Comparative Study of Named Entity Recognition Models

Anonymous EMNLP submission

Abstract

Named Entity Recognition (NER) is a fundamental and non-trivial task in natural language processing, that is crucial for various downstream applications. This paper presents a comprehensive comparative study of NER performance across a spectrum of state-of-the-art models, with a particular focus on the adaptation and fine-tuning of Question Answering (QA) models, such as BERT and RoBERTa, alongside prominent text generation models, including Llama2 (Touvron et al., 2023), Mistral (Jiang et al., 2023), and ChatGPT3.5-Turbo. In this study, we explore the efficacy of QA models when repurposed and adapted to NER tasks and additionally, we examine the zero-shot capabilities of Large Language Models, utilizing them without task-specific fine-tuning to assess their innate ability to recognize named entities. Through extensive experimentation on the benchmark dataset BUSTER, we analyze and compare the precision, recall, and F1 scores of each model variant across various NER categories. Furthermore, we investigate the robustness of these models under different training regimes and evaluation metrics.

1 Introduction

The natural language processing domain is rapidly evolving with the recent developments of Large Language Models (LLMs). The advent of LLMs, such as the Generative Pre-trained Transformers (GPT) series (Radford et al., 2018, 2019; Solaiman et al., 2019; OpenAI et al., 2023), has ushered in a new era of NLP by leveraging the power of deep learning and vast amounts of unlabeled text data. One of the most striking aspects of LLMs is their ability to perform at State-of-the-Art (SOTA) levels across a wide range of tasks without task-specific fine-tuning. Unlike standard baselines, like bidirectional encoder transformers (BERT) (Devlin et al., 2019), which require task-specific modifications and fine-tuning, LLMs are pre-trained on large corpora of text data using unsupervised learning techniques. However, it is essential to acknowledge that, while LLMs showcase remarkable capabilities, they also come with their own set of drawbacks. One of the most notable drawbacks is their complexity. LLMs, especially the larger variants like GPT-3, consist of millions or even billions of parameters, making them computationally intensive to train and deploy. This complexity translates into significant time and resource consumption, both during the training phase and during inference, where the computational requirements can be prohibitive for cost effective and/or real-time applications. However that's not all, we have to mention that LLMs often fall short due to their generalist nature, struggling to grasp the nuanced understanding and specialized knowledge required for tasks in some domains like scientific and technical texts. In this study, we conduct a comparative analysis of various methodologies employing diverse model architectures for Named Entity Recognition (NER). NER involves identifying and classifying entities within text into predefined categories such as persons, organizations, locations, dates, and more. In our case it is restricted on the main actors involved in a business transaction. While on the surface, it may seems similar to other NLP tasks, like sentiment analysis or text classification, it poses unique challenges that set it apart. Firstly, NER requires a deep understanding of language syntax, semantics, and context. This involves parsing and comprehending complex linguistic structures, including ambiguous references, colloquialisms, and variations in naming conventions. For instance, in business transaction data, entities such as company names and financial terms can exhibit significant variability and ambiguity. Terms like "revenue" or "earnings" may have different interpretations based on the context and industry. Exploiting the same reasoning applied for the aforementioned LLMs, we study the behaviour of LLMs like BERT (De-

vlin et al., 2019), RoBERTa (Liu et al., 2019) and Longformers (Beltagy et al., 2020), fine tuned on a QA based Entity Extraction task. As reported in (Liu et al., 2022) the Entity Extraction task is naturally suited to be formulated as a question to a QA model. Finally, the objective of the study is to comprehensively assess various methodologies, quantifying the efficacy of each of these while delineating their relative differences. This evaluation facilitates informed decision-making regarding model selection, incorporating considerations such as deployment costs and resource consumption.

The paper is organized as follow. In the next section, we describe some common approaches to the NER task. In section 3, we introduce and describe the proposed QA approach to the NER task, while in the subsequent section 4, we describe how the Generative models can be exploited to face the NER task. In section 5, we describe the experimental setup with particular focus on the dataset, the different tagging schemes and the evaluation metrics used in the experiment. Then, in section 6, we report the results of the experimental evaluation, also describing the emerging scenarios. Finally in section 7, we draw the conclusions.

2 Related works

Named Entity Recognition (NER) is a crucial task in Natural Language Processing (NLP) that involves identifying and classifying entities mentioned in a text into predefined categories such as names of persons, organizations, locations, dates, and other proper nouns. This task is essential for extracting structured information from unstructured text, enabling various applications such as information retrieval, question answering, and knowledge graph construction. In many works NER tasks are formulated as a sequence tagging or token classification problem (Sang and Meulder, 2003; Devlin et al., 2019; Yang and Katiyar, 2020). In sequence tagging, each token in the input sequence is assigned a label indicating its entity type. The machine learning model predicts labels for each token independently, resulting in a sequence of labels corresponding to the input tokens. Recent works, have explored prompt-based learning in the NER field. This approach has gained a huge popularity in the NLP community in many downstream applications (Liu et al., 2021) owing to its utility in tackling data-intensive and time-consuming tasks such as NER. Various methods have been proposed, such as (Cui et al., 2021) where a template based method in a seq2seq framework is presented, (Chen et al., 2022) where the task is presented as a generation problem. Meanwhile, other works related to NER have explored the use of Question Answering (QA) models (Li et al., 2020; Liu et al., 2022), which traditionally focus on providing answers to questions posed in natural language. These models, particularly those based on transformer architectures like BERT (Devlin et al., 2019), have shown great results in NER tasks. By framing NER as a Machine Reading Comprehension (MCR) task, where the model is asked to find entities that answer specific questions about the text, researchers have been able to leverage the powerful contextual understanding capabilities of QA models.

3 QA-based NER

Question Answering (QA) models can be a flexible tool to tackle the NER task. QA models are designed to comprehend and answer questions based on a given context, requiring a deep understanding of the text and its underlying semantics. This capability inherently involves reading comprehension and context awareness, making QA models well-suited for tasks requiring nuanced understanding of textual data. Unlike traditional sequence tagging approaches, QA models are trained to identify and extract relevant information from passages of text to generate accurate responses to questions. This necessitates an understanding of the context in which entities are mentioned, enabling the model to discern entity boundaries and extract spans accordingly. Furthermore, the inherent similarity between the entity extraction and the question answering tasks lends itself well to the adaptation of QA models for entity extraction. Adapting QA models for multi-span extraction presents distinct challenges. Unlike other proposed approaches (Liu et al., 2022; Li et al., 2020), which focus on generating a single, concise answer, multi-span extraction requires the model to identify and extract multiple spans, potentially with overlapping or nested entities. Although some methods achieve multi-span extraction by submitting the same question multiple times, this approach is less efficient and the exact number of submissions is not always clear. Therefore, modifications to the model architecture and training objectives are necessary to accommodate the accurate extraction of spans from a given context. To adapt the question answering model for token

classification, we integrated a linear layer on top of the network similarly of (Segal et al., 2020). This adjustment allows the model to calculate the probability of each token being part of an entity. The dataset was adapted for our setup by dividing the original samples according to each entity label. Consequently, each document is fed into the network multiple times, corresponding to the number of entity labels it contains. This approach serves two primary purposes: it simplifies the learning by isolating a single class at a time, such that the model's task is easier as it focuses on extracting one entity type per iteration. This reduction in complexity facilitates the learning phase. Furthermore presenting the same document multiple times, each with a different question to focus on, encourages the model to generalize better on the data. For the questions, we opted for simplicity over extensive prompt engineering, following the approach reported in (Liu et al., 2022). We formulated questions using "Who is the [E]?" where [E] stands for the entity type.

4 GenAI-based NER

Since their introduction by (Vaswani et al., 2023), transformer-based models have marked a significant turning point in NLP research. The advent of generative models such as Generative Pre-Trained Transformers (GPT), developed by OpenAI, has further expanded the use of artificial intelligence into everyday life, even for individuals without specific expertise in computer science.

Transformers like BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2019) leverage the transformer architecture for bidirectional training and are pre-trained on two main tasks: masked language modeling and next sentence prediction. These transformers excel in tasks requiring deep comprehension, such as reading comprehension and text classification, due to their ability to exploit the context from both directions.

On the other hand, GPT and other large language models (LLMs) utilize autoregressive training, focusing on next-word prediction to generate coherent text. However, their utility extends beyond mere text generation, making them versatile tools in various domains. Generative models undergo extensive pre-training on diverse and large-scale datasets, which endows them with a wealth of linguistic knowledge and world information. This pre-training enables them to learn rich representations of language, including syntax, semantics, and factual knowledge, facilitating transfer learning when applied to other tasks. Consequently, these models perform well in question answering, language translation, text summarization, and creative writing.

Generative models have also demonstrated impressive few-shot and zero-shot learning capabilities, as highlighted by (Brown et al., 2020). These capabilities allow them to perform tasks with minimal or no task-specific examples, making them highly adaptable to new tasks with little (or no) training data. For this reason we aim at testing them on a hard task like Named Entity Recognition to evaluate their performance in this domain. We mentioned in Section 3 about the similarity between question answering and entity extraction and we pointed out that generative models are suited for this task. Hence, we explored the use of these models in a zero-shot learning context for NER.

5 Experiments

The described approaches have been compared on a specific NER task involving texts about business transactions. In the following we describe the experimental setup.

5.1 Dataset

In our experimentation, we utilized the BUSTER dataset (Zugarini et al., 2023). BUSTER is a business-oriented dataset comprising approximately 10,000 business transaction documents sourced from EDGAR company acquisition reports. The dataset is partitioned into two subsets: GOLD and SILVER corpus. We exclusively employed the GOLD subset due to its human-annotated nature, while the SILVER corpus was annotated using a trained Large Language Model (LLM) as detailed in (Zugarini et al., 2023). The GOLD set consists of 3,779 documents and it is split into five folds.

5.2 QA-based NER experiments setup

To ensure a fair comparison, we employed some baseline networks used in (Zugarini et al., 2023), with the difference that our models are meant for QA and were fine-tuned on the Stanford Question Answering Dataset 2.0 (SQuAD 2.0) (Rajpurkar et al., 2016). Fine-tuning on SQuAD 2.0 was chosen to leverage its comprehensive coverage of question-answering nuances. It incorporates the 100,000 questions from SQuAD 1.1 alongside over

50,000 unanswerable questions, deliberately designed to mimic the occurrence of answerable questions. To perform well on SQuAD 2.0, systems must accurately answer questions when possible and identify when a paragraph does not contain an answer, abstaining from providing a response. This approach is particularly useful for entity extraction in lengthy texts, such as ours, where the entities of interest are relatively sparse. By training on a dataset that includes unanswerable questions, systems improve their ability to discern relevant information from irrelevant text, enhancing their precision in identifying the correct entities. This reduces false positives and increases the efficiency of extracting meaningful data from large volumes of text.

Tagging scheme In the experimentation, we investigated different tagging schemes based on the BIO (Begin-Inside-Other) format to identify the most effective approach for this task.

1. **no-Entity BIO Scheme (nE-BIO)**: We tested a plain BIO scheme without distinguishing between different entity types. In particular, this scheme assumes that the entity type is provided as additional input information to the model. This approach simplifies the model, making it more efficient and potentially improving generalization by reducing the number of classes.

2. **no-Entity BI Scheme (nE-BI)**: This approach follows the idea of the previous approach, with the addition that only the B and I tags are used. The O tag is inferred from the absence of both the B tag and the I tag on a token. In particular, a sigmoid function is applied to the final layer and a token is not considered part of an entity (as for label O in other methods) if the probabilities for both 'B' and 'I' are below a certain threshold. This simplifies the classification process but requires careful threshold tuning to ensure accurate token classification.

3. **per-Entity BI Scheme (pE-BI)**: This method uses specific 'B' (begin) and 'I' (inside) labels for each entity type (e.g., B-PEOPLE, I-PEOPLE). Thus, considering the 6 entity types in the BUSTER dataset, this method results in 12 different predictable classes per token, providing detailed entity information but increasing the complexity of the model. The application of a sigmoid function to the last layer is valid also for this scheme.

4. **no-Entity I-Only Scheme (nE-I)**: This scheme further reduces the number of predictable labels compared to the previous case by eliminating the "B" label and limiting the model output for each token exclusively to a binary choice PART/NO-PART of an entity. Also in this case, a sigmoid function is applied on the final layer to estimate this probability. This scheme, however, has a significant limitation: without the "B" label, the model cannot explicitly mark the start of an entity, potentially leading to the merging of contiguous entities. Such cases, however, are very rare in our experiments.

Non informative samples During the training phase one significant challenge was handling the limited input token capacity of models. While Longformers can have a maximum sequence length of 4096 input tokens, models like BERT and RoBERTa are limited to 512 tokens.

As noted in (Zugarini et al., 2023), most documents in the BUSTER dataset exceed 500 words, surpassing the maximum sequence length of such models. To accommodate these lengths, we divided the documents into overlapping segments, ensuring that entity labels were not inadvertently truncated.

Document segmentation, combined with multiple document submissions, substantially increased the number of samples to be fed to the networks. However, it also resulted in a significant number of non-informative (empty) samples, since entities are sparse, which could impact learning. With the definition "non informative samples" we intend those subdocuments where there are no labels within the text. To address this problem we conducted experiments (see Table 2) comparing the performances with varying numbers of non-informative examples. The number of such samples can affect the performances and it may slow down the training phase.

5.3 GenAI-based NER experiments setup

We conducted experiments in a zero-shot learning context, particularly because our documents are quite long, making true few-shot learning for each class impractical. The main challenge lies in handling long input lengths. While some models can accept very long texts, processing these inputs

demands substantial computational resources and strains the model's ability to maintain coherence and focus.

As input sequences get longer, the model's capacity to focus on relevant parts of the text and maintain coherent context diminishes. This often results in outputs that are less coherent, more repetitive, or diverge from the intended context.

When working with generative models, formulating precise and effective prompts is crucial, as different prompts can significantly impact the quality of the generated responses. We experimented with various prompts across different models, observing that some prompts yield satisfactory results with some models but perform poorly with others. This variability is attributed to each model's specific characteristics, such as its complexity, number of parameters, and training data. Generally, larger models like GPT-3 exhibit greater robustness and generate more coherent answers, being less affected by prompt variations.

To ensure accuracy and consistency in the generated responses, we instructed each model to follow a precise schema and return answers in JSON format. This format is chosen for its simplicity, widespread use, and effectiveness in minimizing the need for extensive post-processing. Non-standard formats, by contrast, are difficult and time-consuming to clean and organize.

To automate the validation process, we developed a script that checks if the model's response adheres to the JSON format. If the response deviates from this format, the script automatically sends a new request to the model. Additionally, to mitigate the issue of hallucinations—where models generate spurious or incorrect outputs—we implemented a check on the labels in the predictions. If the model introduces unrecognized classes, the prediction is discarded, and a new request is issued. However, if the model omits certain classes, we interpret them as not present in the document and accept the prediction as valid. This approach ensures that the generated responses are both accurate and in a standardized format, facilitating easier integration and analysis.

5.4 Metrics

To evaluate the performance of QA-based NER models, we employed precision, recall, and the F1 score, either in a micro- and macro- averaging scheme. These metrics were applied at the entity level, meaning the F1 score considered the entire span of the entities rather than individual tokens. This approach provides a more holistic evaluation of the model's ability to correctly identify complete entities compared to token-level metrics.

For evaluating the generative models' predictions, we used the same metrics. However, comparing generative based NER and Question-Answering NER models is not straightforward since they fundamentally face different tasks. The former models are trained to execute token classification tasks, aiming to find multiple instances of the same entity within a document. In contrast, generative models generate responses to generic questions, making their task inherently different and often less complex, since they are not required to report each different occurrence of a given entity.

Given that Question-Answering NER models are trained to identify multiple instances of the same entity, their task is more challenging. To enable a fair comparison between Question-Answering and generative based models, we adjusted some constraints for the first ones. Specifically, we clustered predictions for each class using Jaccard Index distance with a threshold K.

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{1}$$

Where:

- $|A \cap B|$ is the number of common characters between the two strings.
- $|A \cup B|$ is the total number of unique characters present in both strings.

If the Jaccard Index between two strings is greater than $K = 0.5$ they are considered to represent the same entity.

In the evaluation, we employed the same "relaxed" criterion used to cluster predictions also to match ground truths and predictions. In the relaxed criterion two strings are considered equal if their Jaccard Index is greater than the given threshold. This relaxation allows for minor variations in the strings while still considering them as correct matches.

6 Experimental Results

In this section, we report the results of the different experiments performed.

The impact of the Tagging scheme Firstly, we investigate the impact of using the different Tagging schemes in a QA-based NER approach. The desired goal was to identify the optimal tagging approach for robust entity recognition across different contexts. In particular, each scheme was evaluated using BERT as the underlying model and the results are reported in Table 1. Results show a significant performance discrepancy between the BIO scheme and other tagging schemes.

Tagging scheme	μ-F1	M-F1
pE-BI	72.6 ± 0.8	75.6 ± 1.7
nE-BIO	17.0 ± 3.8	15.0 ± 3.8
nE-BI	73.2 ± 0.7	76.8 ± 0.7
nE-I	73.8 ± 0.4	76.6 ± 1.0

Table 1: F1 scores for different tagging schemes obtained with BERT

The nE-BIO scheme proved ineffective for token classification in our context, primarily due to the strong bias towards the 'O' (Other) label. This bias arises for two main reasons. Firstly, we are targeting a precise, small set of entities within lengthy texts, which on average exceed 500 words. Secondly, in the BUSTER dataset (Zugarini et al., 2023), entities were labeled only in paragraphs where their roles were clearly specified, further reducing their frequency within the documents. Regarding other tagging schemes, our findings indicate that the pE-BI scheme performs slightly worse than the nE-BI and nE-I schemes. This observation suggests that adding more granularity in the tagging does not necessarily translate to better performance. A possible explanation to this behaviour can be found in the models architecture and training procedures. LLMs benefit from extensive pretraining on diverse datasets, which allows them to generalize well across different tasks without needing highly specialized tagging schemes. Their attention mechanisms enable them to effectively manage the relationships between tokens, reducing the reliance on specific tagging details that smaller, less sophisticated models might require.

The impact of non-informative samples The results in Table 2 indicate that increasing the number of non-informative samples generally improved the performance of both BERT and RoBERTa models. Notably, RoBERTa showed significant improvement when the sample size increased from 6000 non-informative examples but no further improvements have been made with more samples. A similar behaviour is observed with BERT, it demonstrated more stable performance across different sample sizes up to 8000 samples, but as RoBERTa performances start to decrease as we further increase the number of non informative samples. Such behaviour suggests that while non-informative samples can enhance training, the optimal number can slightly vary between different models and has to be carefully tuned in order to optimize performances.

Comparison of QA-based models. In this experiment, we compared the performance of several common LLMs when used in a QA-based NER task. Performances were evaluated in a k-fold cross-validation setup, with $k = 5$, according to the original dataset splits. Table 3 reports the obtained results.

QA Model	μ-F1	M-F1
BERT-Large	73.6 ± 0.5	77.2 ± 0.7
RoBERTa-Large	74.4 ± 0.5	78.6 ± 0.8
Longformer	74.2 ± 0.7	78.2 ± 1.0

Table 3: QA-based NER approaches results on BUSTER dataset. The evaluation has been done considering the per-entity F1 score.

BERT and RoBERTa performances were obtained by tuning the number of non-informative samples as described in the paragraph above. RoBERTa was found to be the best model, followed closely by Longformer, whereas BERT showed slightly lower performances than the other two.

QA-based vs GenAI-based approaches to NER Table 4 provides an overview of the performance obtained using the two different approaches. As detailed in Section 5, Question-Answering models applied to NER are evaluated on their ability to predict correct entities within an entire document, rather than on the traditional token classification task for which they were originally designed.

# non-informative samples	BERT		RoBERTa	
	μ-F1	M-F1	μ-F1	M-F1
2000	72.8 ± 0.7	75.8 ± 0.4	73.8 ± 1.0	78.4 ± 0.5
4000	73.0 ± 0.9	76.6 ± 1.0	73.8 ± 0.4	78.2 ± 1.2
6000	73.2 ± 0.7	76.8 ± 0.7	74.4 ± 0.5	78.6 ± 0.8
8000	73.6 ± 0.5	77.2 ± 0.7	73.8 ± 0.7	77.4 ± 1.2
All	73.2 ± 0.7	76.6 ± 0.5	74.4 ± 0.5	78.2 ± 0.7

Table 2: BERT and RoBERTa f1 scores with varying number of non-informative samples.

Model	μ-F1	M-F1
ChatGPT3.5-turbo	64.4 ± 0.6	56.0 ± 0.7
Llama-2-7b-chat-hf	50.8 ± 1.1	40.0 ± 1.0
Llama-2-13b-chat-hf	58.8 ± 0.0	45.1 ± 0.0
Mistral-7b	52.2 ± 1.3	38.3 ± 1.2
BERT-Large	79.6 ± 0.5	79.0 ± 0.4
RoBERTa-Large	80.2 ± 1.1	79.6 ± 0.9
Longformers	79.8 ± 0.7	79.2 ± 1.0

Table 4: Full comparison of QA-based and GenAI-based NER approaches on BUSTER dataset. In this case, the evaluation of QA models have been performed using entity clusters as specified in 5.4.

A comparison of Table 3 and Table 4 reveals that the token classification task is inherently more challenging than the entity extraction performed by generative models. Generative models in a zero-shot learning context do not yet match the performance of smaller models that have been specifically trained on targeted datasets, such as the one used in our study.

This suggests that while generative models offer versatility and the potential for broad application without extensive retraining, specialized token classification models currently provide superior accuracy for NER tasks when trained on domain-specific data. The gap highlights the need for continued development and fine-tuning of generative models to achieve comparable results in specific tasks.

7 Conclusions

In this study, we examined and compared two distinct machine learning approaches for Named Entity Recognition (NER). Our objective was to evaluate their strengths and weaknesses to determine their relative efficacy in NER tasks. Firstly, we explored the use of QA models for NER, detailing the necessary architectural modifications and the evaluation of various tagging schemes. Our analysis revealed that the traditional BIO tagging scheme was suboptimal in our context due to its bias towards the 'O' label, especially in lengthy texts with sparse entities. We also addressed the impact of non-informative offsets generated by models with limited input capacity, highlighting the importance of carefully managing these offsets to avoid performance degradation.

Next, we investigated the performance of Generative models in a zero-shot learning setting, focusing on the choice of prompts and the differences across various models. Despite their versatility and ability to handle zero-shot learning, generative models did not achieve the same level of accuracy as specialized token classification models trained on domain-specific data.

In summary, our comparison indicates that while generative models offer significant flexibility and broad applicability, they currently fall short of the precision provided by models specifically trained for NER tasks, expecially when dealing with specific domains. This underscores the need for continued refinement of generative models to close the performance gap in specialized applications.

Limitations

Despite the promising results of proposed methods some limitations must be acknowledged. Due to limited time our experiments are done only on BUSTER dataset (Zugarini et al., 2023). While in our opinion it is a good benchmark for NER task in a real case scenario it is also a domain specific dataset focused on financial data. Future developments could take into account experimentation on other popular datasets to better compare our approach with others. We have also to consider that training and fine-tuning large models such as BERT, RoBERTa, and Longformer require substan-

tial hardware resources. The computational cost associated with extensive hyper-parameter tuning, k-fold cross-validation, and handling large chunks of data could limit accessibility of our approach. However, it is important to note that once a QA model adapted to NER task is trained, it can leverage prompt learning and few-shot techniques with minimal question tuning. This adaptability is due to the architecture's independence from the specific entity classes being identified.

References

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

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. 2020. Language models are few-shot learners.

Xiang Chen, Lei Li, Shumin Deng, Chuanqi Tan, Changliang Xu, Fei Huang, Luo Si, Huajun Chen, and Ningyu Zhang. 2022. Lightner: A lightweight tuning paradigm for low-resource ner via pluggable prompting.

Leyang Cui, Yu Wu, Jian Liu, Sen Yang, and Yue Zhang. 2021. Template-based named entity recognition using BART. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1835–1845, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b.

Xiaoya Li, Jingrong Feng, Yuxian Meng, Qinghong Han, Fei Wu, and Jiwei Li. 2020. A unified MRC framework for named entity recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5849–5859, Online. Association for Computational Linguistics.

Andy T. Liu, Wei Xiao, Henghui Zhu, Dejiao Zhang, Shang-Wen Li, and Andrew Arnold. 2022. Qaner: Prompting question answering models for few-shot named entity recognition.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

OpenAI et al. 2023. Gpt-4 technical report. Technical report.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. Technical report, OpenAI Blog.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. Technical report, OpenAI.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition.

Elad Segal, Avia Efrat, Mor Shoham, Amir Globerson, and Jonathan Berant. 2020. A simple and effective model for answering multi-span questions.

Irene Solaiman, Miles Brundage, Jack Clark, Amanda Askell, Ariel Herbert-Voss, Jeff Wu, Alec Radford, and Jasmine Wang. 2019. Release strategies and the social impacts of language models. *CoRR*, abs/1908.09203.

Hugo Touvron et al. 2023. Llama 2: Open foundation and fine-tuned chat models. Technical report.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention is all you need.

Yi Yang and Arzoo Katiyar. 2020. Simple and effective few-shot named entity recognition with structured nearest neighbor learning. In *Proceedings of the*

2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 6365–6375, Online. Association for Computational Linguistics.

Andrea Zugarini, Andrew Zamai, Marco Ernandes, and Leonardo Rigutini. 2023. Buster: a "business transaction entity recognition" dataset. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*. Association for Computational Linguistics.