4 Methode

### 4.1 Vorgehen

Um herauszufinden, wie sich die Leistung des Modells gpt-4o-mini von OpenAI für NER in öffentlichen Ausschreibungen durch gezielte Anpassung und Feinabstimmung der Prompts verbessert, wurden die Auswirkungen der Mitgabe eines Übungsbeispiels im Prompt untersucht. In der Tabelle 10 im Anhang werden die relevanten Entitäten in öffentlichen Ausschreibungen sowie deren Synonyme und Bedeutung aufgelistet. Die Leistung des Modells bei der Erkennung und Klassifikation von Entitäten wurde anhand von vier Entitäten untersucht. Eine Entität ist das Veröffentlichungsdatum, das in diversen Datumsformaten vorliegen kann. Des Weiteren wurde der Erfüllungsort betrachtet. Hierbei kann es sich um eine vollständige Adresse mit Straße, Hausnummer, Postleitzahl und Stadt, als auch um eine einfache Ortsangabe in Form einer Stadt handeln. Zudem wurde das Geschäftszeichen sowie die Vergabestelle betrachtet, die häufig als verschachteltes Objekt (beispielsweise „Bundesministerium für Verkehr und digitale Infrastruktur“) oder Adressangabe inklusive Kontaktinformationen wie, Mailadresse und Webseite, vorliegt. Eine Entität gilt nur dann als korrekt erkannt, wenn sie vollständig, das heißt, zum Beispiel inklusive Angabe der Währung und als vollständiges Objekt vom Modell identifiziert wurde. Der Aufbau, andere Bestandteile und die Prompt-Strategie bleiben konstant. Es wird zwischen Zero- und One-Shot unterschieden. Zum einen wurde untersucht, ob bereits die Beigabe eines einzigen Übungsbeispiels im Prompt zu einem verbesserten Ergebnis führt, zum anderen, inwiefern die semantische und syntaktische Ähnlichkeit des Übungsbeispiels (Ausschreibung aus dem gleichen Vergabeportal, identische Bezeichnung der Entitäten und nach dem selben Aufbau der HTML) einen Einfluss auf das Ergebnis hat.

### 4.2 Technischer Aufbau

Für die Durchführung des Experiments wurde ein Python-Programm entwickelt, das auf Basis von 30 Ausschreibungen als Testdaten den entworfenen Prompt, dargestellt in Abbildung 4, für die beiden Prompt-Strategien Zero-Shot und One-Shot ausführt. Das entwickelte Python-Programm (inklusive Testdaten und Ergebnisse) wurde in einem Github-Repository<sup>1</sup> gespeichert und öffentlich zur Verfügung gestellt.

In der Methodik des Experiments wurde zunächst die Entwicklungsumgebung IntelliJ IDEA in der Version 2024.3.2.2 auf dem lokalen Rechner installiert. Hierzu wurde die kostenfrei erhältliche Community Edition verwendet. IntelliJ IDEA wurde als primäre Entwicklungsumgebung ausgewählt, da es umfassende Unterstützung für Python bietet und zahlreiche Plugins und Tools zur Verfügung stellt, die

---

<sup>1</sup>[https://github.com/anschm/bachelor\\_thesis\\_gpt\\_ner\\_tenders\\_prompt\\_finetuning](https://github.com/anschm/bachelor_thesis_gpt_ner_tenders_prompt_finetuning)

die Entwicklung erleichtern. Anschließend wurde das Python SDK in der Version 3.12 lokal installiert, um die notwendige Entwicklungsinfrastruktur bereitzustellen.

Für die Durchführung des Experiments wurde ein neues Python-Projekt innerhalb von IntelliJ IDEA erstellt. Zur effizienten Verwaltung der Python-Abhängigkeiten wurde eine virtuelle Python-Umgebung, basierend auf dem lokal installierten Python-SDK in der Version 3.12, im Projekt eingerichtet. Diese Einrichtung ermöglicht es, eine isolierte Umgebung zu schaffen, in der alle benötigten Python-Bibliotheken installiert werden können, ohne andere Projekte zu beeinflussen. Zur Verwaltung der Abhängigkeiten wurde eine `requirements.txt`-Datei auf oberster Ebene im Projekt angelegt, in der alle notwendigen Bibliotheken spezifiziert wurden. Die Bibliotheken `pandas`, in der Version 2.2.3, `openpyxl`, in der Version 3.1.5, und `openai`, in der Version 1.61.1, wurden in die `requirements.txt`-Datei hinzugefügt. Die Bibliotheken `pandas` und `openpyxl` werden benötigt um Excel-Dateien lesen und schreiben zu können. Dies ist für die Verarbeitung der Testdaten notwendig. Die Bibliothek `openai` wird benötigt um clientseitig die OpenAI Schnittstelle anzusprechen. Die Installation der Abhängigkeiten innerhalb der virtuellen Umgebung wurde mittels des Befehls `pip install -r requirements.txt` vorgenommen. Dies stellt eine unkomplizierte und reproduzierbare Installation der für dieses Experiment notwendigen Bibliotheken sicher.

Für die Implementierung des Experiments wurde der Ordner `src` erstellt. Anschließend wurde die Datei `environment_variables.py` erzeugt. Diese dient der Bereitstellung von Umgebungsvariablen mittels statischen Funktionen, die für die Konfiguration des Experiments benötigt werden. Hierzu können in der `.env` Datei auf oberster Ebene die Umgebungsvariablen `RESULTS_DIR`, `TEST_DATA_DIR`, `OPENAI_ROLE`, `OPENAI_MODEL` und `OPENAI_API_KEY` konfiguriert werden. Über die Umgebungsvariable `TEST_DATA_DIR` kann gesteuert werden, in welchem Verzeichnis die Testdaten für die Durchführung des Experimentes zur Verfügung gestellt werden. Entsprechend kann über die Umgebungsvariable `RESULTS_DIR` das Verzeichnis angegeben werden, in dem die Ergebnisse des Experimentes gespeichert werden sollen. Die Umgebungsvariable `OPENAI_ROLE` definiert die Rolle, zum Beispiel „User“, mit der ein Prompt an die OpenAI Schnittstelle gesendet wird. Für die Durchführung des Experiments wurde als Rolle „User“ gewählt. Zudem kann über die Umgebungsvariable `OPENAI_MODEL` das OpenAI Modell für die Durchführung des Experiments gewählt werden. Im Rahmen dieser Bachelorarbeit wurde das OpenAI Modell `gpt-4o-mini` gewählt und als Umgebungsvariable gesetzt. Über die Umgebungsvariable `OPENAI_API_KEY` kann der OpenAI API Key für die Durchführung des Experiments hinterlegt werden. Es wurde ein Konto bei OpenAI erstellt, über das ein API-Key generiert und als Umgebungsvariable hinterlegt wurde.

Als nächstes wurde die Python-Datei `open_ai_client.py` erstellt, in der die Python-Klasse `OpenAiClient` implementiert wurde. Die Python-Klasse `OpenAiClient` stellt Funktionen bereit, um einen Prompt an die OpenAI Schnittstelle zu senden und ermöglicht die Integration von GPT-Modellen von OpenAI

innerhalb des Projekts. Zudem erhebt die Klasse für jeden Prompt verschiedene Metriken (benötigte Zeit in Sekunden, verbrauchte Tokens, Länge des HTMLs der Ausschreibung). Im Weiteren wurde die Python-Datei `prompt_response.py` erstellt und die Python-Klasse `PromptResponse` implementiert. Diese Klasse repräsentiert das Ergebnis eines Prompts inklusive der erhobenen Metriken.

Für das Erstellen des Prompts wurde die Python-Datei `prompt_builder.py` erstellt und die Python-Klasse `PromptBuilder` implementiert. Diese stellt Funktionen zur Verfügung, um für die beiden Prompt-Strategien Zero-Shot und One-Shot den entsprechenden Prompt zu erstellen (der Prompt-Aufbau wird in Kapitel 4.4) beschrieben.

Um in der Implementierung zwischen den verschiedenen Prompt-Strategien unterscheiden zu können, wurde in Python ein Enum mit dem Namen `PromptStrategy` implementiert und in der Python-Datei `prompt_strategy.py` gespeichert.

Für die Ausführung des Experiments wurde die Python-Datei `main.py` erstellt. Diese stellt den Haupteinstiegspunkt für das Programm dar und orchestriert den Ablauf des Experiments. Dabei werden die verschiedenen Komponenten des Programms kombiniert. Entsprechend der gewählten Prompt-Strategie werden die Testdaten geladen, der Prompt erstellt, an OpenAI gesendet, Metriken erhoben und das Ergebnis in einer Excel-Datei gespeichert.

### **4.3 Aufbereitung der Testdaten**

Für den Aufbau der Testdaten wurden 30 Ausschreibungen von verschiedenen Vergabeportalen gewählt. Die Anzahl der Ausschreibungen pro Vergabeportal ist in der Tabelle 3 dargestellt. Bei der Wahl der Ausschreibungen wurde darauf geachtet, dass nahezu alle Entitäten enthalten sind. Eine Auflistung der Ausschreibungen (inklusive Name und Länge des HTMLs) kann in der Tabelle 11 gefunden werden.

Vergabeportal	Anzahl
e-Vergabe Online	8
Vergabe Bergisch-Rheinischer Wasserverband	2
Deutsche Bahn AG Vergabeportal	3
Vergabe Stadt Köln	2
Vergabe Fraunhofer	3
Vergabe Deutsche Rentenversicherung Bund	2
Vergabe Sachsen	2
Vergabe Mecklenburg-Vorpommern	3
Vergabe Saarland	2
Vergabe Hessen	3

Tabelle 3: Testdaten von verschiedenen Vergabeportalen

Für die Speicherung der Testdaten wurde der Ordner `tender-test-data` auf oberster Ebene im Projekt erstellt. Mit einem Browser wurde die Webseite für jede Ausschreibung besucht und über die Konsole des Browsers der HTML-Inhalt der Ausschreibung ohne Metadaten, Javascript, Header und Footer kopiert und in einer HTML-Datei im Unterordner `tender-websites` gespeichert. Jeder Ausschreibung wurde der Reihe nach ein Identifikator (Index) von 1 bis 30 zugeordnet. Gemäß des Identifikators der Ausschreibung wurde die HTML-Datei benannt (`<index>.html`). Im Weiteren wurde eine Excel-Datei mit dem Namen `tender-test-data.xlsx` als Steuerdatei erstellt. Diese speichert für jede Ausschreibung den Identifikator (Index), die URL der Webseite (Source), das Datum, an dem das HTML kopiert wurde (Scraped at), und den Titel (Title).

Jede Ausschreibung wurde händisch gelabelt und im Unterordner `tender-websites-labeled` nach dem gleichen Namensschema gespeichert. Dabei wurde für jede Ausschreibung die HTML-Datei kopiert und die vier relevanten Entitäten gelabelt.

Von dem Vergabeportal e-Vergabe Online wurde eine Ausschreibung als Übungsbeispiel für die Prompt-Strategie One-Shot ausgewählt. Hierzu wurde der HTML-Inhalt aus der Konsole im Browser kopiert, manuell gelabelt und im Unterordner `tender-example\ONE_SHOT\example.html` gespeichert.

#### 4.4 Aufbau und Ausführung des Prompts

In Abbildung 4 wird der Aufbau des Prompts visuell nach Abschnitten dargestellt. Der Prompt wurde in Anlehnung an die Prompt-KEE-Strategie von Shao et al., 2023, entworfen [49].

Der Prompt besteht aus einer Beschreibung der Aufgabe. Die Aufgabenbeschreibung enthält eine Erklärung zu NER sowie jeweils eine der Ausschreibungen aus

den gesammelten Testdaten, in der vom Modell die betrachteten Entitäten erkannt werden sollen.

Im Abschnitt „Definition der Entitäten“ werden die Entitäten sowie ihre Synonyme benannt und jede Entität wird kontextuell beschrieben.

Im Abschnitt „Aufgabenschwerpunkte“ wird das Modell auf den Schwerpunkt der Aufgabe, Entitäten korrekt (Precision) und vollständig (Recall) zu erkennen, hingewiesen.

Der Abschnitt „Übungsbeispiele“ ist bei der Prompt-Strategie Zero-Shot nicht vorhanden, da in diesem Fall kein Übungsbeispiel mitgegeben wird. Bei der Prompt-Strategie One-Shot wird ein Übungsbeispiel mitgegeben. In den Ergebnissen der Prompt-Strategie One-Shot wird im Weiteren zwei Szenarien betrachtet. In Szenario 1 stammt die Ausschreibung, in der Entitäten erkannt werden sollen, nicht von dem Vergabeportal, das im Übungsbeispiel repräsentiert ist. Das heißt, das Übungsbeispiel ist entweder sehr oder weniger ähnlich zur Aufgabenstellung. Szenario 2 beschreibt den Umstand, dass die in der Aufgabenbeschreibung mitgegebene Ausschreibung von dem Vergabeportal stammt, das auch im Übungsbeispiel vertreten ist und damit ähnlich ist.

Der Abschnitt „Ausgabeindikator“ erläutert dem Modell in welcher Form es das Ergebnis ausliefern soll. Um dem Modell keine weiteren Aufgaben neben der eigentlichen Erkennung von Entitäten aufzutragen, soll das Modell die gesamte mitgegebene Ausschreibung aus der Aufgabenbeschreibung vollständig zurückgeben und die erkannten Entitäten mit @@ als Präfix und \*\* als Suffix kennzeichnen. Die Entitäten zu sammeln und beispielsweise in einer Tabelle auszugeben, würde weitere Aufgaben und Arbeitsschritte für das Modell bedeuten, was außerhalb dieses Experiments, vorzugsweise in einem weiteren Schritt, erfolgen kann. Die einzelnen Abschnitte des Prompts werden jeweils durch Separatoren (beispielsweise ##Aufgabenbeschreibung##) getrennt.

### **##Aufgabenbeschreibung##**

Du sollst Entitäten in dieser öffentlichen Ausschreibung erkennen:

### **##Definitionen der Entitäten##**

Folgende Entitäten sollst du erkennen:

1. Veröffentlichungsdatum (Synonyme: Datum der Veröffentlichung): kann in verschiedenen Formaten vorliegen (bspw. 23.01.2025, 23. Januar 2025).
2. Erfüllungsort (Synonyme: Ort der Ausführung, Leistungsort, Ort der Leistungserbringung): hierbei kann es sich um eine vollständige Adresse mit Straße, Hausnummer, Postleitzahl und Stadt, als auch um eine einfache Ortsangabe in Form einer Stadt handeln.
3. Vergabestelle: die liegt häufig als verschachteltes Objekt vor (bspw. „Bundesministerium für Verkehr und digitale Infrastruktur“).
4. Geschäftszeichen (Synonyme: Vergabenummer, Identifikationsnummer, Vergabernr.): setzt sich typischerweise aus der Behörde oder Institution, dem Jahr der Veröffentlichung, einer fortlaufenden Ausschreibungsnummer und gegebenenfalls der Art der Ausschreibung zusammen (z. B. 0362/25-B-Ö-21).

### **##Aufgabenschwerpunkte##**

Bevor du die Lösung abgibst, überprüfe bei jeder Entität, ob du sie wirklich richtig und vollständig identifiziert hast. Entitäten sind nur dann richtig erkannt, wenn sie vollständig und korrekt erkannt wurden. Entitäten können diskontinuierlich oder geschachtelte Objekte, wie bspw. „Bundesministerium für Verkehr und digitale Infrastruktur“, sein.

### **##Übungsbeispiel##**

...

### **##Ausgabeindikator##**

Gebe den Auszug aus der Ausschreibung vollständig zurück. Markiere die gefundenen Entitäten wie folgt: die Entität wird mit Präfix @@ und dem Suffix \*\* gekennzeichnet. Schreibe in Klammern die Art der Entität dahinter (Veröffentlichungsdatum, Erfüllungsort, Vergabestelle oder Geschäftszeichen).

Abbildung 4: Aufbau des Prompts

Im Experiment wurde der Prompt mit den 30 öffentlichen Ausschreibungen aus den Testdaten ausgeführt. Der erste Durchlauf erfolgte mit der Zero-Shot (ohne Übungsbeispiel) und der zweite Durchlauf erfolgte mit der One-Shot (mit Übungsbeispiel) Prompt-Strategie.

## 4.5 Erhobene Metriken

In dieser Arbeit werden die Metriken Precision, Recall und F1-Score ermittelt. Diese Metriken sind besonders wichtig, um zu verstehen, wie gut das Modell in der Erkennung und Klassifikation von benannten Entitäten arbeitet [44, 64]. Der F1-Score wird in einer Vielzahl von Studien zur Untersuchung der Modellleistung bei NER herangezogen [19] und bietet somit einen guten Vergleichswert. Die Precision misst den Anteil der korrekt erkannten Entitäten an allen vom Modell klassifizierten Entitäten. Eine hohe Precision bedeutet, dass das Modell nur dann eine Entität als solche markiert, wenn es sich sicher ist, dass es wirklich eine benannte Entität ist.

$$\text{Precision} = \frac{\text{Korrekt Positive}}{\text{Korrekt Positive} + \text{Falsch Positive}}$$

Recall gibt an, wie viele der tatsächlichen benannten Entitäten im Text korrekt vom Modell erkannt wurden. Ein hoher Wert zeigt, dass das Modell in der Lage ist, die meisten relevanten Entitäten zu identifizieren.

$$\text{Recall} = \frac{\text{Korrekt Positive}}{\text{Korrekt Positive} + \text{Falsch Negative}}$$

Der F1-Score stellt das harmonische Mittel von Precision und Recall dar. Ein hoher F1-Score deutet darauf hin, dass das Modell sowohl eine hohe Precision als auch einen hohen Recall aufweist. Das ist bei NER-Aufgaben wichtig, da sowohl die korrekte Klassifikation als auch das Identifizieren aller relevanten Entitäten wichtig sind.

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Die Wahl dieser Metriken ist in der NER besonders sinnvoll, da sie verschiedene Aspekte der Modellleistung abbilden. Die Precision stellt sicher, dass erkannte Entitäten korrekt sind. Der Recall stellt sicher, dass alle relevanten Entitäten vollständig erkannt werden. Der F1-Score kombiniert die beiden Metriken, um eine ausgeglichene Bewertung des Modells zu liefern.

## 5 Ergebnisse

Tabelle 5 zeigt die erhobenen Metriken für Precision, Recall und den berechneten F1-Score für alle betrachteten Prompt-Strategien und Szenarien. Es wurde nicht zwischen der Performance in Bezug auf verschiedene Entitäten unterschieden, sondern die Gesamtleistung bei der Erkennung aller Entitäten betrachtet. Die Ergebnisse wurden manuell bestimmt, indem die vom Modell erkannten Entitäten mit den manuell gelabelten Entitäten aus den Testdaten verglichen wurden. Unvollständig erkannte Entitäten wurden als falsch bzw. nicht erkannt bewertet. Entitäten, die zwar erkannt, aber nicht korrekt klassifiziert wurden (zum Beispiel ein Erfüllungsort, der als Zuschlagsstelle erkannt wurde), wurden ebenfalls als nicht erkannt bewertet. Für Precision, Recall und F1-Score wird jeweils der Mittelwert und die Standardabweichung<sup>2</sup> angegeben. Die Standardabweichung ist ein Maß für die Streuung der Werte, das angibt, wie weit die einzelnen Werte für Precision, Recall und F1-Score pro Ausschreibung im Durchschnitt von dem errechneten Mittelwert entfernt sind.

Prompt-Strategie	Szenario	Precision (MW ± Std.-Abw.)	Recall (MW ± Std.-Abw.)	F1-Score (MW ± Std.-Abw.)
Zero-Shot	-	0,38 ± 0,15	0,50 ± 0,20	0,40 ± 0,15
One-Shot	-	0,39 ± 0,25	0,41 ± 0,25	0,37 ± 0,23
	Szenario 1	0,36 ± 0,24	0,35 ± 0,22	0,33 ± 0,21
	Szenario 2	0,56 ± 0,24	0,70 ± 0,24	0,59 ± 0,24

Tabelle 5: Leistungsmetriken (Precision, Recall, und F1-Score) als Mittelwert (MW) und Standardabweichung (Std.-Abw.) für die Prompt-Strategien Zero-Shot und One-Shot

In Tabelle 7 sind der durchschnittliche Recall, die Precision und der F1-Score über alle Ausschreibungen hinweg pro Entität inklusive der Standardabweichung zu sehen.

<sup>2</sup>Der Mittelwert kann nicht negativ werden und ist daher nach unten durch 0 begrenzt.

Prompt-Strategie	Szenario	Precision (MW $\pm$ Std.-Abw.)	Recall (MW $\pm$ Std.-Abw.)	F1-Score (MW $\pm$ Std.-Abw.)
<b>Veröffentlichungsdatum</b>				
<b>Zero-Shot</b>	-	0,11 $\pm$ 0,22	0,37 $\pm$ 0,49	0,15 $\pm$ 0,25
<b>One-Shot</b>	-	0,21 $\pm$ 0,37	0,37 $\pm$ 0,49	0,23 $\pm$ 0,37
	Szenario 1	0,14 $\pm$ 0,33	0,24 $\pm$ 0,44	0,15 $\pm$ 0,33
	Szenario 2	0,54 $\pm$ 0,43	1,00 $\pm$ 0,00	0,62 $\pm$ 0,36
<b>Vergabestelle</b>				
<b>Zero-Shot</b>	-	0,15 $\pm$ 0,26	0,27 $\pm$ 0,41	0,17 $\pm$ 0,28
<b>One-Shot</b>	-	0,21 $\pm$ 0,34	0,20 $\pm$ 0,34	0,19 $\pm$ 0,31
	Szenario 1	0,13 $\pm$ 0,31	0,12 $\pm$ 0,30	0,11 $\pm$ 0,26
	Szenario 2	0,60 $\pm$ 0,22	0,60 $\pm$ 0,22	0,60 $\pm$ 0,22
<b>Geschäftszeichen</b>				
<b>Zero-Shot</b>	-	0,81 $\pm$ 0,36	0,87 $\pm$ 0,35	0,83 $\pm$ 0,35
<b>One-Shot</b>	-	0,67 $\pm$ 0,48	0,48 $\pm$ 0,40	0,54 $\pm$ 0,41
	Szenario 1	0,64 $\pm$ 0,49	0,41 $\pm$ 0,36	0,49 $\pm$ 0,39
	Szenario 2	0,80 $\pm$ 0,45	0,80 $\pm$ 0,45	0,80 $\pm$ 0,45
<b>Erfüllungsort</b>				
<b>Zero-Shot</b>	-	0,43 $\pm$ 0,46	0,50 $\pm$ 0,49	0,45 $\pm$ 0,46
<b>One-Shot</b>	-	0,50 $\pm$ 0,45	0,60 $\pm$ 0,50	0,53 $\pm$ 0,46
	Szenario 1	0,54 $\pm$ 0,45	0,64 $\pm$ 0,49	0,57 $\pm$ 0,46
	Szenario 2	0,30 $\pm$ 0,45	0,40 $\pm$ 0,55	0,33 $\pm$ 0,47

Tabelle 7: Leistungsmetriken (Precision, Recall, und F1-Score) als Mittelwert und Standardabweichung für die Prompt-Strategien Zero-Shot und One-Shot pro Entität

In Tabelle 9 sind zudem die durchschnittliche Dauer, die durchschnittlich benötigten Tokens und die durchschnittliche Länge des HTMLs der Ausschreibung pro Prompt-Strategie sowie die Standardabweichung zu sehen.

Prompt-Strategie	Zeit [s] (MW $\pm$ Std.-Abw.)	Tokens (MW $\pm$ Std.-Abw.)	Länge HTML (MW $\pm$ Std.-Abw.)
Zero-Shot	97,24 $\pm$ 56,08	11.187,53 $\pm$ 5.302,66	25.667,17 $\pm$ 12.623,25
One-Shot	77,46 $\pm$ 25,85	16.318,73 $\pm$ 3.727,56	25.667,17 $\pm$ 12.623,25

Tabelle 9: Metriken (benötigte Zeit, verbrauchte Tokens, und Länge des HTMLs) für die Prompt-Strategien One-Shot und Zero-Shot

## 5.1 Andere Befunde

Während des Experiments konnten diverse Nebenfunde erhoben werden. Bei der Anwendung der One-Shot Prompt-Strategie gab das Modell in einem Prompt statt der Testausschreibung die Ausschreibung aus dem mitgegebenen Übungsbeispiel zurück. Zudem wurden Entitäten, die aus mehreren Sub-Entitäten bestehen, wie beispielsweise die Vergabestelle aus einer Adresse und einem Internetauftritt, nicht vollständig erkannt. Wobei häufig dort der Inhalt abgeschnitten wurde, wo es auch in der Formatierung durch das HTML einen Bruch gab (`<br>`, `<span>` oder `<p>`). Zudem konnte festgestellt werden, dass Inhaltselemente aus dem HTML im Ergebnis in einem anderen Format als dem ursprünglich mitgegebenen HTML zurückgegeben wurde. Dabei wurden einzelne Tags oder ganze HTML-Abschnitte häufig ohne erkennbares Muster entfernt. Ferner wurde das Veröffentlichungsdatum in einer Ausschreibung beim Labeln der Entität im Ergebnis mit einem Schreibfehler (ohne „g“) zurückgegeben.

## 6 Diskussion

Diese Arbeit untersucht inwiefern der Einsatz eines nicht feinabgestimmten Modells gpt-4o-mini von OpenAI bei NER in öffentlichen Ausschreibungen eingesetzt werden kann und welche Faktoren bei der Gestaltung des Prompts die Effizienz steigern können.

Im Vordergrund der Untersuchung stand der Einfluss eines Übungsbeispiels im Sinne von One-Shot auf das Ergebnis. Wie aus der Literatur hervorgeht können Modelle durch Zugabe eines oder mehrerer Übungsbeispiele im Kontext des Prompts mehr über den angestrebten Lösungsweg oder dem Ergebnis der Aufgabe lernen und so ihr Ergebnis verbessern. Allerdings konnte bereits in einigen Arbeiten gezeigt werden, dass dabei sowohl die Anzahl und Reihenfolge als auch die Ähnlichkeit der Übungsbeispiele zur Aufgabenstellung eine Rolle spielten [70, 28, 59]. Letzteres wird durch die Ergebnisse teilweise bestätigt.

In Szenario 2, in dem das Übungsbeispiel ähnlich zur Ausschreibung ist, werden bessere Ergebnisse sowohl in der Precision, dem Recall als auch dem F1-Score erzielt (F1-Score bei 0,40 bei Zero-Shot zu 0,59 bei One-Shot in Szenario 2). Die Ähnlichkeit bezieht sich auf den Aufbau des HTMLs sowie die Bezeichnung der Entitäten innerhalb der mitgegebenen Ausschreibung im Übungsbeispiel. Die Ausschreibungen stammen zudem vom gleichen Vergabeportal. In One-Shot Szenario 2 kann ein Übungsbeispiel im Prompt die Leistung des Modells insgesamt zwar verbessern, beim Geschäftszeichen und dem Erfüllungsort führte die Mitgabe einer ähnlichen Beispielausschreibung im Übungsbeispiel nicht zu einer Verbesserung, sondern sogar zu einer leichten Verschlechterung bei der Erkennung der beiden Entitäten (Geschäftszeichen: F1-Score 0,83 bei Zero-Shot zu 0,80 bei One-Shot in Szenario 2, Erfüllungsort: F1-Score 0,45 Zero-Shot zu 0,33 bei One-Shot in Szenario 2). Die schlechten Ergebnisse aus Szenario 2 bei diesen Entitäten können weitestgehend darauf zurückgeführt werden, dass das Modell hier häufig gar keine Entität ermitteln konnte, obwohl diese existent waren. Wohingegen das Modell bei One-Shot Szenario 1 und Zero-Shot eher optimistisch vorging und eher zu viel als zu wenig erkannte und klassifizierte. Das führte zwar zu falsch Positiven, jedoch wurde eher „zufällig“ häufig auch die korrekten Entitäten erkannt und klassifiziert.

Grundsätzlich agierte das Modell bei der Mitgabe eines ähnlichen Übungsbeispiels (One-Shot Szenario 2) konservativer. Dabei erkannte und klassifizierte das Modell Entitäten eher gar nicht, statt sie falsch zu erkennen und zu klassifizieren. Hierdurch ist insbesondere die Precision in Bezug auf das Veröffentlichungsdatum und die Vergabestelle augenscheinlich besser, da weniger falsch Positive erkannt wurden.

Vergleicht man Szenario 1 und 2, schneidet das Modell bei Mitgabe eines ähnlichen Übungsbeispiels sowohl beim Veröffentlichungsdatum, der Vergabestelle als auch dem Geschäftszeichen deutlich besser ab als bei One-Shot mit einem nicht ähnlichen Übungsbeispiel (Szenario 1), wobei es beim Erfüllungsort leicht schlechter performt. Die Mitgabe eines Übungsbeispiels insgesamt (Szenario 1 und 2) verbes-

sert das Ergebnis nur minimal gegenüber Zero-Shot, ausgenommen beim Geschäftszeichen (F1-Score bei 0,83 bei Zero-Shot zu F1-Score 0,54 One-Shot gesamt). Aus den gemittelten Ergebnissen des F1-Scores, 0,40 für Zero-Shot, 0,33 für One-Shot in Szenario 1 und 0,59 in One-Shot Szenario 2, über alle Entitäten und Ausschreibungen hinweg, kann entnommen werden, dass die Mitgabe eines Übungsbeispiels im Prompt die Erkennung von Entitäten in öffentlichen Ausschreibungen zwar verbessern kann, allerdings nur, wenn das Übungsbeispiel eine Ähnlichkeit zur zu labelnden Ausschreibung aufweist.

Im Falle von One-Shot Szenario 1 scheint ein Übungsbeispiel das Modell eher zu „verwirren“ und sowohl zu einer schlechteren Precision als auch Recall bei der Vorhersage zu führen. Im Rahmen von One-Shot in Szenario 1 hat ein nicht ähnliches Übungsbeispiel zu teilweise starker Manipulation der Ausgabe geführt. Hier wurde HTML ohne erkennbares Muster entfernt und in einem Fall wurde die Übungsausschreibung statt der mitgegeben Ausschreibung zurückgegeben. Die Annahme ist, dass das Modell sich im Falle von Zero-Shot an selbst erlernten Mustern und Vorgehensweisen orientiert hat und auf ein Default-Verhalten zurückgefallen ist, das konstant gehalten wurde. Dies wird auch durch eine geringere Standardabweichung der Ergebnisse gestützt. Somit macht das Modell zwar Fehler, aber erkennt eine Entität konsequent nach dem gleichen selbst erlernten Muster falsch.

Gibt man dem Modell ein nicht ähnliches Übungsbeispiel, versucht es scheinbar daraus zu lernen und sein Lösungsvorgehen anzupassen, schafft jedoch keine Transferleistungen für die Erkennung von Entitäten, die nicht exakt so im Übungsbeispiel präsentiert werden, da Synonyme oder ein anderer Aufbau genutzt werden.

Ein ähnliches Übungsbeispiel kann unterstützend wirken, jedoch können wir hier eine starke Varianz in den Ergebnissen sehen und das Ergebnis ist von Entität zu Entität sehr unterschiedlich. Das kann vor allem auf die geringe Datenmenge - 30 Ausschreibungen gesamt, davon 5, die dem Übungsbeispiel ähnlich sind - zurück geführt werden.

Mit der Zugabe eines ähnlichen Übungsbeispiels (Szenario 2) wurde beim Veröffentlichungsdatum eine starke Verbesserung erzielt. Im Falle von Zero-Shot hat das Modell zuverlässig alle Daten als Veröffentlichungsdatum erkannt, selbst wenn eine eindeutige Kennzeichnung in der Ausschreibung gegeben war. Hier scheint das Modell auf ein sehr einfaches Muster zurückgefallen zu sein und nur das Format, nicht aber auf den Kontext, zu achten. Das Übungsbeispiel hat das Modell scheinbar sensitiviert nicht alle Daten zu labeln, sondern auf weitere Kriterien beim Erkennen von Entitäten (hier Titel „Veröffentlichungsdatum“) zu achten. Bei der Entität Vergabestelle, die weitestgehend als diskontinuierliches Objekt mit Adresse, Mail-Adresse und Internetauftritt sowie einem geschachtelten Titel vorlag, war die Performanz insbesondere bei Zero-Shot (F1-Score 0,17) und One-Shot Szenario 1 (F1-Score 0,11) sehr schlecht.

Insgesamt waren sowohl der Aufbau des HTMLs, die Bezeichnung und die Existenz diverser Entitäten sowie deren Beschaffenheit in den Testausschreibungen sehr unterschiedlich. Das Modell war nicht in der Lage durch vorhandenes Wissen - bei-

spielsweise über die Domäne öffentlicher Ausschreibungen, bekannte Synonyme oder Wissen über die Formatierung von HTML - selbstständig Entitäten korrekt zu erkennen und zu klassifizieren. Das zeigt sich vor allem in den schlechten Gesamtergebnissen in der Precision, dem Recall und dem F1-Score bei Zero-Shot. Insgesamt ist der erhobene Recall in allen Szenarien besser oder mindestens fast genauso gut wie die Precision. Das Modell erkennt und klassifiziert eher falsch Positive, übersieht dabei aber seltener tatsächlich existierende Entitäten.

Wie von Zhang et al. 2023 erwähnt, sind Entitäten, welche eine Schachtelung oder Diskontinuität aufweisen, für ein Modell scheinbar schwieriger korrekt zu identifizieren [68]. Das Geschäftszeichen hat als einzige Entität unter Zero-Shot (F1-Score 0,83) und One-Shot Szenario 2 einen F1-Score von 0,8 erreicht, was im Vergleich mit beispielsweise dem Leaderboard für NER von Modellen, die speziell für NER mit dem CoNLL 2003 Datensatz auf Englisch trainiert wurden, als Richtlinie für eine gute Modellleistung angesehen werden kann [63]. Das ist auf die einfache Beschaffenheit - das Geschäftszeichen besteht aus einer einzigen lückenlosen Zeichenkette - sowie auf die Eindeutigkeit des Geschäftszeichens zurückzuführen. Beim Veröffentlichungsdatum handelt es sich im Gegensatz um ein Datumsformat, welches auch unter anderem für die Ausschreibungsfristen genutzt wird. Für alle untersuchten Entitäten wurden alle in den genutzten Ausschreibungen vorhandenen Synonyme im Prompt mitgegeben.

Der Aufbau des HTMLs einer Ausschreibung hat Einfluss auf die Klassifikation, da die Vergabestelle häufig nur teilweise erkannt oder sogar doppelt erkannt wurde. Der Name der Vergabestelle und der Ort wurden konstant separat erkannt. Die Bruchstellen, an denen eine Entität nicht als vollständige Einheit erkannt wurde, wurden durch formatierende HTML-Tags (beispielsweise `<br>`) bestimmt. 2020 ermittelten C. Ashby und D. Weir eine Verbesserung der Ergebnisse durch die Mitgabe von HTML-Informationen im Gegensatz zu bereinigten und damit reinen Textrepräsentation des Inhalts von Webseiten [7]. Hierbei ist zu beachten, dass sie zwei Modelle, basierend auf CRF und CNN-basierten Feature-Extractor, nutzen. Das Modell gpt-4o-mini von OpenAI war in dieser Untersuchung nicht in der Lage die vollständigen Zusammenhänge und Layout-Informationen aus dem mitgegebenen HTML zu extrahieren oder entsprechend zur NER zu verwenden. Der entworfene Prompt weist das Modell nicht explizit an, die HTML- und CSS-Fragmente zu beachten.

Bei One-Shot ist die Verarbeitungszeit kürzer als bei Zero-Shot (97,14 Sekunden bei Zero-Shot zu 77,46 Sekunden bei One-Shot). Ebenso wurde die Standardabweichung der Inferenzzeit verkürzt (56,08 Sekunden bei Zero-Shot zu 25,85 Sekunden bei One-Shot), obwohl die Anzahl der Tokens um mehr als 5.000 Tokens anstieg, bedingt durch die Mitgabe eines Übungsbeispiels. Jeder Prompt bestand je nach Länge des HTMLs der Ausschreibung (durchschnittlich 25.667,17 Zeichen) aus einer unterschiedlichen Anzahl von Tokens. Dennoch war die Anzahl der Tokens im Output pro Prompt für eine Ausschreibung in Zero-Shot und One-Shot jeweils gleich, da die mitgegebene Ausschreibung jeweils die gleiche war und damit die gleiche Länge aufwies. Dass die Inferenzzeit sich so stark unterscheidet, kann darauf zu-

rückzuführen sein, dass das Übungsbeispiel das Modell tatsächlich in der Lösung der Aufgabe unterstützt. Das kann insbesondere bei NER über viele Ausschreibungen hinweg einen Zeitvorteil verschaffen.

Zuletzt sei noch darauf hinweisen, dass das Modell die Aufgabenstellung die Entitäten mit @@ und \*\* zu kennzeichnen, die Entität entsprechend zu erkennen und die gesamte Ausschreibung anschließend zurückzugeben, fast überall korrekt ausgeführt hat und somit Instruktion und Prompt deutlich waren. Teilweise wurden jedoch über alle Prompts und Szenarien hinweg ganze HTML-Abschnitte abgeschnitten, darunter auch Abschnitte, welche relevante Entitäten enthielten. Das Modell baute zudem einen Schreibfehler im Veröffentlichungsdatum ein. Für dieses irrationale Verhalten konnte weder ein Muster noch ein Auslöser identifiziert werden.

Alle durch Nadeau und Sekine, 2007, definierten typischen NER- und Classification (NERC)-Fehler sind aufgetreten. Das Modell hat Entitäten an Stellen erkannt, wo sie nicht waren, Entitäten vollständig übersehen und Entitäten zwar erkannt, aber falsch. Ebenso wurden die Grenzen von Entitäten beim Erkennen falsch identifiziert oder sogar zusätzlich der falsche Bezeichner vergeben [31]. Der F1-Score bezieht alle diese Fehler mit ein und entspricht mit einem Gesamtwert von 0,40 bei Zero-Shot und dem höchsten Wert von 0,59 bei One-Shot Szenario 2 nicht einer guten Modellleistung nach SOTA.

## 7 Fazit

Die vorliegende Arbeit zeigt, dass das decoderbasierte Modell gpt-4o-mini von OpenAI grundsätzlich nicht optimal für NER in öffentlichen Ausschreibungen geeignet ist. Dennoch lässt sich durch den Einsatz von kontextuellem Lernen - hier der Integration von einem Übungsbeispiel in den Prompt - eine verbesserte Leistung erzielen. Die Ergebnisse verdeutlichen, dass die Effektivität eines Übungsbeispiels stark von dessen syntaktischer und semantischer Ähnlichkeit zur Zielaufgabe abhängt.

In Szenario 2, bei dem ein ähnliches Übungsbeispiel verwendet wurde, konnten die Ergebnisse gegenüber Zero-Shot signifikant verbessert werden, wobei der F1-Score anstieg. Besonders bei der Entität „Veröffentlichungsdatum“ zeigte sich eine deutliche Verbesserung der Precision, da weniger falsch-positive Ergebnisse auftraten. Im Gegensatz dazu zeigte die Verwendung eines unähnlichen Übungsbeispiels (Szenario 1) eine ausgeprägte Verschlechterung der Modellleistung.

Die Ergebnisse zeigen zudem, dass die geringe Datenmenge sowie die hohe Variabilität im Aufbau öffentlicher Ausschreibungen die Modellleistung einschränken. Unterschiedliche HTML-Templates, die Verschachtelung von Entitäten und fehlende Domänenkenntnisse erschweren eine konsistente Erkennung und Klassifikation von Entitäten. Beispielsweise wurden beim Erkennen und Klassifizieren der Entität „Vergabestelle“, die häufig diskontinuierliche oder geschachtelte Strukturen aufwies, niedrige F1-Scores erzielt. Selbst bei vergleichsweise einfachen Entitäten wie dem „Geschäftszeichen“ war eine zufriedenstellende Leistung nur durch die Mitgabe umfassender Synonyme und eindeutiger Merkmale möglich. Typische NER-Probleme, wie das Übersehen von Entitäten, fehlerhaftes Labeln oder falsche Grenzziehung, traten unter allen untersuchten Bedingungen auf.

Zusammenfassend zeigt die Arbeit, dass das Design des Prompts bei der Nutzung von gpt-4o-mini von OpenAI für die Modelleistung bei NER ausschlaggebend ist. Verbesserungen sind unter spezifischen Bedingungen, wie der Mitgabe ähnlicher Übungsbeispiele, möglich. Die Leistung des Modells für NER in öffentlichen Ausschreibungen kann durch weitere gezielte Anpassungen und Feinabstimmung des Prompts, wie der Mitgabe geeigneter Übungsbeispiele für diverse Vergabeportale und Ausschreibungsformate im Prompt, gegebenenfalls weiter verbessert werden. Inwiefern dies auf Dauer kostengünstiger und schneller als zum Beispiel bei der Nutzung von encoderbasierten Modellen erfolgen kann und damit langfristig einen Mehrwert bietet, bleibt zu hinterfragen.

### 7.1 Limitation und zukünftige Forschung

Die vorliegende Untersuchung weist mehrere Einschränkungen auf, die die Aussagekraft der Ergebnisse beeinflussen. Zum einen war die Stichprobe mit 30 öffentlichen Ausschreibungen vergleichsweise klein und die untersuchten Entitäten beschränkten sich auf vier ausgewählte Entitäten, während Ausschreibungen in der Regel deutlich mehr relevante Entitäten umfassen. Zum anderen lässt sich nicht ausschließen, dass andere Faktoren im designten Prompt die Ergebnisse beeinflusst

haben, wie etwa der fehlende Hinweis auf die Bedeutung der mitgegebenen HTML-Strukturen. Des Weiteren war das eingesetzte Modell, gpt-4o-mini von OpenAI, nicht auf NER in öffentlichen Ausschreibungen feinabgestimmt und es ist nicht bekannt, wie gut das Modell speziell auf die untersuchte Domäne und Aufgabe vortrainiert ist.

Es wurde lediglich ein GPT-Modell vom Anbieter OpenAI und zwei Prompt-Strategien untersucht, was eine begrenzte methodische Vielfalt darstellt. Prompt Engineering ist als Forschungsfeld noch relativ jung und die Wechselwirkung zwischen Prompt und Modelleleistung variiert laut Literatur je nach Modell. Die geringe Menge an Testdaten und die hohe Variabilität der untersuchten Ausschreibungen erschweren eine Generalisierung der Ergebnisse.

Für die zukünftigen Forschungen ergeben sich zahlreiche Ansätze, um die Ergebnisse der vorliegenden Arbeit zu erweitern und zu verbessern. Ein vielversprechender Ansatz wäre die Feinabstimmung des Modells gpt-4o-mini von OpenAI auf die spezifische Domäne öffentlicher Ausschreibungen und das Auslesen von HTML-Templates, um dessen Verständnis für die Struktur und Sprache solcher Dokumente zu verbessern. Besonders die Extraktion und Interpretation von HTML-Templates könnte eine Schlüsselrolle spielen, da diese oft entscheidend für die korrekte Klassifikation von Entitäten sind.

Darüber hinaus wäre eine systematische Erforschung weiterer Prompt-Strategien sinnvoll, einschließlich Kombinationen verschiedener Techniken und Muster, um deren Einfluss auf die Ergebnisse zu evaluieren. In Bezug auf Few-Shot Prompting wäre es interessant zu erforschen, inwiefern eine höhere Anzahl an Beispielen, die diverse Vergabeportale und Formate öffentlicher Ausschreibungen repräsentieren, die Ergebnisse verbessern und somit die Herausforderungen in Bezug auf diskontinuierliche, verschachtelte und mehrdeutige Entitäten eliminieren können. Wang et al. zeigten 2023 eine Methode zur Auswahl ähnlicher Übungsbeispiele bei NER auf [59]. Es sollte untersucht werden, ob gpt-4o-mini mit einer Auswahl diverser Übungsbeispielen von verschiedenen Vergabeportalen und in mehreren Formaten, besser performt. Hier muss zusätzlich ein passender Ansatz gefunden werden, mit dem sich das Modell effizient ein geeignetes Übungsbeispiel auswählen kann.

Im Bereich der NER für öffentliche Ausschreibungen gibt es bisher keine uns bekannten standardisierten Benchmarks oder Datensätze. Allerdings können Anpassungen bestehender NER-Datensätze oder die Erstellung spezifischer Datensätze und Benchmarks angegangen werden. Allgemeine NER-Datensätze wie CoNLL-2003 oder OntoNotes enthalten potentiell Entitäten, die auch in Ausschreibungsdokumenten vorkommen, wie u.a. Organisationen und Orte. Die Herausforderung besteht darin, diese Datensätze so zu erweitern oder anzupassen, dass sie spezifische Entitäten und Informationen enthalten, die in dieser Domäne relevant sind. Eignen würde sich als Rahmen für die Standardisierung von beschaffungsbezogenen Daten TED (Tenders Electronic Daily), die offizielle Datenbank der Europäischen Union für öffentliche Ausschreibungen [1], oder eProcurement Ontology, ein von der Europäischen Kommission entwickelter Standard, der darauf abzielt, die Kon-

sistenz und Interoperabilität von Daten im Bereich der öffentlichen Beschaffung zu gewährleisten [56]. Beide könnten genutzt werden, um annotierte Datensätze und Benchmarks für das Trainieren von verschiedenen Modellen in dieser Domäne zu definieren.

## Literatur

- [1] Tenders electronic daily, 2025. Offizielle Datenbank der Europäischen Union für öffentliche Ausschreibungen.
- [2] Sparro M. Albano, G. L. Flexible strategies for centralized public procurement. *Review of Economics and Institutions*, 1(2), 2010.
- [3] Amazon Web Services. How sequence-to-sequence works, 2023. Accessed: 2025-04-26.
- [4] Emilia Apostolova and Noriko Tomuro. Combining visual and textual features for information extraction from online flyers. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1924–1929, Doha, Qatar, October 2014. Association for Computational Linguistics.
- [5] Sue Arrowsmith. *Public Procurement Regulation: An Introduction*. Hart Publishing, Oxford, 1 edition, 2011. Provides a comprehensive overview of public procurement processes, including principles of transparency, fairness, competition, efficiency, and innovation in public tenders.
- [6] Birgit Aschhoff and Wolfgang Sofka. Innovation on demand—can public procurement drive market success of innovations? *Research Policy*, 38(8):1235–1247, 2009.
- [7] Colin Ashby and David Weir. Leveraging HTML in free text web named entity recognition. In Donia Scott, Nuria Bel, and Chengqing Zong, editors, *Proceedings of the 28th International Conference on Computational Linguistics*, pages 407–413, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics.
- [8] Dhananjay Ashok and Zachary C. Lipton. Promptner: Prompting for named entity recognition, 2023.
- [9] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020.
- [10] DAIR.AI. Prompt engineering guide. <https://www.promptingguide.ai/introduction/elements>, 2024. Accessed: 2024-09-15.

- [11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.
- [12] Piga G. Spagnolo G. Dimitri, N. Handbook of procurement. 2006.
- [13] Charles Edquist and Jon Mikel Zabala-Iturriagoitia. Public procurement for innovation as mission-oriented innovation policy. *Research Policy*, 41(10):1757–1769, 2012.
- [14] Auftragsberatungszentrum Bayern e.V. (ABZ). E-vergabe-plattformen und anbieter in deutschland. <https://www.abz-bayern.de/abz/inhalte/Anhaenge/uebersicht-e-vergabe-anbieter-deutschland.pdf>, 2021. Accessed: 2024-10-08.
- [15] Hugging Face. Datasets - hugging face. <https://huggingface.co/docs/datasets/en/index>, 2024. Accessed: 2024-09-08.
- [16] Dimitra Farmakiotou, Vangelis Karkaletsis, Georgios Samaritakis, Georgios Petasis, and Constantine D. Spyropoulos. Named Entity Recognition from Greek Web Pages. In Ioannis P. Vlahavas and Constantine D. Spyropoulos, editors, *Proceedings of the 2nd Hellenic Conference on Artificial Intelligence (SETN-02), Companion Volume*, pages 91–102, Thessaloniki, Greece, April 11–12 2002. <http://lpis.csd.auth.gr/setn02/>.
- [17] Zihao Fu, Wai Lam, Qian Yu, Anthony Man-Cho So, Shengding Hu, Zhiyuan Liu, and Nigel Collier. Decoder-only or encoder-decoder? interpreting language model as a regularized encoder-decoder, 2023.
- [18] Tianyu Gao, Adam Fisch, and Danqi Chen. Making pre-trained language models better few-shot learners, 2021.
- [19] Yao Ge, Yuting Guo, Yuan-Chi Yang, Mohammed Ali Al-Garadi, and Abeed Sarker. Few-shot learning for medical text: A systematic review, 2022.
- [20] Luke Georghiou, Jakob Edler, Elvira Uyarra, and Jillian Yeow. Policy instruments for public procurement of innovation: Choice, design and assessment. *Technological Forecasting and Social Change*, 86:1–12, 2014.
- [21] Bernal Jiménez Gutiérrez, Nikolas McNeal, Clay Washington, You Chen, Lang Li, Huan Sun, and Yu Su. Thinking about gpt-3 in-context learning for biomedical ie? think again, 2022.
- [22] Maximilian Hofer, Andrey Kormilitzin, Paul Goldberg, and Alejo Nevado-Holgado. Few-shot learning for named entity recognition in medical text, 2018.
- [23] Imed Keraghel, Stanislas Morbieu, and Mohamed Nadif. A survey on recent advances in named entity recognition, 2024.

- [24] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *arXiv preprint arXiv:2205.11916*, 2022.
- [25] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Ivan Koychev, Vishrav Chaudhary, Wen-tau Wang, Stan Tau, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, 2020.
- [26] Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. What makes good in-context examples for GPT-3? In Eneko Agirre, Marianna Apidianaki, and Ivan Vulić, editors, *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114, Dublin, Ireland and Online, May 2022. Association for Computational Linguistics.
- [27] Yanqing Lu, Yifan Wang, Lianhua Liu, Hui Wang, and Min Li. Biomedical named entity recognition: A survey of machine-learning-based methods. *Briefings in Bioinformatics*, 22(2):bbaa206, 2021.
- [28] Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098, Dublin, Ireland, May 2022. Association for Computational Linguistics.
- [29] Stephen Mayhew, Terra Blevins, Shuheng Liu, Marek Šuppa, Hila Gonen, Joseph Marvin Imperial, Börje F. Karlsson, Peiqin Lin, Nikola Ljubešić, LJ Miranda, Barbara Plank, Arij Riabi, and Yuval Pinter. Universal ner: A gold-standard multilingual named entity recognition benchmark, 2024.
- [30] Gordon Murray and Andrew Dainty. Public procurement as a driver of innovation in construction: Building the future by skimming the past? *International Journal of Public Sector Management*, 22(6):462–479, 2009.
- [31] David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Linguisticae Investigationes*, 30(1):3–26, 2007.
- [32] Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. A comprehensive overview of large language models, 2024.
- [33] Max Nye, Anders Andreassen, Alexander L Gaunt, et al. Show your work: Scratchpads for intermediate computation with language models. *arXiv preprint arXiv:2112.00114*, 2021.

- [34] OECD. *Preventing Corruption in Public Procurement*. OECD Publishing, 2016.
- [35] OpenAI. What are tokens and how to count them? <https://help.openai.com/en/articles/4936856-what-are-tokens-and-how-to-count-them>, 2024. Accessed: 2024-09-04.
- [36] Ethan Perez, Douwe Kiela, and Kyunghyun Cho. True few-shot learning with language models. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.
- [37] Alec Radford and Karthik Narasimhan. Improving language understanding by generative pre-training. 2018.
- [38] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- [39] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.
- [40] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2023.
- [41] Luke Reynolds and Kyle McDonell. Prompt programming for large language models: Beyond the few-shot paradigm. *arXiv preprint arXiv:2102.07350*, 2021.
- [42] Susan Rose-Ackerman. *Corruption and Government: Causes, Consequences, and Reform*. Cambridge University Press, 1999.
- [43] Sameendra Samarawickrama. Focused web crawling using named entity recognition for narrow domains. *International Journal of Research in Engineering and Technology*, 02:358–366, 03 2013.
- [44] Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the conll-2003 shared task: Language-independent named entity recognition, 2003.
- [45] Mourad Sarrouiti, Carson Tao, and Yoann Mamy Randriamihaja. Comparing encoder-only and encoder-decoder transformers for relation extraction from biomedical texts: An empirical study on ten benchmark datasets. In Dina Demner-Fushman, Kevin Bretonnel Cohen, Sophia Ananiadou, and Junichi Tsujii, editors, *Proceedings of the 21st Workshop on Biomedical Language Processing*, pages 376–382, Dublin, Ireland, May 2022. Association for Computational Linguistics.

- [46] Timo Schick and Hinrich Schütze. True Few-Shot Learning with Prompts—A Real-World Perspective. *Transactions of the Association for Computational Linguistics*, 10:716–731, 06 2022.
- [47] Isabel Segura-Bedmar, Paloma Martínez, and María Herrero-Zazo. SemEval-2013 task 9 : Extraction of drug-drug interactions from biomedical texts (DDI-Extraction 2013). In Suresh Manandhar and Deniz Yuret, editors, *Second Joint Conference on Lexical and Computational Semantics (\*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 341–350, Atlanta, Georgia, USA, June 2013. Association for Computational Linguistics.
- [48] Agam Shah, Ruchit Vithani, Abhinav Gullapalli, and Sudheer Chava. Finer: Financial named entity recognition dataset and weak-supervision model, 2023.
- [49] Wujun Shao, Pengli Ji, Dongwei Fan, Yaohua Hu, Xiaoran Yan, Chenzhou Cui, Linying Mi, Lang Chen, and Rui Zhang. Astronomical knowledge entity extraction in astrophysics journal articles via large language models, 2024.
- [50] Taylor Shin, Yasaman Razeghi, Robert L. Logan IV au2, Eric Wallace, and Sameer Singh. Autoprompt: Eliciting knowledge from language models with automatically generated prompts, 2020.
- [51] N Sobhana, Pabitra Mitra, and SK Ghosh. Conditional random field based named entity recognition in geological text. *International Journal of Computer Applications*, 1(3):143–147, 2010.
- [52] Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and policy considerations for deep learning in nlp. *arXiv preprint arXiv:1906.02243*, 2019. This paper discusses the high costs and energy consumption associated with training large deep learning models in NLP, including the creation and annotation of datasets.
- [53] SuperGLUE. Superglue. <https://super.gluebenchmark.com/>, 2024. Accessed: 2024-09-06.
- [54] Universitätsklinikum Tübingen. Skript zur recherche in pubmed. [https://www.medizin.uni-tuebingen.de/files/download/8YWbqX1Velm6DxKRLxkr9apM/Skript\\_PubMed.pdf](https://www.medizin.uni-tuebingen.de/files/download/8YWbqX1Velm6DxKRLxkr9apM/Skript_PubMed.pdf), 2021. Accessed: 2024-10-09.
- [55] Ali Alvi und Paresh Kharya. Using deepspeed and megatron to train megatron-turing nlg 530b, the world’s largest and most powerful generative language model. <https://www.microsoft.com/en-us/research/blog/using-deepspeed-and-megatron-to-train-megatron-turing-nlg-530b-the-worlds> 2021. Accessed: 2024-09-12.

- [56] European Union. eprocurement ontology. <https://ted.europa.eu/en/simap/eprocurement-ontology>, 2024. Accessed: 2024-09-08.
- [57] Linguistic Data Consortium University of Pennsylvania. Ace. <https://www ldc.upenn.edu/collaborations/past-projects/ace>, 2024. Accessed: 2024-09-08.
- [58] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Łukasz Kaiser Jones, Ilya Polosukhin, et al. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.
- [59] Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, and Guoyin Wang. Gpt-ner: Named entity recognition via large language models, 2023.
- [60] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.
- [61] Yucheng Wang, Bowen Yu, Hongsong Zhu, Tingwen Liu, Nan Yu, and Limin Sun. Discontinuous named entity recognition as maximal clique discovery, 2021.
- [62] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*, 2022.
- [63] Papers with Code. Named entity recognition on conll-2003 leaderboard. <https://paperswithcode.com/sota/named-entity-recognition-on-conll-2003-3>, 2023. Accessed: 2025-04-19.
- [64] V. Yadav and S. Bethard. A survey on recent advances in named entity recognition from deep learning models. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018)*, pages 2145–2158, 2019.
- [65] Zhigang Yang und Rongjian Zhao Yan Zhen, Puning Zhang. Frequent words and syntactic context integrated biomedical discontinuous named entity recognition method. *The Journal of Supercomputing*, 79:102153, 2023.
- [66] Hui-Syuan Yeh, Thomas Lavergne, and Pierre Zweigenbaum. Decorate the examples: A simple method of prompt design for biomedical relation extraction. In Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, H el ene Mazo, Jan Odiijk, and Stelios Piperidis, editors, *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3780–3787, Marseille, France, June 2022. European Language Resources Association.

- [67] Dharunish Yugeswardeenoo, Kevin Zhu, and Sean O’Brien. Question-analysis prompting improves llm performance in reasoning tasks, 2024.
- [68] Jingyi Zhang, Ying Zhang, Yufeng Chen, and Jinan Xu. Structure and label constrained data augmentation for cross-domain few-shot NER. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 518–530, Singapore, December 2023. Association for Computational Linguistics.
- [69] Tony Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. In *Proceedings of the 38th International Conference on Machine Learning*, pages 12697–12706. PMLR, 2021.
- [70] Tony Z. Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models, 2021.
- [71] Tiantian Zhu, Yang Qin, Qingcai Chen, Baotian Hu, and Yang Xiang. Enhancing entity representations with prompt learning for biomedical entity linking. In Lud De Raedt, editor, *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 4036–4042. International Joint Conferences on Artificial Intelligence Organization, 7 2022. Main Track.

## 8 Anhang

<b>Name</b>	<b>Synonym</b>	<b>Beschreibung</b>	<b>Wertebereich</b>
Geschäftszeichen	Vergabenummer, Identifikationsnummer, Vergabennr., Projektnummer, Interne Kennung	Ein Identifikationscode für die Ausschreibung. Wird häufig zur Referenzierung verwendet.	Eigenname (zum Beispiel 12345/2025)
Veröffentlichungsdatum	Datum der Veröffentlichung, Bekanntmachung	Das Datum, an dem die Ausschreibung veröffentlicht wurde.	Datum (zum Beispiel 01.01.2025)
Vergabestelle	Auftraggeber, Beschaffer	Die Organisation oder Behörde, die die Ausschreibung erstellt hat und den Auftrag vergibt.	Eigenname (zum Beispiel Deutsche Bahn)
Zweck der Ausschreibung	Ausschreibungsziel	Eine kurze Beschreibung des Ziels der Ausschreibung, wie zum Beispiel Beschaffung von Produkten oder Dienstleistungen.	Text (zum Beispiel „Bau einer neuen Brücke“)
Erfüllungsort	Ort der Ausführung, Leistungsort, Ort der Leistungserbringung	Der geografische Ort, an dem die Leistung erbracht oder das Projekt umgesetzt werden soll.	Eigenname (zum Beispiel Berlin)

<b>Name</b>	<b>Synonym</b>	<b>Beschreibung</b>	<b>Wertebereich</b>
Vergabeverfahren	Ausschreibungsverfahren	Das Verfahren, nach dem das Angebot ausgewählt wird (zum Beispiel offene Ausschreibung, Verhandlungsverfahren).	Text (zum Beispiel „Offenes Verfahren“)
Rechtsrahmen	-	Die gesetzlichen Grundlagen, die das Ausschreibungsverfahren regeln.	Text (zum Beispiel „VOB/A“)
Angebotsfrist	-	Das Datum, bis zu dem Angebote eingereicht werden müssen.	Datum (zum Beispiel 15.03.2025)
Zeitraum der Ausführung	-	Der Zeitraum, in dem das Projekt durchgeführt werden soll.	Zeitraum (zum Beispiel „01.04.2025 - 31.12.2025“)
Lose	-	Die Ausschreibung kann in mehrere Lose unterteilt sein, zum Beispiel verschiedene Teile eines Projekts.	Zahl (zum Beispiel „2 Lose“)
Bewerbungsfrist	-	Das Datum, bis zu dem sich Bieter für die Ausschreibung bewerben müssen.	Datum (zum Beispiel 01.02.2025)

<b>Name</b>	<b>Synonym</b>	<b>Beschreibung</b>	<b>Wertebereich</b>
Zuschlagskriterien	-	Die Kriterien, die für die Auswahl des erfolgreichen Angebots maßgeblich sind (zum Beispiel Preis, Qualität).	Text (zum Beispiel „Niedrigster Preis“)
Verhandlungsfrist	-	Zeitraum, in dem Verhandlungen über die Angebotsdetails stattfinden können.	Zeitraum (zum Beispiel „01.03.2025 - 15.03.2025“)
Auftragsvolumen	-	Der geschätzte Gesamtwert des Projekts oder der Beschaffung.	Zahl (zum Beispiel „500.000 EUR“)
Vertragsdauer	-	Die Dauer des Vertrags, der nach erfolgreicher Vergabe abgeschlossen wird.	Zeitraum (zum Beispiel „2 Jahre“)
Zahlungsbedingungen	-	Die Bedingungen, unter denen Zahlungen für den Auftrag erfolgen.	Text (zum Beispiel „Zahlung innerhalb von 30 Tagen nach Rechnungstellung“)
Zuschlagsstelle	-	Die Stelle, die den Zuschlag für das Angebot erteilt.	Eigenname (zum Beispiel „Bundesministerium für Verkehr und digitale Infrastruktur“)
Leistungsbeschreibung	-	Eine detaillierte Beschreibung der zu erbringenden Leistungen.	Text (zum Beispiel „Lieferung von 1000 Computern“)

<b>Name</b>	<b>Synonym</b>	<b>Beschreibung</b>	<b>Wertebereich</b>
Vertragspartner	-	Die Partei, mit der der Vertrag abgeschlossen wird.	Eigenname (zum Beispiel „XYZ GmbH“)

Tabelle 10: Übersicht von relevanten Entitäten in öffentlichen Ausschreibungen

<b>Index</b>	<b>Titel der Ausschreibung</b>	<b>Länge HTML</b>
1	Abschluss eines Vertrages über die Durchführung des Winterdienstes in der Bw-Liegenschaft HIL GmbH	29.575
2	Rahmenvertrag Uniformjacken Polizei	30.521
3	Instandsetzung Schwimmgreifer SG Luchs	25.236
4	SGB-Textausgabe 2025	30.168
5	Neubau eines Basisbaus am Albinmüller-Turm - Förderprogramm „Lebendige Zentren“; Heinrich-Heine-Platz 2, 39114 Magdeburg, Los 3.07 Fassadenarbeiten 30-ZV-0049/25	29.704
6	REZ SW UB AA Balingen	48.273
7	Projekträgerschaft zur Förderung von Projekten im Nationalen Artenhilfsprogramm	41.805
8	Konzeption, Organisation und Durchführung von konventionellen und digitalen Veranstaltungen für das Bundesministerium für Wirtschaft und Klimaschutz	72.016
9	Erneuerung Multifunktionsgeräte WV-2025-0012	16.212
10	Kanalsanierung 2025 (offene Bauweise) Hückeswagen WV-2025-0004	19.074
11	S-Bahn Köln, S13 Troisdorf-Bonn-Oberkassel, SSW Meindorf A59	12.243
12	Elektro 50 HZ Karlsruhe	10.617
13	BIM-Planung Empfangsgebäude Passau - TGA inkl. Bestandsaufnahme. - 50 Hz, ITK,HLS/GA	10.461
14	Barrierefreie Buskaps Bf. Loevenich 2025-0004-665-2	18.345
15	Rahmenvertrag Dachdeckerarbeiten im gesamten Stadtgebiet Köln mit 8 Losen 2025-0041-267-1	18.990
16	Ausschreibung Elektroarbeiten 73-00877-3470-8	23.796
17	Ausschreibung Rohbauarbeiten 70-00001-2880-338	32.070
18	Ausschreibung PV-Anlage 73-00460-2030-1	23.570
19	KTW - Energetische Sanierung Trakt C - Feste Möblierung ZV-2025-0045-Bau	23.401
20	75922 Fachplanungsleistungen Technische Ausrüstung, Klinik Lipperland FV15-25-0161-10-05	35.479
21	KG 410 / 420 Heizung und Sanitär 25A40081	19.208
22	Metallbauarbeiten 25A30066	18.673

<b>Index</b>	<b>Titel der Ausschreibung</b>	<b>Länge HTML</b>
23	Mikrowellenhochdruckautomat 221-028-Br-25	14.184
24	Beschaffung von drei Traktoren LAFO-2025-0034	38.434
25	KiTa Hangard - Elektroarbeiten SNEUNK-2025-0011	21.891
26	Entwicklung, Implementierung einer Buchungsplattform auf Basis von Biletado LKWND-2025-07	13.257
27	Rahmenvereinbarung über die Durchführung elektronischer Prüfungen 221-254-Pe-24	25.043
28	A.0435.120513_JLU GI SEM1_Technische Dämmung	21.285
29	Gestellung WD-Fahrzeug SM Bad Arolsen 2025 - 2029	26.511
30	Ö - 246410 SM Meißner - Erweiterung Betriebsgebäude - Rohbauarbeiten	19.973

Tabelle 11: Testdaten mit 30 Ausschreibungen inklusive Länge des HTMLs

<b>Index</b>	<b>benötigte Zeit [s]</b>	<b>verbrauchte Tokens</b>
1	134,64	10.392
2	92,29	10.749
3	76,39	8.814
4	101,94	10.416
5	122,01	11.029
6	236,49	22.087
7	77,25	14.157
8	279,68	30.976
9	33,20	5.846
10	71,88	9.515
11	66,26	6.832
12	52,98	5.856
13	50,58	5.964
14	73,13	9.003
15	107,11	9.294
16	40,58	7.549
17	40,74	9.949
18	83,55	10.098
19	89,93	12.065
20	144,46	17.318
21	68,66	9.520
22	68,66	9.220
23	51,57	7.106
24	199,72	19.286
25	93,75	9.217
26	62,76	6.702
27	104,79	12.595
28	114,40	10.703
29	101,76	13.243
30	76,04	10.125
MW ± Std.-Abw.	97,24 ± 56,08	11.187,53 ± 5.302,66

Tabelle 12: Metriken für die Zero-Shot Prompt-Strategie (benötigte Zeit in Sekunden, verbrauchte Tokens) für jede Ausschreibung

Index	Korrekt Positive	Falsch Positive	Falsch Negative	Precision	Recall	F1-Score
Veröffentlichungsdatum						
1	1	7	0	0,13	1,00	0,22
2	1	1	0	0,50	1,00	0,67
3	1	3	0	0,25	1,00	0,40
4	1	0	0	1,00	1,00	1,00
5	1	6	0	0,14	1,00	0,25
6	1	3	0	0,25	1,00	0,40
7	1	1	0	0,50	1,00	0,67
8	1	8	0	0,11	1,00	0,20
9	0	3	0	0,00	0,00	0,00
10	0	5	0	0,00	0,00	0,00
11	1	7	0	0,13	1,00	0,22
12	1	5	0	0,17	1,00	0,29
13	1	4	0	0,20	1,00	0,33
14	0	5	0	0,00	0,00	0,00
15	0	4	0	0,00	0,00	0,00
16	0	0	0	0,00	0,00	0,00
17	0	2	0	0,00	0,00	0,00
18	0	9	0	0,00	0,00	0,00
19	0	4	0	0,00	0,00	0,00
20	0	4	0	0,00	0,00	0,00
21	0	5	0	0,00	0,00	0,00
22	0	5	0	0,00	0,00	0,00
23	0	2	0	0,00	0,00	0,00
24	0	8	0	0,00	0,00	0,00
25	0	5	0	0,00	0,00	0,00
26	0	3	0	0,00	0,00	0,00
27	0	4	0	0,00	0,00	0,00
28	0	5	0	0,00	0,00	0,00
29	0	4	0	0,00	0,00	0,00
30	0	5	0	0,00	0,00	0,00
MW ±				0,11 ±	0,37 ±	0,15 ±
Std.- Abw.				0,22	0,49	0,25

Index	Korrekt Positive	Falsch Positive	Falsch Negative	Precision	Recall	F1-Score
<b>Vergabestelle</b>						
1	1	1	0	0,50	1,00	0,67
2	1	1	1	0,50	0,50	0,50
3	1	1	1	0,50	0,50	0,50
4	1	1	1	0,50	0,50	0,50
5	1	1	1	0,50	0,50	0,50
6	1	5	0	0,17	1,00	0,29
7	1	2	0	0,33	1,00	0,50
8	1	20	0	0,05	1,00	0,09
9	0	1	1	0,00	0,00	0,00
10	0	2	1	0,00	0,00	0,00
11	1	1	0	0,50	1,00	0,67
12	0	1	1	0,00	0,00	0,00
13	1	0	0	1,00	1,00	1,00
14	0	1	1	0,00	0,00	0,00
15	0	1	1	0,00	0,00	0,00
16	0	1	1	0,00	0,00	0,00
17	0	2	1	0,00	0,00	0,00
18	0	1	1	0,00	0,00	0,00
19	0	0	1	0,00	0,00	0,00
20	0	3	1	0,00	0,00	0,00
21	0	2	1	0,00	0,00	0,00
22	0	2	1	0,00	0,00	0,00
23	0	1	1	0,00	0,00	0,00
24	0	2	1	0,00	0,00	0,00
25	0	1	1	0,00	0,00	0,00
26	0	1	1	0,00	0,00	0,00
27	0	3	1	0,00	0,00	0,00
28	0	1	1	0,00	0,00	0,00
29	0	5	1	0,00	0,00	0,00
30	0	1	1	0,00	0,00	0,00
MW ±				0,15 ±	0,27 ±	0,17 ±
Std.- Abw.				0,26	0,41	0,28

Index	Korrekt Positive	Falsch Positive	Falsch Negative	Precision	Recall	F1-Score
<b>Geschäftszeichen</b>						
1	2	0	0	1,00	1,00	1,00
2	2	0	0	1,00	1,00	1,00
3	2	0	0	1,00	1,00	1,00
4	2	0	0	1,00	1,00	1,00
5	3	0	0	1,00	1,00	1,00
6	2	0	0	1,00	1,00	1,00
7	2	0	0	1,00	1,00	1,00
8	3	0	0	1,00	1,00	1,00
9	1	0	0	1,00	1,00	1,00
10	2	0	0	1,00	1,00	1,00
11	0	0	1	0,00	0,00	0,00
12	0	0	1	0,00	0,00	0,00
13	0	0	1	0,00	0,00	0,00
14	2	0	0	1,00	1,00	1,00
15	2	0	0	1,00	1,00	1,00
16	2	0	0	1,00	1,00	1,00
17	2	3	0	0,40	1,00	0,57
18	2	2	0	0,50	1,00	0,67
19	2	0	0	1,00	1,00	1,00
20	0	0	0	0,00	0,00	0,00
21	2	0	0	1,00	1,00	1,00
22	2	0	0	1,00	1,00	1,00
23	2	0	0	1,00	1,00	1,00
24	1	0	0	1,00	1,00	1,00
25	2	0	0	1,00	1,00	1,00
26	2	0	0	1,00	1,00	1,00
27	2	2	0	0,50	1,00	0,67
28	1	0	0	1,00	1,00	1,00
29	1	0	0	1,00	1,00	1,00
30	1	0	0	1,00	1,00	1,00
MW ±				0,81 ±	0,87 ±	0,83 ±
Std.- Abw.				0,36	0,35	0,35

Index	Korrekt Positive	Falsch Positive	Falsch Negative	Precision	Recall	F1-Score
<b>Erfüllungsort</b>						
1	0	3	0	0,00	0,00	0,00
2	0	3	0	0,00	0,00	0,00
3	0	1	0	0,00	0,00	0,00
4	0	1	1	0,00	0,00	0,00
5	1	2	1	0,33	0,50	0,40
6	0	4	4	0,00	0,00	0,00
7	0	3	2	0,00	0,00	0,00
8	0	6	2	0,00	0,00	0,00
9	1	0	1	1,00	0,50	0,67
10	1	1	0	0,50	1,00	0,67
11	1	0	0	1,00	1,00	1,00
12	1	0	0	1,00	1,00	1,00
13	1	0	0	1,00	1,00	1,00
14	1	0	0	1,00	1,00	1,00
15	1	0	0	1,00	1,00	1,00
16	1	2	0	0,33	1,00	0,50
17	0	4	1	0,00	0,00	0,00
18	1	2	0	0,33	1,00	0,50
19	0	1	1	0,00	0,00	0,00
20	0	5	1	0,00	0,00	0,00
21	0	3	1	0,00	0,00	0,00
22	1	0	0	1,00	1,00	1,00
23	1	0	0	1,00	1,00	1,00
24	0	7	4	0,00	0,00	0,00
25	1	1	0	0,50	1,00	0,67
26	1	0	0	1,00	1,00	1,00
27	0	2	1	0,00	0,00	0,00
28	1	0	0	1,00	1,00	1,00
29	0	4	1	0,00	0,00	0,00
30	1	0	0	1,00	1,00	1,00
MW ±				0,43 ±	0,50 ±	0,45 ±
Std.- Abw.				0,46	0,49	0,46

Index	Korrekt Positive	Falsch Positive	Falsch Negative	Precision	Recall	F1-Score
<b>Über alle 4 Entitäten</b>						
1				0,41	0,75	0,47
2				0,50	0,63	0,54
3				0,44	0,63	0,48
4				0,63	0,63	0,63
5				0,49	0,75	0,54
6				0,35	0,75	0,42
7				0,46	0,75	0,54
8				0,29	0,75	0,32
9				0,50	0,38	0,42
10				0,38	0,50	0,42
11				0,41	0,75	0,47
12				0,29	0,50	0,32
13				0,55	0,75	0,58
14				0,50	0,50	0,50
15				0,50	0,50	0,50
16				0,33	0,50	0,38
17				0,10	0,25	0,14
18				0,21	0,25	0,29
19				0,25	0,25	0,25
20				0,00	0,00	0,00
21				0,25	0,25	0,25
22				0,50	0,50	0,50
23				0,50	0,50	0,50
24				0,25	0,25	0,25
25				0,38	0,50	0,42
26				0,50	0,50	0,50
27				0,13	0,25	0,17

Index	Korrekt Positive	Falsch Positive	Falsch Negative	Precision	Recall	F1-Score
28				0,50	0,50	0,50
29				0,25	0,25	0,25
30				0,50	0,50	0,50
MW $\pm$				0,38 $\pm$	0,50 $\pm$	0,40 $\pm$
Std.- Abw.				0,15	0,20	0,15

Tabelle 13: Leistungsmetriken für die Zero-Shot Prompt-Strategie (Korrekt Positive, Falsch Positive, Falsch Negative, Precision, Recall, und F1-Score) als Mittelwert und Standardabweichung für jede Ausschreibung und pro Entität sowie über alle 4 Entitäten

<b>Index</b>	<b>benötigte Zeit [s]</b>	<b>verbrauchte Tokens</b>
1	81,55	15.705
2	79,37	16.093
3	74,52	14.296
4	78,50	15.954
5	83,29	16.483
6	83,36	21.674
7	72,94	19.161
8	71,22	25.675
9	61,81	13.508
10	80,52	14.998
11	44,92	12.297
12	43,32	11.480
13	42,74	11.453
14	66,43	14.496
15	78,70	14.792
16	83,41	15.850
17	116,85	20.286
18	23,10	12.119
19	93,23	17.554
20	131,06	22.813
21	65,59	14.985
22	71,55	14.692
23	50,28	12.605
24	147,67	24.768
25	79,21	14.717
26	49,13	12.185
27	100,48	18.056
28	84,86	16.421
29	98,96	18.670
30	85,23	15.776
MW ± Std.-Abw.	77,46 ± 25,85	16.318,73 ± 3.727,56

Tabelle 14: Metriken für die One-Shot Prompt-Strategie (benötigte Zeit in Sekunden, verbrauchte Tokens) für jede Ausschreibung

Index	Szenario	Korrekt Positi- ve	Falsch Positi- ve	Falsch Negati- ve	Precision	Recall	F1- Score
<b>Veröffentlichungsdatum</b>							
1	2	1	6	0	0,14	1,00	0,25
2	2	1	4	0	0,20	1,00	0,33
3	2	1	2	0	0,33	1,00	0,50
4	2	1	0	0	1,00	1,00	1,00
5	2	1	0	0	1,00	1,00	1,00
6	1	1	0	0	1,00	1,00	1,00
7	1	1	0	0	1,00	1,00	1,00
8	1	1	0	0	1,00	1,00	1,00
9	1	0	4	0	0,00	0,00	0,00
10	1	0	5	0	0,00	0,00	0,00
11	1	1	7	0	0,13	1,00	0,22
12	1	1	5	0	0,17	1,00	0,29
13	1	1	4	0	0,20	1,00	0,33
14	1	0	5	0	0,00	0,00	0,00
15	1	0	5	0	0,00	0,00	0,00
16	1	0	5	0	0,00	0,00	0,00
17	1	0	6	0	0,00	0,00	0,00
18	1	0	2	0	0,00	0,00	0,00
19	1	0	4	0	0,00	0,00	0,00
20	1	0	4	0	0,00	0,00	0,00
21	1	0	8	0	0,00	0,00	0,00
22	1	0	5	0	0,00	0,00	0,00
23	1	0	2	0	0,00	0,00	0,00
24	1	0	8	0	0,00	0,00	0,00
25	1	0	5	0	0,00	0,00	0,00
26	1	0	2	0	0,00	0,00	0,00
27	1	0	3	0	0,00	0,00	0,00
28	1	0	5	0	0,00	0,00	0,00
29	1	0	4	0	0,00	0,00	0,00
30	1	0	0	0	0,00	0,00	0,00
MW ±	-				0,21 ±	0,37 ±	0,23 ±
Std.- Abw.					0,37	0,49	0,37

Index	Szenario	Korrekt Positi- ve	Falsch Positi- ve	Falsch Negati- ve	Precision	Recall	F1- Score
MW ± Std.- Abw.	1				0,14 ± 0,33	0,24 ± 0,44	0,15 ± 0,33
MW ± Std.- Abw.	2				0,54 ± 0,43	1,00 ± 0,00	0,62 ± 0,36

#### Vergabestelle

1	2	1	0	0	1,00	1,00	1,00
2	2	1	1	1	0,50	0,50	0,50
3	2	1	1	1	0,50	0,50	0,50
4	2	1	1	1	0,50	0,50	0,50
5	2	1	1	1	0,50	0,50	0,50
6	1	1	0	1	1,00	0,50	0,67
7	1	2	1	0	0,67	1,00	0,80
8	1	2	2	0	0,50	1,00	0,67
9	1	0	1	1	0,00	0,00	0,00
10	1	0	1	1	0,00	0,00	0,00
11	1	0	0	1	0,00	0,00	0,00
12	1	0	0	1	0,00	0,00	0,00
13	1	0	0	1	0,00	0,00	0,00
14	1	0	1	1	0,00	0,00	0,00
15	1	0	1	1	0,00	0,00	0,00
16	1	0	2	1	0,00	0,00	0,00
17	1	0	1	1	0,00	0,00	0,00
18	1	0	1	1	0,00	0,00	0,00
19	1	0	0	1	0,00	0,00	0,00
20	1	0	0	1	0,00	0,00	0,00
21	1	0	2	1	0,00	0,00	0,00
22	1	0	1	1	0,00	0,00	0,00
23	1	0	1	1	0,00	0,00	0,00
24	1	0	1	1	0,00	0,00	0,00
25	1	0	2	1	0,00	0,00	0,00
26	1	1	0	1	1,00	0,50	0,67
27	1	0	1	1	0,00	0,00	0,00

Index	Szenario	Korrekt Positi- ve	Falsch Positi- ve	Falsch Negati- ve	Precision	Recall	F1- Score
28	1	0	1	1	0,00	0,00	0,00
29	1	0	0	1	0,00	0,00	0,00
30	1	0	1	1	0,00	0,00	0,00
MW ± Std.- Abw.	-				0,21 ± 0,34	0,20 ± 0,34	0,19 ± 0,31
MW ± Std.- Abw.	1				0,13 ± 0,31	0,12 ± 0,30	0,11 ± 0,26
MW ± Std.- Abw.	2				0,60 ± 0,22	0,60 ± 0,22	0,60 ± 0,22
<b>Geschäftszeichen</b>							
1	2	2	0	0	1,00	1,00	1,00
2	2	2	0	0	1,00	1,00	1,00
3	2	0	0	2	0,00	0,00	0,00
4	2	2	0	0	1,00	1,00	1,00
5	2	3	0	0	1,00	1,00	1,00
6	1	1	0	1	1,00	0,50	0,67
7	1	2	0	1	1,00	0,67	0,80
8	1	2	0	1	1,00	0,67	0,80
9	1	1	0	1	1,00	0,50	0,67
10	1	1	0	1	1,00	0,50	0,67
11	1	0	0	1	0,00	0,00	0,00
12	1	0	0	1	0,00	0,00	0,00
13	1	0	0	1	0,00	0,00	0,00
14	1	1	0	1	1,00	0,50	0,67
15	1	1	0	1	1,00	0,50	0,67
16	1	2	0	0	1,00	1,00	1,00
17	1	0	0	2	0,00	0,00	0,00
18	1	2	0	0	1,00	1,00	1,00
19	1	0	0	2	0,00	0,00	0,00
20	1	0	3	0	0,00	0,00	0,00
21	1	1	0	1	1,00	0,50	0,67

Index	Szenario	Korrekt Positi- ve	Falsch Positi- ve	Falsch Negati- ve	Precision	Recall	F1- Score
22	1	1	0	1	1,00	0,50	0,67
23	1	1	0	1	1,00	0,50	0,67
24	1	0	0	2	0,00	0,00	0,00
25	1	1	0	1	1,00	0,50	0,67
26	1	1	0	1	1,00	0,50	0,67
27	1	0	2	2	0,00	0,00	0,00
28	1	1	0	0	1,00	1,00	1,00
29	1	0	0	1	0,00	0,00	0,00
30	1	1	0	0	1,00	1,00	1,00
MW ± Std.- Abw.	-				0,67 ± 0,48	0,48 ± 0,40	0,54 ± 0,41
MW ± Std.- Abw.	1				0,64 ± 0,49	0,41 ± 0,36	0,49 ± 0,39
MW ± Std.- Abw.	2				0,80 ± 0,45	0,80 ± 0,45	0,80 ± 0,45
<b>Erfüllungsort</b>							
1	2	1	0	0	1,00	1,00	1,00
2	2	0	3	1	0,00	0,00	0,00
3	2	0	1	1	0,00	0,00	0,00
4	2	0	1	1	0,00	0,00	0,00
5	2	1	1	0	0,50	1,00	0,67
6	1	0	2	4	0,00	0,00	0,00
7	1	0	0	2	0,00	0,00	0,00
8	1	0	3	2	0,00	0,00	0,00
9	1	1	1	0	0,50	1,00	0,67
10	1	1	1	0	0,50	1,00	0,67
11	1	1	0	0	1,00	1,00	1,00
12	1	1	0	0	1,00	1,00	1,00
13	1	1	0	0	1,00	1,00	1,00
14	1	1	0	0	1,00	1,00	1,00
15	1	1	0	0	1,00	1,00	1,00

Index	Szenario	Korrekt Positi- ve	Falsch Positi- ve	Falsch Negati- ve	Precision	Recall	F1- Score
16	1	1	1	0	0,50	1,00	0,67
17	1	0	6	1	0,00	0,00	0,00
18	1	1	0	0	1,00	1,00	1,00
19	1	0	2	1	0,00	0,00	0,00
20	1	0	1	5	0,00	0,00	0,00
21	1	1	0	0	1,00	1,00	1,00
22	1	1	0	0	1,00	1,00	1,00
23	1	1	1	0	0,50	1,00	0,67
24	1	0	6	4	0,00	0,00	0,00
25	1	1	1	0	0,50	1,00	0,67
26	1	1	0	0	1,00	1,00	1,00
27	1	0	2	1	0,00	0,00	0,00
28	1	1	0	0	1,00	1,00	1,00
29	1	0	1	1	0,00	0,00	0,00
30	1	1	0	0	1,00	1,00	1,00
MW ± Std.- Abw.	-				0,50 ± 0,45	0,60 ± 0,50	0,53 ± 0,46
MW ± Std.- Abw.	1				0,54 ± 0,45	0,64 ± 0,49	0,57 ± 0,46
MW ± Std.- Abw.	2				0,30 ± 0,45	0,40 ± 0,55	0,33 ± 0,47
<b>Über alle 4 Entitäten</b>							
1	2				0,79	1,00	0,81
2	2				0,43	0,63	0,46
3	2				0,21	0,38	0,25
4	2				0,63	0,63	0,63
5	2				0,75	0,88	0,79
6	1				0,75	0,50	0,58
7	1				0,67	0,67	0,65
8	1				0,63	0,67	0,62
9	1				0,38	0,38	0,33

Index	Szenario	Korrekt Positi- ve	Falsch Positi- ve	Falsch Negati- ve	Precision	Recall	F1- Score
10	1				0,38	0,38	0,33
11	1				0,28	0,50	0,31
12	1				0,29	0,50	0,32
13	1				0,30	0,50	0,33
14	1				0,50	0,38	0,42
15	1				0,50	0,38	0,42
16	1				0,38	0,50	0,42
17	1				0,00	0,00	0,00
18	1				0,50	0,50	0,50
19	1				0,00	0,00	0,00
20	1				0,00	0,00	0,00
21	1				0,50	0,38	0,42
22	1				0,50	0,38	0,42
23	1				0,38	0,38	0,33
24	1				0,00	0,00	0,00
25	1				0,38	0,38	0,33
26	1				0,75	0,50	0,58
27	1				0,00	0,00	0,00
28	1				0,50	0,50	0,50
29	1				0,00	0,00	0,00
30	1				0,50	0,50	0,50
MW ± Std.- Abw.	-				0,39 ± 0,25	0,41 ± 0,25	0,37 ± 0,23
MW ± Std.- Abw.	1				0,36 ± 0,24	0,35 ± 0,22	0,33 ± 0,21
MW ± Std.- Abw.	2				0,56 ± 0,24	0,70 ± 0,24	0,59 ± 0,24

Tabelle 15: Leistungsmetriken für die One-Shot Prompt-Strategie (Korrekt Positive, Falsch Positive, Falsch Negative, Precision, Recall, und F1-Score) als Mittelwert und Standardabweichung für jede Ausschreibung und pro Entität sowie über alle 4 Entitäten

Quellcode 1: Extrahierte HTML einer Ausschreibung von dem e-Vergabe Online-Portal

```
<div id="content" class="no-bg">
  <main class="container">
    <div><h2>Ausschreibungsdetails</h2>
      <p>
        <p></p></p></div>
    <div class="distance"></div>
    <form class="form-horizontal">
      <div class="procedure-infos">
        <section class="clearfix"><h4>Lieferung eines Ionenchromatographen
          gem
          Leistungsbeschreibung inkl.
          Installation, Inbetriebnahme & Einweisung/Schulung</h4>
          <div class="ev-tooltip">
            <button class="ev-tooltip-toggle" type="button"
              data-toggle="popover"
              name="headerSection:contentLinkTenderingProcedureDetails:
                tooltipContainer:toggleButton"
              id="id106"
              aria-label="Zusatzinformation"><span
                class="ev-tooltip-icon glyphicon glyphicon-info-sign"
                aria-hidden="true"></span></button>
            <div class="ev-tooltip-content">
              <div class="ev-tooltip-body_wiki-markup-container"><p>Zur
                Einhaltung der Teilnahmefrist whlen Sie vor
                deren Ablauf 'Teilnahme aktivieren' und laden mit Hilfe des
                AnA-Web die Vergabeunterlagen herunter.</p>
              </div>
              <span class="ev-tooltip-footer"><a
                href="https://www.evergabe-online.info/e-Vergabe/DE/3%20
                  Unternehmen/BereitsRegistriert/node_BereitsRegistriert.html
                ">Mehr lesen</a></span>
            </div>
          </div>
        </div>
      </div>
    </div>
    <div class="row_procedure-info">
      <div class="col-xs-12_col-sm-7">
        <div class="form-group" id="id107"><label
          class="col-xs-12_col-sm-6_col-md-5_control-label"><span>
            Verffentlichungsdatum:</span></label>
        <div class="col-xs-12_col-sm-6_col-md-7"><p
```

```

        class="form-control-static">21.08.2024</p></div>
</div>
<div class="form-group" id="id108"><label
  class="col-xs-12_col-sm-6_col-md-5_control-label"><span>
    Teilnahme aktivieren bis:</span></label>
  <div class="col-xs-12_col-sm-6_col-md-7"><p
    class="form-control-static">16.09.2024 09:00</p></div>
</div>
<div class="form-group" id="id109"><label
  class="col-xs-12_col-sm-6_col-md-5_control-label"><span>
    Abgabefrist Angebot:</span></label>
  <div class="col-xs-12_col-sm-6_col-md-7"><p
    class="form-control-static">16.09.2024 09:00</p></div>
</div>
<div class="form-group" id="id10a"><label
  class="col-xs-12_col-sm-6_col-md-5_control-label">
    Geschäftszeichen:</label>
  <div class="col-xs-12_col-sm-6_col-md-7"><p
    class="form-control-static">102-10125345</p></div>
</div>
<div class="form-group" id="id10b"><label
  class="col-xs-12_col-sm-6_col-md-5_control-label">
    Vergabestelle:</label>
  <div class="col-xs-12_col-sm-6_col-md-7"><p
    class="form-control-static">Bundesanstalt fr
    Geowissenschaften und Rohstoffe</p></div>
</div>
<div class="form-group" id="id10c"><label
  class="col-xs-12_col-sm-6_col-md-5_control-label">Letzte
  nderung:</label>
  <div class="col-xs-12_col-sm-6_col-md-7"><p
    class="form-control-static">21.08.2024 15:31</p></div>
</div>
</div>
<div class="col-xs-12_col-sm-5">
  <form id="id10d" method="post"
    action="/tenderdetails.html?2-2.-headerSection-
    procedureActionsForm&id=710053"></form>
  <div class="bordered-section"><h3 class="bordered-section-title">
    Meine e-Vergabe</h3>
  <div class="bordered-section-body">
    <ul class="link-box_list-group">
      <li class="list-group-item"><a class="icon" id="idf4"

```

```

href="/tenderdetails.html
?2-2.-headerSection-
notLoggedInSubscribeButtonContainer
-
notLoggedInSubscribeButton
&id=710053"
title="Teilnahme_aktivieren"
rel="nofollow"><i
class="fa_fa-arrow-circle-o-right"></i><span>Teilnahme
aktivieren</span></a>
</li>
</ul>
</div>
</div>
</div>
</div>
<div class="button-group"><a class="btn_btn-primary" id="id10e"
href="/tenderdocuments.html?id=710053">
Ausschreibungsunterlagen
einsehen</a></div>
</div>
<hr>
</form>
<div>
<div class="distance-higher"></div>
<div id="htmlDocument">
<div class="procedure-details_nat-form">
<div class="procedure-header">
<div class="footer-content"><strong>DE</strong> Formular &nbsp;
Auftragsbekanntmachung
</div>
<h2>Auftragsbekanntmachung</h2>
<h3>ffentliche Ausschreibung nach UVgO</h3>
<p>Liefer- / Dienstleistungsauftrag</p></div>
<div class="section">
<div class="chapter">
<h4>
<div class="chapter-number">1.</div>
<div class="chapter-headline">ffentlicher Auftraggeber
(Vergabestelle)
</div>
</h4>
<div class="chapter-content">

```

```

<div class="chapter">
  <h4>
    <div class="chapter-number">a</div>
    <div class="chapter-headline">Hauptauftraggeber (zur
      Angebotsabgabe auffordernde Stelle)
    </div>
  </h4>
  <div class="chapter-content">
    <div><span class="content-left">Name:</span> <span
      class="content-right">Bundesanstalt fr Geowissenschaften
      und Rohstoffe (BGR)</span>
    </div>
    <div><span class="content-left">Strae, Hausnummer:</span>
      <span
        class="content-right">Stilleweg 2</span></div>
    <div><span class="content-left">Postleitzahl (PLZ):</span>
      <span class="content-right">30655</span>
    </div>
    <div><span class="content-left">Ort:</span> <span
      class="content-right">Hannover</span></div>
    <div><span class="content-left">E-Mail:</span> <span
      class="content-right"><a
      href="mailto:luca.magiera@bgr.de">luca.magiera@bgr.de</
      a> </span>
    </div>
  </div>
</div>
<div class="chapter">
  <h4>
    <div class="chapter-number">b</div>
    <div class="chapter-headline">Zuschlag erteilende Stelle
    </div>
  </h4>
  <div class="chapter-content"><span>Wie Hauptauftraggeber
    siehe&nbsp;a)</span>
  </div>
</div>
</div>
<div class="section">
  <div class="chapter">

```

```

<h4>
  <div class="chapter-number">2.</div>
  <div class="chapter-headline">Angaben zum Verfahren</div>
</h4>
<div class="chapter-content">
  <div class="chapter">
    <h4>
      <div class="chapter-number">a)</div>
      <div class="chapter-headline">Verfahrensart</div>
    </h4>
    <div class="chapter-content">ffentliche Ausschreibung nach
      UVgO
    </div>
  </div>
  <div class="chapter">
    <h4>
      <div class="chapter-number">b)</div>
      <div class="chapter-headline">Vertragsart</div>
    </h4>
    <div class="chapter-content">Liefer- /
      Dienstleistungsauftrag
    </div>
  </div>
  <div class="chapter">
    <h4>
      <div class="chapter-number">c)</div>
      <div class="chapter-headline">Geschftszeichen</div>
    </h4>
    <div class="chapter-content">102-10125345</div>
  </div>
</div>
</div>
<div class="section">
  <div class="chapter">
    <h4>
      <div class="chapter-number">3.</div>
      <div class="chapter-headline">Angaben zu Angeboten</div>
    </h4>
    <div class="chapter-content">
      <div class="chapter">
        <h4>
          <div class="chapter-number">a)</div>

```

```

    <div class="chapter-headline">Form der Angebote</div>
</h4>
<div class="chapter-content">
  <ul
    class="bullet-list">
    <li>elektronisch
      <ul class="bullet-list_bullet-list-level">
        <li>ohne elektronische Signatur (Textform)</li>
        <li>mit fortgeschrittener elektronischer Signatur /
          fortgeschrittenem elektronischen Siegel
        </li>
        <li>mit qualifizierter elektronischer Signatur /
          qualifiziertem elektronischen Siegel
        </li>
      </ul>
    </li>
  </ul>
  <br
  >
</div>
</div>
<div class="chapter">
  <h4>
    <div class="chapter-number">b</div>
    <div class="chapter-headline">Fristen</div>
  </h4>
  <div class="chapter-content">
    <div class="chapter_no-chapter-number">
      <h4>
        <div class="chapter-headline">Ablauf der Angebotsfrist
        </div>
      </h4>
      <div class="chapter-content">16.09.2024 – 09:00 Uhr</div>
    >
  </div>
  <div class="chapter_no-chapter-number">
    <h4>
      <div class="chapter-headline">Ablauf der Bindefrist
      </div>
    </h4>
    <div class="chapter-content">15.10.2024</div>
  </div>
</div>

```

```

</div>
<div class="chapter">
  <h4>
    <div class="chapter-number">c</div>
    <div class="chapter-headline">Sprache</div>
  </h4>
  <div class="chapter-content">&lt;deutsch</div>
</div>
</div>
</div>
</div>
<div class="section">
  <div class="chapter">
    <h4>
      <div class="chapter-number">4.</div>
      <div class="chapter-headline">Angaben zu Vergabeunterlagen</div>
    </h4>
    <div class="chapter-content">
      <div class="chapter">
        <h4>
          <div class="chapter-number">a</div>
          <div class="chapter-headline">Vertraulichkeit</div>
        </h4>
        <div class="chapter-content">Die Vergabeunterlagen stehen fr
          einen uneingeschrnkten und
          vollstndigen direkten Zugang gebhrenfrei zur Verfgung
          unter<br
          >
          <a
            href="https://www.evergabe-online.de/tenderdetails.html?id=710053"
            target="_blank">https://www.evergabe-online.de/
            tenderdetails.html?id=710053</a>
        </div>
      </div>
    </div>
  <div class="chapter">
    <h4>
      <div class="chapter-number">b</div>
      <div class="chapter-headline">Zugriff auf die
        Vergabeunterlagen
      </div>
    </h4>
  </div>
</div>

```

```

    <div class="chapter-content">Vergabeunterlagen werden nur
      elektronisch zur Verfügung gestellt
    </div>
  </div>
  <div class="chapter">
    <h4>
      <div class="chapter-number">c</div>
      <div class="chapter-headline">Zuständige Stelle</div>
    </h4>
    <div class="chapter-content">
      <div class="chapter_no-chapter-number"><h4></h4>
        <div class="chapter-content"><span>Hauptauftraggeber
          siehe 1.a)</span>
        </div>
      </div>
    </div>
  </div>
  <div class="chapter">
    <h4>
      <div class="chapter-number">d</div>
      <div class="chapter-headline">Anforderungsfrist</div>
    </h4>
    <div class="chapter-content">16.09.2024 – 09:00 Uhr</div>
  </div>
</div>
<div class="section">
  <div class="chapter">
    <h4>
      <div class="chapter-number">5.</div>
      <div class="chapter-headline">Angaben zur Leistung</div>
    </h4>
    <div class="chapter-content">
      <div class="chapter">
        <h4>
          <div class="chapter-number">a</div>
          <div class="chapter-headline">Art und Umfang der Leistung
        </div>
        </h4>
        <div class="chapter-content">Gegenstand dieses Vertrags ist
          die Lieferung eines
          Ionenchromatographiesystems inkl. Installation,

```

```

        Inbetriebnahme, Einweisung/Schulung durch den
        Auftragnehmer.<br></div>
</div>
<div class="chapter">
  <h4>
    <div class="chapter-number">c</div>
    <div class="chapter-headline">Ort der Leistungserbringung
    </div>
  </h4>
  <div class="chapter-content">DAP Raum D24 Hannover
    Incoterms
    2010 – Stilleweg 2, 30655<br>
    Hannover<br></div>
</div>
</div>
</div>
<div class="section">
  <div class="chapter">
    <h4>
      <div class="chapter-number">7.</div>
      <div class="chapter-headline">Zulassung von Nebenangeboten<
      /div>
    </h4>
    <div class="chapter-content">Nein</div>
  </div>
</div>
<div class="section">
  <div class="chapter">
    <h4>
      <div class="chapter-number">8.</div>
      <div class="chapter-headline">Bestimmungen ber die
      Ausführungsfrist
      </div>
    </h4>
    <div class="chapter-content">Der Auftragnehmer erbringt die
      Leistungen vollstndig bis zum 30.11.2024.<br>
      Eine frhere Auftragsausfhrung ist ausdrcklich erwnscht.<br>
    </div>
  </div>
</div>
<div class="section">
  <div class="chapter">

```

```

<h4>
  <div class="chapter-number">9.</div>
  <div class="chapter-headline">Hhe geforderter
    Sicherheitsleistungen
  </div>
</h4>
  <div class="chapter-content">Keine.<br></div>
</div>
<div class="section">
  <div class="chapter">
    <h4>
      <div class="chapter-number">10.</div>
      <div class="chapter-headline">Wesentliche
        Zahlungsbedingungen
      </div>
    </h4>
    <div class="chapter-content">Der Auftragnehmer kann die
      Zahlung
      der vereinbarten Vergütung nach
      vollstndiger<br> Erbringung der Leistung durch den
      Auftragnehmer und Abnahme der Leistung durch<br> den
      Auftraggeber verlangen. Teilzahlungen sind nur zugelassen,
      sofern in diesem<br> Vertrag unter Ziff. 3.2
      Ergnzende Vertragsbedingugen in sich abgeschlossenen und
      eigenstndig nutzbaren<br> Teilleistungen
      definiert wurden.<br> Zahlungen des Auftraggebers erfolgen mit
      einem Zahlungsziel von 30 Tagen ohne<br>
      Abzug oder innerhalb der angebotenen Skontofrist unter Abzug
      des
      angebotenen<br> Skontos nach Eingang
      der Rechnung auf das vom Auftragnehmer in der Rechnung<br>
      benannte Konto.<br> Magebend fr die
      Rechtzeitigkeit der Zahlung ist der Zugang des
      berweisungsauf-<br> trags beim Zahlungsinstitut des
      Auftraggebers.<br></div>
    </div>
  </div>
<div class="section">
  <div class="chapter">
    <h4>
      <div class="chapter-number">11.</div>
      <div class="chapter-headline">Unterlagen und Anforderungen
  
```

zur  
 Beurteilung der Eignung des Bieters und  
 des Nichtvorliegens von Ausschlussgründen  
 </div>  
 </h4>  
 <div class="chapter-content">1. Unterlagen zur Beurteilung der  
 Eignung des Bieters und des  
 Nichtvorliegens von Ausschlussgründen<br> Der Bieter muss  
 nachweisbar in der Lage sein, das verlangte  
 Aufgabenspektrum<br> kompetent und fristgemäß durchzuführen.  
 Er  
 hat mit seinem Angebot die<br>  
 nachfolgend genannten Unterlagen zum Nachweis seiner Eignung  
 und  
 des<br> Nichtvorliegens von  
 Ausschlussgründen vorzulegen. Es handelt sich ausnahmslos<br>  
 um  
 Ausschlusskriterien. Das bedeutet, dass  
 der Bieter ausgeschlossen wird, wenn er<br> die Anforderungen  
 nicht erfüllt.<br> Der Bieter muss alle  
 nachfolgend geforderten Nachweise zur Eignung mit der<br>  
 Angebotsabgabe vorlegen.<br> 1.1 Befähigung  
 und Erlaubnis zur Berufsausübung<br> Der den  
 Vergabeunterlagen  
 beigefügte Vordruck Eigenerklärung zu  
 den<br> Ausschlussgründen ist zu unterschreiben und dem  
 Angebot beizufügen.<br> Vor der  
 Auftragsvergabe wird von der Vergabestelle eine<br>  
 Gewerbezentralregisterauskunft eingeholt. Für einen  
 Zuschlag kommt nur ein Bieter<br> in Frage, der keine  
 auftragsverhindernden Eintragungen besitzt.<br>  
 1.2 Nachweise zur wirtschaftlichen und finanziellen  
 Leistungsfähigkeit<br> Der den Vergabeunterlagen  
 beigefügte Vordruck Eigenerklärung<br> Unternehmensangaben ist  
 zu unterschreiben und dem Angebot  
 beizufügen.<br> 1.3 Nachweise zur technischen und beruflichen  
 Leistungsfähigkeit<br> 1.3.1 Nachweis der  
 Leistungsmerkmale<br> Der Bieter hat mit seinem Angebot  
 geeignete Unterlagen vorzulegen, die die<br>  
 Einhaltung der Mindestvorgaben der Leistungsbeschreibung  
 bezogen  
 auf die<br> Leistungsmerkmale der  
 angebotenen Geräte/des angebotenen Gerätes belegen.<br> Dabei

ist sowohl das angebotene Fabrikat,  
Modell/Typ sowie alle zum Beleg<br>erforderlichen technischen  
Datenblätter mit Bezug zu den  
Mindestanforderungen<br>anzugeben. Hinsichtlich der  
Ausgestaltung der Unterlagen wird ausdrücklich auf  
Ziff.<br>1 dieser BHW verwiesen.<br>Enthält die  
Leistungsbeschreibung bei einer Teil-/Leistung eine  
Produktangabe mit<br>Zusatz oder gleichwertig, wird vom  
Bieter dazu eine Produktangabe verlangt  
(siehe<br>Preisblatt). Es ist das Fabrikat (insbesondere  
Herstellerangabe und genaue<br>Typenbezeichnung) auch dann anzugeben, wenn der Bieter das  
vorgegebene<br>Fabrikat anbieten will. Dies  
kann nur unterbleiben, wenn er im Angebotsschreiben<br>explizit  
erklärt, dass er das in der  
Leistungsbeschreibung benannte Produkt anbietet.<br></div>

</div>

</div>

<div class="section">

<div class="chapter">

<h4>

<div class="chapter-number">12.</div>

<div class="chapter-headline">Zuschlagskriterien</div>

</h4>

<div class="chapter-content">Alleiniges Zuschlagskriterium ist der  
Preis.<br></div>

</div>

</div>

<div class="section">

<div class="chapter">

<h4>

<div class="chapter-number">14.</div>

<div class="chapter-headline">Sonstige Angaben</div>

</h4>

<div class="chapter-content">2. Sonstige vorzulegende Nachweise,  
Unterlagen und Angaben<br>Des Weiteren  
sind die nachfolgend genannten Unterlagen und Angaben  
vorzulegen.<br>Der Bieter muss diese mit der  
Angebotsabgabe vorlegen.<br>2.1 Formular Checkliste und  
Erklärung des Bieters zu den geforderten<br>Nachweisen und Angaben<br>Den Vergabeunterlagen liegt ein  
Formular Checkliste und Erklärung des

```

        Bieters zu<br> den geforderten Nachweisen und Angaben an.<br>
        >
        Diese Tabelle ist vollstndig auszufllen
        und dem Angebot beizufgen.<br> 2.2 Eigenerklärung zur
        Beachtung
        der Verordnung PR 30/53<br> Der den
        Vergabeunterlagen beigefgte Vordruck Eigenerklärung zur
        Beachtung der<br> Verordnung PR 30/53 ist zu
        unterschreiben und dem Angebot beizufgen.<br></div>
    </div>
</div>
</div>
</div>
<div class="row_button-group">
    <div class="col-xs-12_col-sm-12_col-md-12"><a class="btn btn-
        primary"
                                                    id="id10f"
                                                    href="/tenderdocuments.
                                                    html?id=710053">
                                                    Ausschreibungsunterlagen
                                                    </a>
                                                    </div>
                                                    </div>
                                                    <hr>
                                                    <div class="link-container_text-right"><a rel="nofollow" target="_blank"
                                                    class="icon"
                                                    href="/Bekanntmachung.pdf?id
                                                    =710053"
                                                    title="PDF_der_
                                                    Bekanntmachung"><i
                                                    class="fa fa-file-pdf-o"></i> PDF</a><a rel="nofollow" target="_blank
                                                    "
                                                    class="icon"
                                                    href="/download/
                                                    Bekanntmachung.xml?id
                                                    =710053"
                                                    title="XML_der_
                                                    Bekanntmachung"><i
                                                    class="fa fa-file-text-o"></i> XML</a>
                                                    </div>
                                                    <hr>
                                                    <div class="row">
                                                    <div class="col-sm-6_gui-id"><label>XVergabe GUID:</label><span

```

```
class="form-control-static">85b4e48c-156a-4b4f-bc76-050eeaecc04d<
  /span>
</div>
<div class="col-sm-6_link-procedure"><label>Link zu dieser
  Bekanntmachung:</label><a
  href="./tenderdetails.html?id=710053">https://www.evergabe-online.
  de/tenderdetails.html?id=710053</a>
</div>
</div>
</div>
</main>
</div>
```

Quellcode 2: Extrahierte HTML einer Ausschreibung von dem Vergabeportal der Deutschen Bahn AG

```
<mat-card-content _ngcontent-ng-c3147740996=""
  class="mat-mdc-card-content_flex-row_content-details">
<div _ngcontent-ng-c3147740996=""
  class="flex-column_flex-100_gap-05-percent_flex-align-space-evenly">
  <div _ngcontent-ng-c3147740996="" class="flex-row">
    <div _ngcontent-ng-c3147740996="" class="flex-div"><h4
      _ngcontent-ng-c3147740996="">
      Auftraggeber</h4></div>
  </div>
  <mat-divider _ngcontent-ng-c3147740996="" role="separator"
    class="mat-divider_divider_mat-divider-horizontal"
    aria-orientation="horizontal"></mat-divider>
  <div _ngcontent-ng-c3147740996="" class="flex-row">
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-40">
      Auftraggebertyp
    </div>
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-60">
      Sektorenauftraggeber
    </div>
  </div>
  <div _ngcontent-ng-c3147740996="" class="flex-row">
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-40">
      Auftraggeber
    </div>
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-60">
      DB
      InfraGO AG
      Geschäftsbereich Fahrweg (Bukr 16)
    </div>
  </div>
  <div _ngcontent-ng-c3147740996="" class="flex-row">
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-40">
      Adresse
    </div>
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-60">
      <div _ngcontent-ng-c3147740996="" class="flex-row">Adam-Riese-
        Strae
        11-13
      </div>
    </div>
  </div>
  <div _ngcontent-ng-c3147740996="" class="flex-row">60327&nbsp;
```

```

        Frankfurt
        Main
    </div>
</div>
</div>
<div _ngcontent-ng-c3147740996="" class="flex-row">
    <div _ngcontent-ng-c3147740996="" class="flex-div"><h4
        _ngcontent-ng-c3147740996="">
        Verfahren</h4></div>
</div>
<mat-divider _ngcontent-ng-c3147740996="" role="separator"
    class="mat-divider_divider_mat-divider-horizontal"
    aria-orientation="horizontal"></mat-divider>
<div _ngcontent-ng-c3147740996="" class="flex-row">
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-40">
        Projektnummer
    </div>
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-60">
        24FEI76033
    </div>
</div>
<div _ngcontent-ng-c3147740996="" class="flex-row">
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-40">
        Titel
    </div>
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-60">
        Errichtung von 5
        Berhrungschutzanlagen Bad Neuenahr Ahrweiler, Strecke 3000, km
        12,363–km 14,342
    </div>
</div><!------>
<div _ngcontent-ng-c3147740996="" class="flex-row">
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-40">
        Vergabeordnung
    </div>
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-60">
        VOB/A
    </div>
</div>
<div _ngcontent-ng-c3147740996="" class="flex-row">
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-40">
        Leistungsart
    </div>

```

```

    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-60">
      Bauleistung
    </div>
  </div>
  <div _ngcontent-ng-c3147740996="" class="flex-row">
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-40">
      Vergabeart
    </div>
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-60">
      Offenes Verfahren
    </div>
  </div>
  <div _ngcontent-ng-c3147740996="" class="flex-row_ng-star-inserted">
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-40">
      Vertragsart
    </div>
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-60">
      Einzelbestellung
    </div>
  </div>
  <div _ngcontent-ng-c3147740996="" class="flex-row_ng-star-inserted">
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-40">
      Ausführungsort
    </div>
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-60">
      53474
      Bad
      Neuenahr-Ahrweiler
    </div>
  </div>
  <div _ngcontent-ng-c3147740996="" class="flex-row_ng-star-inserted">
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-40">
      Nebenangebote
    </div>
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-60">
      Nebenangebote sind nur
      in Verbindung mit einem Hauptangebot zugelassen
    </div>
  </div><!------>
  <div _ngcontent-ng-c3147740996="" class="flex-row_ng-star-inserted">
    <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-40">
      Mehrere Hauptangebote
      zulässig
    </div>
  </div>

```

```

</div>
<div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-60">Ja
</div>
</div>
<div _ngcontent-ng-c3147740996="" class="flex-row_ng-star-inserted">
  <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-40">
    Unterteilung in Lose
  </div>
  <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-60">
    Nein
  </div>
</div><!--><!-->
<div _ngcontent-ng-c3147740996="" class="flex-row">
  <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-40">
    Verwendung
    elektronischer Mittel
  </div>
  <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-60">
    Die
    Einreichung der
    Angebote/Teilnahmeantrge darf nur elektronisch erfolgen
  </div>
</div>
<div _ngcontent-ng-c3147740996="" class="flex-row">
  <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-40">
    Klassifizierung
  </div>
  <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-60">
    21120999 – Brcken u
    Massive Sttzbauwerke, geringf Leistungen ohne P<br>45221100–3 –
    Bauarbeiten fr
    Brcken<br></div>
</div>
<div _ngcontent-ng-c3147740996="" class="flex-row">
  <div _ngcontent-ng-c3147740996="" class="flex-div"><h4
    _ngcontent-ng-c3147740996="">Fristen
    und Termine</h4></div>
</div>
<mat-divider _ngcontent-ng-c3147740996="" role="separator"
  class="mat-divider_divider_mat-divider-horizontal"
  aria-orientation="horizontal"></mat-divider>
<div _ngcontent-ng-c3147740996="" class="flex-row">
  <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-40">

```

Bekanntmachung

</div>

<div \_ngcontent-ng-c3147740996="" class="flex-column\_flex-basis-60">

26.08.2024, 12:56:09

</div>

</div>

<div \_ngcontent-ng-c3147740996="" class="ng-star-inserted">

<div \_ngcontent-ng-c3147740996="" class="flex-row\_ng-star-inserted">

<div \_ngcontent-ng-c3147740996="" class="flex-column\_flex-basis-40">

Einreichungsfrist

</div>

<div \_ngcontent-ng-c3147740996="" class="flex-column\_flex-basis-60">

20

Sept. 2024, 09:00

</div>

</div><!------><!------>

<div \_ngcontent-ng-c3147740996="" class="flex-row\_ng-star-inserted">

<div \_ngcontent-ng-c3147740996="" class="flex-column\_flex-basis-40">

Frist Bieterfragen

</div>

<div \_ngcontent-ng-c3147740996="" class="flex-column\_flex-basis-60">

12

Sept. 2024, 12:00

</div>

</div><!------>

<div \_ngcontent-ng-c3147740996="" class="flex-row\_ng-star-inserted">

<div \_ngcontent-ng-c3147740996="" class="flex-column\_flex-basis-40">

Bindefrist

</div>

<div \_ngcontent-ng-c3147740996="" class="flex-column\_flex-basis-60">

18.10.2024

</div>

</div><!------>

<div \_ngcontent-ng-c3147740996="" class="flex-row\_ng-star-inserted">

<div \_ngcontent-ng-c3147740996="" class="flex-column\_flex-basis-40">

Auftragsdauer von

</div>

<div \_ngcontent-ng-c3147740996="" class="flex-column\_flex-basis-60">

02.01.2025

</div>

</div><!------>

<div \_ngcontent-ng-c3147740996="" class="flex-row\_ng-star-inserted">

<div \_ngcontent-ng-c3147740996="" class="flex-column\_flex-basis-40">

```

    Auftragsdauer bis
  </div>
  <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-60">
    31.12.2025
  </div>
  </div><!--><!--></div><!--><!-->
<div _ngcontent-ng-c3147740996="" class="flex-row">
  <div _ngcontent-ng-c3147740996="" class="flex-div"><h4
    _ngcontent-ng-c3147740996="">
    Verfahrensbeschreibung</h4></div>
</div>
<mat-divider _ngcontent-ng-c3147740996="" role="separator"
  class="mat-divider_divider_mat-divider-horizontal"
  aria-orientation="horizontal"></mat-divider>
<div _ngcontent-ng-c3147740996="" class="flex-row">
  <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-40">
    Beschreibung
  </div>
  <div _ngcontent-ng-c3147740996="" class="flex-column_flex-basis-60">
    Elektrifizierung der
    Strecke 3000 – Ahrstrecke, Errichtung von 5 Berhrungschutzanlagen an
    Fugnger- und
    Straenbrcken der Strecke Bad Neuenahr Ahrweiler, Strecke 3000, km
    12,363–km 14,342
  </div>
</div>
</div>
</div>
</mat-card-content>

```