2024 11th International Conference on Machine Intelligence Theory and Applications (MiTA) | 979-8-3503-8567-0/24/$31.00 ©2024 IEEE | DOI: 10.1109/MiTA60795.2024.10751683

Decoding Domain-Specific NER: A Performance Evaluation of ChatGPT, Bi-LSTM, and BERT

Khushnood Adil Rafique, Maulik Pansuriya, Frank Wawrzik, Christoph Grimm
University of Kaiserslautern-Landau (RPTU), Chair of Cyber-Physical Systems, Kaiserslautern, Germany
[khushnood.rafique|wawrzik|grimm]@cs.uni-kl.de, pansuriy@rhrk.uni-kl.de

***Abstract*—Named Entity Recognition (NER) is a vital method in natural language processing (NLP), extracting information by identifying and categorizing entities in textual data. This study delves into the dynamic landscape of NER models, considering recent advances in transfer learning with transformers like Bidirectional Encoder Representations from Transformers (BERT) and large language models such as ChatGPT. We evaluate and compare their performances on a custom domain-specific and limited corpus. Utilizing state-of-the-art techniques, we model multi-layered prompts for ChatGPT, concurrently utilizing the Bidirectional Long Short-Term Memory Networks (Bi-LSTM) model and fine-tuning a pre-trained BERT model for NER on the corpus. Our study offers crucial insights into these models' viability under constrained, domain-specific dataset conditions, given the effectiveness of transfer learning and zero-shot classification with large language models. Our experimental results highlight that Bi-LSTM excels over ChatGPT and fine-tuned BERT on the custom dataset, with the added benefit of faster training times. We emphasize the importance of considering domain and task-specific factors and corpus size when selecting a model.**

***Index Terms*—Named Entity Recognition, Transfer learning, ChatGPT, BERT, Bi-LSTM, Information Extraction, LLMs, Transformers, Natural Language Processing, Knowledge Base Construction**

I. Introduction

Named Entity Recognition (NER) stands as a crucial component across diverse domains, facilitating information extraction and representation. Its role is particularly pronounced in scenarios with high data volumes, such as news articles, scientific papers, or legal documents, enabling the identification and extraction of structured information from unstructured text. NER can also prove to be useful in knowledge base construction (KBC), enabling the extraction of specific entities from unstructured text. Entity extraction contributes to the creation of interconnected relationships or triples within the knowledge base, generating a comprehensive understanding of the domain. As demonstrated in our previous work, Wawrzik et al. (2023) [1], we showcase the application of a Bidirectional Long Short-Term Memory Networks (Bi-LSTM) [2] model to perform NER on text segments from the microelectronics domain demonstrating the practical application of NER methodologies in systems engineering and ontology learning.

However, in the contemporary scientific realm, NER methodologies have evolved, particularly with the emergence of transfer learning. State-of-the-art models like Bidirectional Encoder Representations from Transformers (BERT) [3], GPT [4], and RoBERTa [5] have reshaped its execution, overshadowing conventional approaches like Bi-LSTMs. Despite the prevailing trend favoring large language models (LLMs) and pre-trained models like BERT, even for relatively minor tasks, our study seeks to demonstrate the nuanced outcomes of this bias. The study intends to illustrate that this approach does not consistently yield optimal results. Our experimental findings highlight the advantages of Bi-LSTM over ChatGPT and fine-tuned BERT on our tailored dataset, coupled with quicker training times. This underlines the criticality of factoring in domain and task-specific elements, along with corpus size, in model selection.

A. Named Entity Recognition with ChatGPT

The advent of LLMs has notably streamlined NER, especially with the accessibility of zero-shot classification. Zero-shot classification has become more accessible with LLMs like ChatGPT [6], and the quality of their NER results is contingent on the quality of prompts provided to them. Zero-shot classification allows models to make predictions on classes absent from the training set, showcasing their ability to generalize to novel scenarios. ChatGPT, which is based on the Generative Pre-trained Transformer 3 and 4 (GPT-3 and 4) architecture [7], while powerful in understanding and generating human-like text, may not be as optimized for tasks that require fine-tuned classification capabilities, especially in a zero-shot setting, though many attempts have been made to make ChatGPT perform NER with better prompt engineering which is further discussed in section II of this paper.

B. Named Entity Recognition with Bi-LSTM & BERT

Both Bi-LSTM and BERT engage distinct methodologies. Bi-LSTM, a recurrent neural network variant, processes sequential data bidirectionally, capturing contextual information through its hidden states. It excels at recognizing patterns and dependencies in sequential input. On the other hand, BERT, which combines bidirectional transformers and transfer learning, utilizes attention mechanisms to consider the entire context of a word within a sentence simultaneously. It goes through a pre-training phase on massive datasets, learning contextualized representations of words. During the fine-tuning

phase, BERT adapts to specific tasks like NER by adjusting its parameters. The key distinction lies in how these models comprehend contextual information. Bi-LSTM processes data sequentially, making it capable of capturing sequential patterns. In contrast, BERT leverages attention mechanisms to grasp the global context, allowing it to understand relationships between words regardless of their sequential position. BERT's contextual embeddings often lead to nuanced and accurate NER results.

C. Role of Our Study

- **Performance Evaluation:** Our study systematically evaluates and objectively compares the NER performances of transfer learning models like ChatGPT and BERT, and a traditional model like Bi-LSTM, on a domain-specific and limited dataset.
- **Scientific Gap Fill:** This comprehensive three-way comparison, conducted on a level playing field with a consistent custom dataset, using more expressive and flexible metrics, addresses a gap in the scientific research community. It is crucial to highlight, through objective analysis, the distinctions among these systems and how their performances vary under less-than-ideal conditions.
- **Methodological Considerations:** Also, the methods used in the existing studies, to compare their performances did not take into account the subtleties of information extraction like partial entity or entity boundary extraction. We address such subtleties in this work.
- **Focus on Constrained Conditions:** The focus of this study lies in discerning the advantages of using one approach over another in constrained conditions.
- **Exploring Viability:** We measure the degree of superiority, if any, of zero-shot classification to approaches that require labeled training data, while also testing if traditional methods are still viable in the era of transfer learning.

II. State of the Art

This section discusses applications of NER and its implementation through various NLP models, including ChatGPT. The discussion includes state-of-the-art methodologies in prompt engineering and explores studies comparing transfer learning and traditional models.

Wawrzik et al. (2023) harnessed the capabilities of Bi-LSTM to execute entity extraction within the knowledge base construction pipeline with satisfactory NER results. On the other hand, Luoma et al. (2020) [8] and Souza et al. (2019) [9] applied BERT models to NER in languages other than English and their study consistently enhanced NER performance across languages. While Hakala et al. (2019) [10] used Spanish biomedical NER to apply a CRF-based baseline approach and multilingual BERT to achieve excellent results.

To perform NER with ChatGPT we require innovative prompt modeling strategies. Wei et al. (2023) [11] introduced novel techniques for prompt generation in ChatGPT, leading to notable improvements in NER outcomes. The study emphasizes zero-shot information extraction, a method used for its advantage of eliminating the need for training data. The study discusses relation-triple extraction, NER, and event extraction. The model adopts a two-stage prompt engineering process through dialogue with ChatGPT. The first stage involves identifying the existing types of entities, while the second stage comprises multiple question-and-answer (QA) sessions. The authors provide customizable templates applicable to various datasets. Henceforth in this paper, we simplify the reference to their work as *"**Chat2Extract**"*. In another study by Xie et al. (2023) [12], the authors adapt established reasoning methods to NER and introduce four strategies: decomposed-QA, syntactic prompting, tool augmentation, and two-stage majority voting. The evaluation spans seven benchmarks, and the results indicate that the proposed strategies enhance zero-shot NER across domain-specific out-of-distribution and general-domain datasets, including both Chinese and English languages. For the sake of brevity, we will refer to this work as *"**Zero-GPT NER**"* in this paper.

A study by Hu et al. (2023) [13], evaluated ChatGPT's performance for clinical named entity recognition (NER) tasks in a zero-shot setting, comparing it with GPT-3 and a fine-tuned BioClinicalBERT model. ChatGPT outperformed GPT-3 in the zero-shot scenario, showcasing the potential for clinical NER tasks without requiring annotations. However, its performance was still lower than the supervised BioClinicalBERT model, indicating the influence of prompt strategies on ChatGPT's performance.

In another study by Monteiro et al. (2023) [14] exploring methods to enhance NER performance in transformer-based models for Portuguese, researchers investigated various BERT-based models. Their findings highlighted the effectiveness of techniques such as in-domain pretraining, revealing significant improvements in NER accuracy providing valuable insights into optimizing NER models for specific languages and domains.

Studies comparing transfer learning and traditional approaches have also been conducted in the last few years. Aysu Ezen-Can (2020) [15] in their work addresses the performance of Bi-LSTM and fine-tuned BERT on a small dataset at intent classification. The experiments reveal that Bi-LSTM outperforms BERT significantly with higher accuracy in both validation and test data.

In the context of text analysis, a study conducted by Wen et al. (2024) [16] evaluated the effectiveness of ChatGPT alongside fine-tuned models like BART, ConvBERT, and GPT-2 using the Media Bias Identification Benchmark (MBIB). While ChatGPT demonstrates comparable performance to fine-tuned models in identifying hate speech and text-level context bias, it struggles with more nuanced biases such as fake news, racial, gender, and cognitive biases.

BERT was also compared with ChatGPT in a study by Qui et al. (2024) [17]. The authors conducted a comparative analysis between a fine-tuned BERT model and ChatGPT for the development of an intelligent design support system on a domain-specific dataset. ChatGPT performed comparably to the fine-tuned BERT model in sentence-level classification tasks but faced challenges with datasets featuring short sequences.

ChatGPT in most studies exhibits promising potential for applications in knowledge transfer and extraction.

III. Methodology

In this section, we detail the methodology of the comparative evaluation of Bi-LSTM, fine-tuned BERT, and ChatGPT at entity extraction. Additionally, we delve into specifics about the utilized dataset, the training procedure for Bi-LSTM, the fine-tuning process for the BERT model, and an analysis of the prompts used as input for ChatGPT.

A. Dataset

In contrast to existing methods that primarily focus on training to aggregate categories like places, dates, persons, etc., our approach aims to investigate the capabilities of the aforementioned models to make finer distinctions within class definitions. The dataset and the entities adapted from our work [1] are detailed in Table I. We explore whether the models can discern subtleties such as differentiating between a *hardware part* and a *hardware subpart* or a *component* and a *system*, acknowledging the nuanced nature of this distinction compared to more straightforward differentiations like people and places. This research is situated within the German GENIAL! project. Here, we utilize the GENIAL! Basic Ontology (GBO) as a shared vocabulary for exchanging information on microelectronic systems, components, functions, properties, and dependencies. The dataset from the domain-specific GBO comprised 2291 phrases and 55,400 tokens manually labeled under the supervision of ontology experts.

B. Information Extraction Model for ChatGPT

We devise an information extraction (IE) model to generate three distinct types of inputs for ChatGPT, ensuring a fair and expansive assessment of its performance.

The first approach of the model includes a carefully crafted input, engineered through our CRIO technique (**C**ontext-aware **R**ole-play induced **I**nput with **O**utput cue). This approach introduces the task context through a role-playing scenario and provides input guidelines. It incorporates external information to furnish additional context, helping the model in producing more precise responses. Moreover, the model incorporates output cues, specifying the desired type or format of the output. This facilitates the shaping of responses.

The technique is systematically structured in a way that ensures clarity in conveying the specific tasks expected of the model. An introduction denoted as role-playing, informs the system about its smart and intelligent capabilities. It then outlines the primary task of extracting specific entities from sentences and mentions that the system would be provided with entity definitions, sentences for extraction, and the desired output format with examples which form the context. This introduction sets the stage for the system's role and the context. The input cue serves as a detailed guide for entity definitions and tagging conventions in the NER task. Specific instructions are provided for assigning tags such as 'B-sys', 'I-sys', 'B-comp', 'I-comp', etc., according to our dataset detailed in section III-A. The cue outlines when to assign tags, whether it's the first identification in a sentence ('B-' tags) or subsequent ones ('I-' tags). Subsequently, the output cue is provided, instructing that results should be organized into categories. If no entities are found in a category, it should be marked as 'O.' This format ensures clarity and standardization in presenting the NER results. Additionally, the prompt includes examples that provide a practical template for the system to follow. The breakdown of entity types and their tagging conventions in this prompt strengthens the foundation for accurate and standardized NER results.

We devise a ChatGPT IE model based on *CRIO*, and two other techniques discussed in section II, *Chat2Extract*, and *Zero-GPT NER*. We use the GBO dataset to assess their effectiveness and performance. Figure 1 illustrates the ChatGPT IE model.

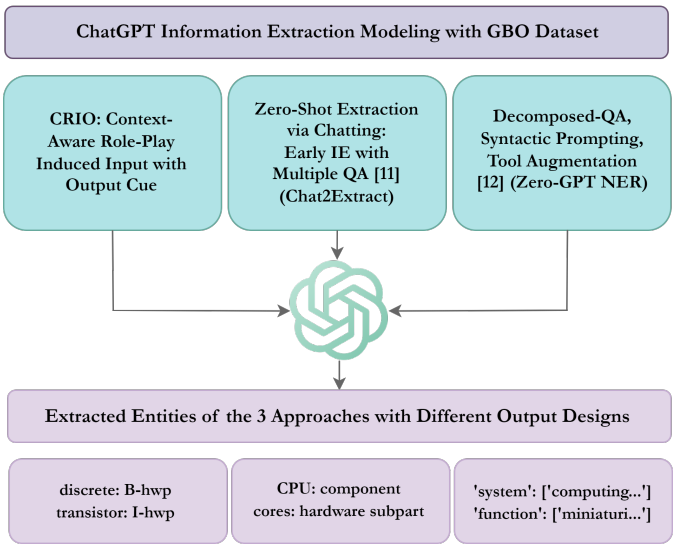

Fig. 1. Illustration of the IE model of ChatGPT showcasing the different aspects of prompt engineering approaches and their output designs.

C. Fine-tuning BERT

In the fine-tuning phase of the BERT model, we utilize the *bert-base-cased* [18] [19] pre-trained version, configuring the model architecture. The determination of the number of output labels was based on the *tag2idx* mapping length. We customize the configuration to exclude attention and hidden states from the output. We set the *full_finetuning* to *True*, and that in turn fine-tuned all parameters of the model. Fine-tuning also involves careful optimization using the *AdamW* optimizer, with varying weight decay rates for different parameter groups to enhance learning. Conversely, if *full_finetuning* is set to *False*, only the parameters of the classifier undergo fine-tuning. This configuration aims to strike a balance between model adaptability and computational efficiency.

During fine-tuning, the *AdamW optimizer*, with a learning rate of 2e-5 and an epsilon value of 2e-8, is applied to update model parameters. Essential training parameters, including the number of epochs, maximum gradient norm, and total training steps, are carefully set to conduct an effective training process.

TABLE I
GBO DATASET: ENTITIES AND THEIR MEANINGS (ADOPTED AND MODIFIED FROM [1])

Entity	Description
system	A system-level element, as defined by ISO 26262, constitutes a collection of components or subsystems that establish interconnections among at least a sensor, a controller, and an actuator.
component	As per ISO 26262: an entity that is logically or technically separable and encompasses multiple hardware parts or one or more software units.
hardware component	A non-system level element: an entity that is logically or technically separable and consists of multiple hardware parts.
hardware part	An individual piece of hardware defined as a segment of a hardware component at the initial level of hierarchical decomposition.
hardware subpart	A segment of a hardware part that can be logically divided and signifies a second or subsequent level of hierarchical decomposition.
function	A function that an element, such as a system, component, hardware, or software, executes or carries out.
software	A "software element" is executed by a "processing unit."
quantity	A standardized representation of a measurable aspect, like length, mass, or time.
measure	A standard used to express the amounts of quantities.
unit	A standard used for comparison in measurements.

Additionally, a *linear scheduler* with *warm-up* is applied to dynamically adjust the learning rate throughout training. To assess the model's performance and behavior, metrics such as loss values, precision, recall, and F1-score are used.

D. Bi-LSTM Development

The input layer of the Bi-LSTM model is designed to handle sentences with a maximum length of fifty tokens. An embedding layer initializes with random weights to learn word embeddings from the training data. In the embedding layer, the tokens, and labels are converted into index numbers. A spatial dropout1D layer follows, dropping entire 1D feature maps, and a bi-directional LSTM layer with two hidden layers processes input tokens in both directions. Outputs from this layer are concatenated. The time-distributed layer (Dense) serves as the output layer, corresponding to the maximum length and maximum number of tags. Training involves using index numbers for tokens and labels, simplifying the process. We train the model for 50 epochs, minimizing over- and under-fitting. Hyper-parameters, including LSTM units and dropouts, are fine-tuned through iterative experimentation and comparisons.

IV. EXPERIMENTAL DESIGN

NER training typically emphasizes token-level metrics like precision, recall, and F1-score, assessing individual token recognition. However, for subsequent tasks like generating triples for knowledge graphs, evaluating the NER system at a full named entity level is crucial. Post-NER, practical applications necessitate focusing on accuracy and completeness in recognizing entire named entities. In typical NER applications, limitations arise from a simple classification evaluation based on scenarios like false negatives (FN), true positives (TP), false negatives (FN), and false positives (FP), providing metrics for each entity type. This overlooks complexities as is the case in [11] and [12], omitting considerations for partial matches and cases where the NER model identifies the entity "boundary" correctly but assigns the wrong type. A comprehensive assessment is needed, beyond individual tokens, accounting for correct identification of the entire entity, including partial recognition or entity type misclassification.

MUC-5 Evaluation Metrics by Chinchor and Sundheim (1993) [20] introduced comprehensive metrics, categorizing errors based on model predictions versus ground truth (GT). Extending beyond strict classification, these metrics consider partial matching, aligning with outlined scenarios. The assessment emphasizes differences in NER output and GT, considering both the boundary and entity type [20]. Table II highlights the MUC-5 metrics.

TABLE II
MUC-5 EVALUATION METRICS [20]

Metric	Description
Correct	Both model prediction and GT match.
Incorrect	Model prediction and GT do not match.
Partial	Model prediction and GT are somewhat similar but not identical.
Missed	GT annotation is not captured by the model.
Spurious	Model produces a prediction that doesn't exist in the GT.

SemEval-2013 Task 9 by Segura-Bedmar et al. (2013) [21] introduced four distinct measurement approaches, as shown in Table III, building on the metrics defined by MUC-5. Each SemEval-2013 metric addresses MUC-5 metric scenarios of correct, incorrect, partial, missed, and spurious in distinct ways. These metrics are computed for entire entities and across both

dimensions (type and boundary) to assess the performance of each dimension independently.

TABLE III
SEMEVAL-2013 TASK 9 EVALUATION METRICS [21]

Metric	Description
Strict	Requires an exact boundary match for both boundary and entity type.
Exact	Requires an exact boundary match over the boundary, irrespective of the entity type.
Partial	Allows for a partial boundary match over the boundary, irrespective of the entity type.
Type Match	Requires some overlap between the model-tagged entity and the GT in terms of the entity type.

V. RESULTS

This section discusses the conducted experiments and the evaluation of the tested models, aiming for fairness within the resource constraints. Each model was tested on approximately 2250 unseen tokens from the microelectronics domain text. The focus is on assessing the performance, under restricted conditions, of ChatGPT utilizing our IE model, BERT fine-tuned on our dataset, and the trained Bi-LSTM model. The objective is to gain a comprehensive insight into the viability of these models within the domain-specific NER realm.

A. ChatGPT NER Analysis

In this subsection, we delve into the evaluation of the ChatGPT IE model, examining its performance across three distinct prompt models. The goal is to compare and assess the effectiveness of each prompt strategy.

1) CRIO Model: As discussed in section III-B, the CRIO model, designed as a prompting strategy with the fundamental principles of prompt engineering, is evaluated on our test data. The results as depicted in Figure 2, reveal suboptimal performance, with a high volume of ”spurious” instances indicating a high number of FPs. Despite this, there is a decent number of correct matches across all metrics. Partial overlap in type matching is respectable, along with the successful extraction of partial and exact entity boundaries. Strict matching, which only considers the exact type and boundary matches, still yields a decent number of correct matches, showing promise in certain aspects.

2) Chat2Extract: We used the templates provided by the authors to evaluate the model's performance on our test data. From Figure 3, it is evident that similar to the CRIO model, it generates a notable number of FPs. However, it surpasses the CRIO model in terms of boundary matching, with generally correct identification of ”partial” and ”exact” boundaries, indicating sufficient entity extraction.

The model outperforms the CRIO model in recognizing the structure of entities, demonstrating its strength in the extraction aspect. However, it faces challenges in assigning correct labels to extracted entities, particularly with classes that exhibit subtle differences, such as a *”hardware part”* and a *”hardware component”*. Additionally, the results include instances of non-existent classes such as *”events”*, further adding to the ”spurious” instances. The model relatively missed fewer occurrences of entities or their partial and strict boundaries, producing lower numbers of FNs.

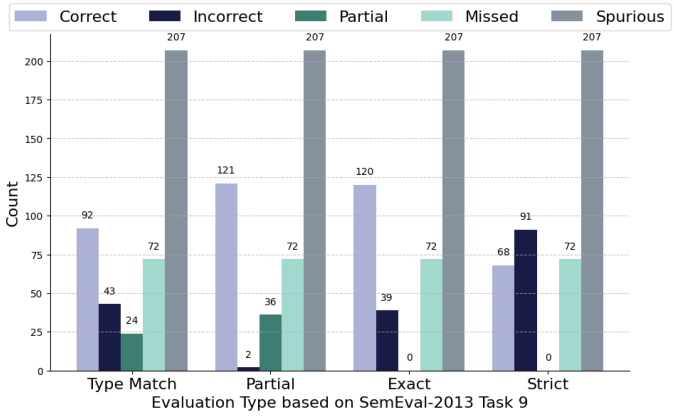

Fig. 2. CRIO Model: Comparison of correct, incorrect, partial, missed, and spurious metrics across different evaluation types (Type Match, Partial, Exact, Strict). The figure provides insights into the performance of the CRIO model under varying evaluation criteria.

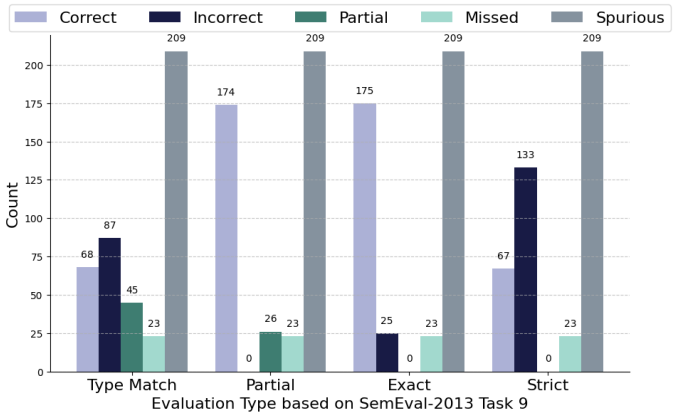

Fig. 3. Chat2Extract: Comparison of correct, incorrect, partial, missed, and spurious metrics across different evaluation types (Type Match, Partial, Exact, Strict).

3) Zero-GPT NER: In contrast to CRIO and Chat2Extract, this method exhibited an apparent reduction in FPs, producing low numbers of type and boundary ”spurious” results. However, its type-matching performance was subpar, yielding significantly fewer ”correct” outcomes in both partial and strict type matching. Boundary matching was comparable to the CRIO model, but Chat2Extract demonstrated superior performance in this regard. The frequency of FNs was the highest among the three approaches indicated by the high number of ”missed” instances. Although it generated fewer ”incorrect” outcomes than Chat2Extract, it was surpassed by the CRIO model in this aspect. Overall, the Zero-GPT NER approach was the least effective among the three methods.

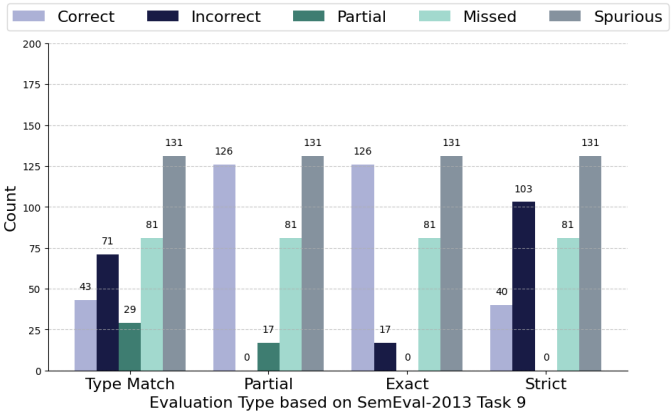

Fig. 4. Zero-GPT NER: Comparison of correct, incorrect, partial, missed, and spurious metrics across different evaluation types (Type Match, Partial, Exact, Strict).

B. Fine-Tuned BERT NER Analysis

The BERT fine-tuning process is applied to our GBO dataset, yielding improved overall results when compared to the ChatGPT IE Model, comprising the three different approaches. It exhibits notably fewer ”spurious” outcomes or FPs, registering the lowest incidence among all the ChatGPT IE approaches under discussion. Its performance fares well in crucial metrics, particularly in ”strict” boundary plus type matching, achieving a high count of ”correct” outcomes as illustrated in Figure 5. Moreover, BERT surpasses ChatGPT in the ”type match” category (partial type overlap), attaining a respectable number of ”correct” outcomes, despite producing a higher number of incorrect classifications. The extraction of ”partial” and ”exact” boundaries also outperforms ChatGPT. Given the domain-specific and highly limited nature of the dataset, the superior performance of the fine-tuned BERT to the ChatGPT IE Model indicates the potential for even greater effectiveness with dataset augmentation.

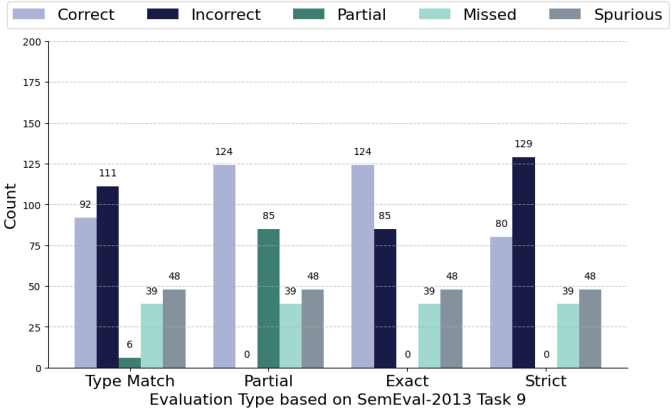

Fig. 5. Fine-tuned BERT: Comparison of correct, incorrect, partial, missed, and spurious metrics across different evaluation types (Type Match, Partial, Exact, Strict).

C. Bi-LSTM NER Analysis

The Bi-LSTM model exhibited superior rates of partial overlap in type matching compared to ChatGPT and fine-tuned BERT, accompanied by an exceptionally low count of incorrect outcomes. Remarkably, the ”correct” occurrences of both ”partial” and ”exact” boundary matching were noteworthy, albeit with a higher occurrence of incorrect ”exact” boundary extractions than the ChatGPT IE Model but lower than the fine-tuned BERT. The most noteworthy achievement was observed in ”strict” type and boundary matching, where the Bi-LSTM model outperformed both ChatGPT IE Model and fine-tuned BERT. Additionally, the Bi-LSTM model displayed fewer false negatives (”missed” instances) compared to ChatGPT. The occurrences of ”spurious” outcomes were also minimal. The Bi-LSTM model demonstrated superior performance in comparison to both ChatGPT and the fine-tuned BERT.

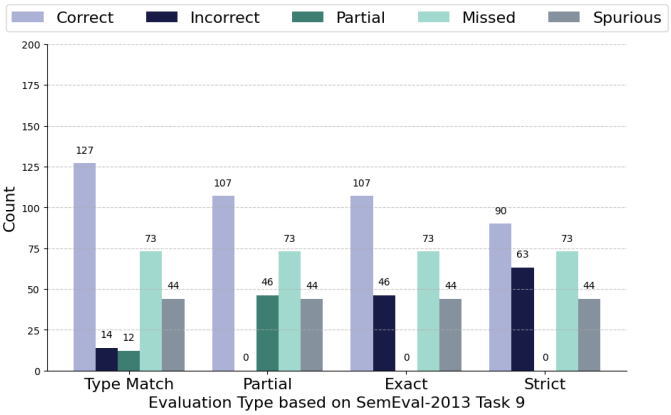

Fig. 6. Bi-LSTM: Comparison of correct, incorrect, partial, missed, and spurious metrics across different evaluation types (Type Match, Partial, Exact, Strict).

D. Discussion

The computational analysis is performed in two distinct manners, dependent upon the desired matching criteria—whether it involves an exact match (referred to as ”strict & exact”) or a partial match (termed ”partial & type”), following the SemEval-2013 task 9 metrics and the calculation methodology proposed in [22]. It reveals insights into the performance of the three ChatGPT prompt approaches, the fine-tuned BERT, and the Bi-LSTM model, as illustrated in Figure 7.

Among three ChatGPT prompt approaches, Chat2Extract exhibits superior performance on our domain-specific test data, closely followed by the CRIO model. Notably, Chat2Extract’s two-stage approach allows for refining queries, providing an advantage over CRIO. On the other hand, Zero-GPT NER demonstrates inadequate performance, particularly in terms of relatively lower recall, indicative of a substantial number of FNs in both type and boundary matching, in ”strict” and ”partial” extractions. Chat2Extract, while showcasing a decent performance in extracting ”partial” overlaps in type and boundary (reflected in high recall for ”partial & type”),

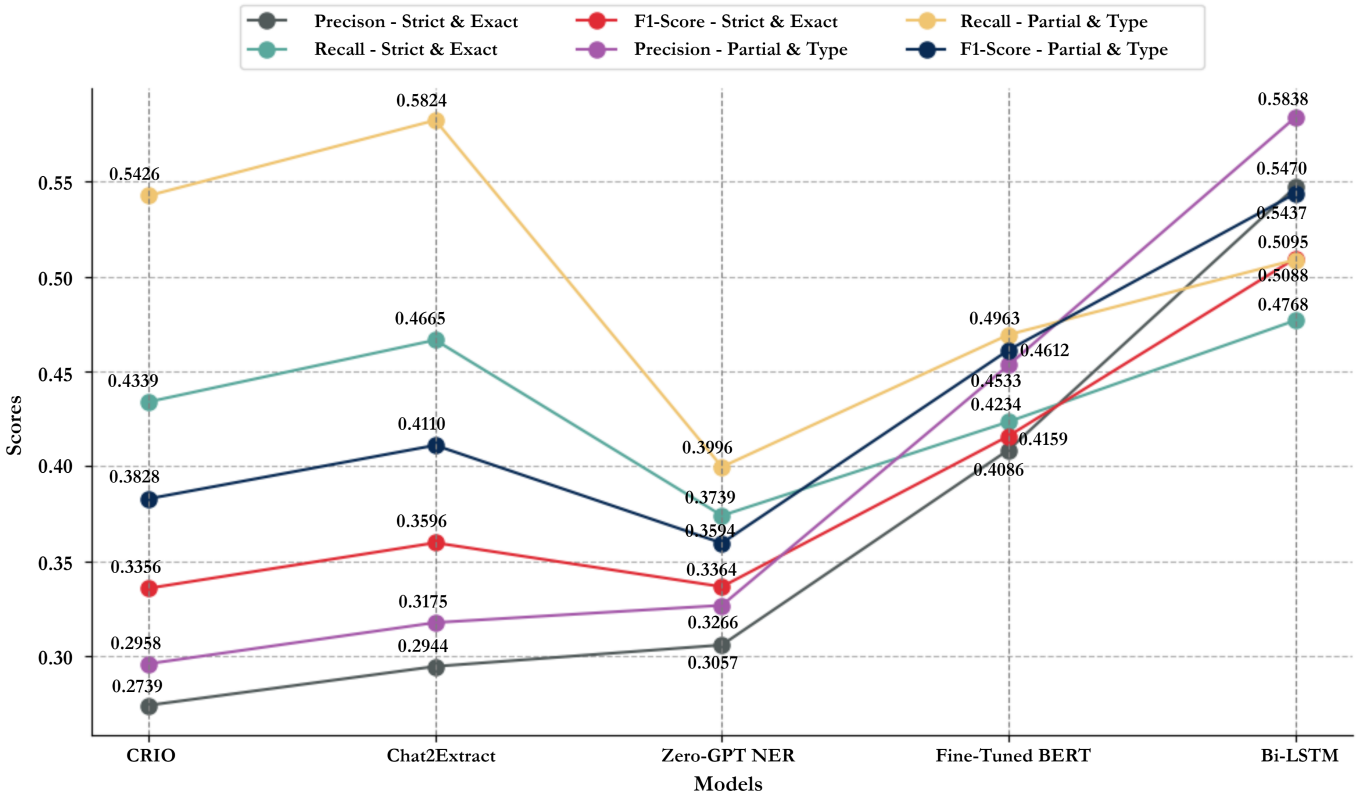

Fig. 7. Comparison of all experimented models: precision, recall, and F1-scores across ”strict & exact” and ”partial & type match” metrics.

struggles with accurate label assignment to the extracted entities, resulting in a high number of ”spurious” outcomes or FPs. This pattern is consistent across CRIO and Zero-GPT NER as well.

ChatGPT, known for its contextual generalization capabilities, tends to expectedly overshoot entity and boundary predictions in domain-specific NER, leading to higher volumes of FPs. This characteristic proves advantageous in scenarios where extracting entity and boundary is more critical than correctly labeling the extracted entity, as observed in cases like drug-drug interactions discussed in SemEval-2013 Task 9. It is to be noted that, owing to the zero-shot classification method employed by ChatGPT IE Model prompts, their performances remain unaffected by the limited dataset.

On the other hand, the fine-tuned BERT outperformed the ChatGPT IE Model, achieving higher F1-scores on both ”strict & exact” and ”partial & type” metrics. The fine-tuned BERT exhibited statistically superior performance in exact matches, with higher precision, indicating more accurate label assignments compared to all ChatGPT prompting strategies. In terms of partial matches, it also demonstrated higher precision, suggesting better extraction of entities and more accurate label assignments than ChatGPT.

The Bi-LSTM model showcased much better performance across all metrics. It achieved the highest precision and recall for strict matching, indicating a robust recognition and extraction rate, along with correct label assignments, outperforming both ChatGPT and the fine-tuned BERT. In partial matching, the Bi-LSTM model demonstrated superior precision compared to the other models, with only Chat2Extract exhibiting better recall values. The overall performance, as indicated by the partial and strict F1-scores, was significantly higher than those of the fine-tuned BERT and the individual ChatGPT models. The extent of improvement achieved by Bi-LSTM over the other models is quantified in Table IV.

In their recent study, Wei et al. (2023) assert the robustness and efficiency of zero-shot classification using LLMs like ChatGPT, emphasizing the advantage of not requiring explicit training data. Contrary to this claim, our experiments reveal that both the fine-tuned BERT and the trained Bi-LSTM derive significant benefits from the training they undergo. This observation holds particularly true for not-so-popular domain-specific datasets like GBO. Despite being considered less favorable in the era of transformers, the Bi-LSTM remains a viable option, especially in scenarios with limited and highly domain-specific corpus, involving classes with nuanced differences in definitions. While the choice of the model is dependent on specific use cases, our experiments serve as a reminder not to dismiss traditional methods prematurely.

VI. Conclusion

In this study, we explored various methodologies for NER, ranging from contemporary transfer learning models such as BERT and ChatGPT to the more conventional Bi-LSTM.

TABLE IV
PERCENTAGE IMPROVEMENT OF BI-LSTM OVER OTHER MODELS

Models	% Improvement (Strict)	% Improvement (Partial)
Bi-LSTM vs CRIO	51.8%	42.0%
Bi-LSTM vs Chat2Extract	41.7%	32.3%
Bi-LSTM vs Zero-GPT NER	51.5%	51.3%
Bi-LSTM vs BERT	22.5%	17.9%

Through the implementation of these approaches, we conducted a thorough analysis of their performances, based on metrics from MUC-5 and SemEval-2013 Task 9. The Bi-LSTM considered a more traditional and older technology, statistically outperformed state-of-the-art transformers. A tradeoff exists that correlates training data, F1-scores, and tagging quality with prompt engineering skills. Our experimentation with transfer learning revealed that the Bi-LSTM architecture yielded better results for our small GBO dataset, showcasing superior F1-scores in both strict and partial metrics. Apparently, for smaller datasets, we observed that BERT tends to overfit more compared to the Bi-LSTM architecture. While transfer learning with advanced transformers offers outstanding versatility and power, a thoughtful approach to their application is crucial. The selection of the appropriate model should consider various factors, including the size and prominence of the corpus, as well as the specific domain requirements.

VII. ACKNOWLEDGMENT

This work was funded by the BMBF Projects, KI4Boardnet and GENIAL. The authors would also like to acknowledge that ChatGPT was sparingly used to enhance the linguistic clarity of this manuscript, specifically the spelling and grammar. Consolidated results, code, and ChatGPT prompt templates used in this paper can be found at *https://github.com/khushnood-rafique/Domain-Specific-NER-A-Comparative-Analysis*.

REFERENCES

[1] F. Wawrzik, K. A. Rafique, F. Rahman, and C. Grimm, "Ontology Learning Applications of Knowledge Base Construction for Microelectronic Systems Information," Information, vol. 14, no. 3, p. 176, 2023. https://doi.org/10.3390/info14030176

[2] Graves, A., Fernández, S., Gomez, F., & Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. Neural Networks, 18(5-6), 602-610. doi:10.1016/j.neunet.2005.06.042

[3] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*, arXiv preprint arXiv:1810.04805 (2018).

[4] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, *Improving Language Understanding by Generative Pre-training*, OpenAI, Tech. Rep., 2018.

[5] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, *RoBERTa: A Robustly Optimized BERT Approach*, arXiv preprint arXiv:1907.11692, 2019.

[6] OpenAI. (2023). ChatGPT (Mar 14 version) [Large language model].

[7] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., & Agarwal, S. (2020). Language models are few-shot learners. arXiv preprint arXiv:2005.14165.

[8] Jouni Luoma and Sampo Pyysalo, *Exploring Cross-sentence Contexts for Named Entity Recognition with BERT*, 2020, https://arxiv.org/abs/2006.01563, arXiv:2006.01563 [cs.CL].

[9] Fábio Souza, Rodrigo Nogueira, and Roberto Lotufo, *Portuguese Named Entity Recognition using BERT-CRF*, 2020, https://arxiv.org/abs/1909.10649, arXiv:1909.10649 [cs.CL].

[10] Kai Hakala and Sampo Pyysalo. "Biomedical Named Entity Recognition with Multilingual BERT." In *Proceedings of the 5th Workshop on BioNLP Open Shared Tasks*, Nov 2019, Hong Kong, China. Edited by Jin-Dong Kim, Claire Nédellec, Robert Bossy, Louise Deléger. Published by the Association for Computational Linguistics. Pages 56–61. DOI: https://aclanthology.org/D19-5709.

[11] Xiang Wei, Xingyu Cui, Ning Cheng, Xiaobin Wang, Xin Zhang, Shen Huang, Pengjun Xie, Jinan Xu, Yufeng Chen, Meishan Zhang, Yong Jiang, and Wenjuan Han. "Zero-Shot Information Extraction via Chatting with ChatGPT." arXiv preprint arXiv:2302.10205, 2023. https://arxiv.org/abs/2302.10205.

[12] Tingyu Xie, Qi Li, Jian Zhang, Yan Zhang, Zuozhu Liu, and Hongwei Wang. "Empirical Study of Zero-Shot NER with ChatGPT." arXiv preprint arXiv:2310.10035, 2023. https://arxiv.org/abs/2310.10035.

[13] Yan Hu, Iqra Ameer, Xu Zuo, Xueqing Peng, Yujia Zhou, Zehan Li, Yiming Li, Jianfu Li, Xiaoqian Jiang, Hua Xu, *Zero-shot clinical entity recognition using ChatGPT*, *arXiv preprint* arXiv:2303.16416, (2023), https://arxiv.org/abs/2303.16416.

[14] Monique Monteiro and Cleber Zanchettin, *Optimization Strategies for BERT-Based Named Entity Recognition*, *Brazilian Conference on Intelligent Systems*, pages 80–94, (2023), https://link.springer.com/chapter/10.1007/978-3-030-91488-7_7.

[15] Aysu Ezen-Can, *A Comparison of LSTM and BERT for Small Corpus*, (2020), https://arxiv.org/abs/2009.05451.

[16] Zehao Wen and Rabih Younes, *ChatGPT vs Media Bias: A Comparative Study of GPT-3.5 and Fine-tuned Language Models*, *arXiv preprint arXiv:2403.20158*, (2024).

[17] Yunjian Qiu and Yan Jin. "ChatGPT and finetuned BERT: A comparative study for developing intelligent design support systems." *Intelligent Systems with Applications*, Volume 21, 2024, Article 200308. ISSN 2667-3053. https://doi.org/10.1016/j.iswa.2023.200308.

[18] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." *arXiv preprint arXiv:1810.04805*, 2019. https://arxiv.org/abs/1810.04805.

[19] Google Research. *BERT: Pre-trained models and downstream applications*. GitHub repository, 2018. https://github.com/google-research/bert.

[20] Nancy Chinchor and Beth Sundheim, *MUC-5 Evaluation Metrics*. In *Fifth Message Understanding Conference (MUC-5): Proceedings of a Conference Held in Baltimore, Maryland, August 25-27, 1993*, 1993. https://aclanthology.org/M93-1007.

[21] Isabel Segura-Bedmar, Paloma Martínez, and María Herrero-Zazo, *SemEval-2013 Task 9: Extraction of Drug-Drug Interactions from Biomedical Texts (DDIExtraction 2013)*. In *Second Joint Conference on Lexical and Computational Semantics (SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, edited by Suresh Manandhar and Deniz Yuret, June 2013, Atlanta, Georgia, USA, Association for Computational Linguistics, https://aclanthology.org/S13-2056, pages 341–350.

[22] David Batista, *Named Entity Recognition and Classification: A Review*. [Online]. Available: https://www.davidsbatista.net/blog/2018/05/09/Named_Entity_Evaluation/. [Accessed: Date Accessed].