

SUNDER: Selective Unmasking for Text Understanding

Alphaeus Dmonte¹, Tharindu Ranasinghe², Marcos Zampieri¹

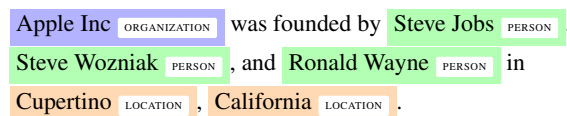
¹George Mason University, USA, ²Lancaster University, UK
admonte@gmu.edu

Abstract

Decoder-only large language models (LLMs) employ autoregressive text processing, attending exclusively to preceding tokens when generating subsequent ones. Although this architectural design excels in text generation applications, it constrains model performance on certain text understanding tasks. In this work, we propose SUNDER, a novel approach that enables partial bidirectional attention computation. We validate the proposed approach on three token spans identification tasks, such as Named Entity Recognition (NER), Aspect-based Sentiment Analysis (ABSA), and Definition Extraction (DEF). Results show that SUNDER improves LLM performance by up to 5 percentage points compared to traditional instruction-tuning. We further validate the robustness of our approach on multiple model families, model sizes, and prompting techniques. Our evaluation extends to multilingual scenarios and single-hop question answering (QA), confirming SUNDER's effectiveness on multiple settings. The code is made available to the community.¹

1 Introduction

Token span identification tasks, such as Named Entity Recognition (NER) (Seow et al., 2025), Toxic Spans Detection (TSD) (Pavlopoulos et al., 2021), and Aspect-Based Sentiment Analysis (ABSA) (Hua et al., 2024), are among the most widely studied tasks in NLP. The goal of these tasks is to detect and label specific tokens or contiguous sequences of tokens in the input text with predefined categories (Figure 1) (Papay et al., 2020). Accurate token span identification not only improves task performance but also enhances explainability in downstream applications, for example, by highlighting which tokens in a sentence convey negative sentiment or polarity (Minh et al., 2022).



Apple Inc ORGANIZATION was founded by Steve Jobs PERSON, Steve Wozniak PERSON, and Ronald Wayne PERSON in Cupertino LOCATION, California LOCATION.

Figure 1: An example of token spans identification task (Named Entity Recognition).

Successful token span identification requires a holistic understanding of the input text, as the meaning of a token or span often depends on its broader context (Poddar et al., 2024). Consequently, these tasks are crucial for evaluating the language understanding capabilities of language models and are commonly included in standard benchmarks, such as GLUE (Wang et al., 2018) and SuperGLUE (Wang et al., 2019), along with their multilingual adaptations (Kakwani et al., 2020; Plum et al., 2026). Results from these benchmarks indicate that transformer-based encoder-only models, such as BERT (Devlin et al., 2019), achieve state-of-the-art performance in identifying and extracting token spans (Spala et al., 2020; Ranasinghe and Zampieri, 2021; Kaur et al., 2024).

More recent decoder-only LLMs, such as LLaMA-3 and GPT-4, have demonstrated remarkable capabilities across a wide range of NLP tasks (Brown et al., 2020). Their left-to-right, autoregressive text processing has achieved state-of-the-art results in text generation tasks (Lei and Huang, 2025; Afzal et al., 2025) and dialog (Mo et al., 2025; Oruche et al., 2024, 2025). However, these models often underperform smaller encoder-only models, such as BERT (Devlin et al., 2019) and RoBERTa, on semantic understanding tasks like token span identification and sequence labeling (Han et al., 2023; Wang et al., 2025; Dmonte et al., 2025). As these tasks often require access to both past and future context to fully capture semantic relationships, the autoregressive nature of decoder-only models limits their effectiveness

¹<https://github.com/alphaeusdm/SUNDER.git>

due to the causal attention mask blocking future context.

Multiple approaches have proposed incorporating bidirectional attention into decoder-only models. Recent methods propose removing the causal mask in all or selected layers (Li et al., 2023; Dukić and Šnajder, 2024), which requires fine-tuning the model with sequence- or token-level classification objectives, deviating from the original causal language modeling pre-training objective. This creates a mismatch between pre-training and fine-tuning, undermining the autoregressive property and, in practice, converting these decoder-only models into encoder-like models, thus reducing their effectiveness for generative tasks (Yang et al., 2024).

To address this gap, we introduce **Selective Unmasking for Text Understanding - SUNDER**, a method that selectively removes the causal mask for specific tokens. We answer a fundamental research question (RQ): **Does selectively removing the causal mask from decoder-only models bring performance improvement to vanilla LLMs?** To achieve this, the mask is removed only for the actual input text, while it is retained for instructions and other meta-text in the prompt, such as formatting guidelines or constraints. This selective unmasking allows the model to access future tokens within the input, enabling bidirectional attention and improving text understanding. Notably, SUNDER supports evaluation in zero- and few-shot settings, without requiring fine-tuning with sequence- or token-level classification objectives.

The main **contributions** of this work are three-fold:

1. We **propose SUNDER**, a novel method that introduces selective bidirectionality in decoder-only LLMs, improving their ability to perform token span identification while preserving the autoregressive property of LLMs during output generation.
2. We **demonstrate the generalizability** of SUNDER across multiple tasks, model families, and model sizes.
3. We **show SUNDER’s applicability in multiple languages** such as English, French, German, Hindi, Italian, Sinhala, and Spanish; **and multiple tasks** such as NER, definition extraction, aspect-based sentiment analysis, and single-hop QA.

2 Related Work

2.1 Token Spans Identification

Token span identification requires models to detect and classify contiguous sequences of tokens into predefined categories (Papay et al., 2020; Zampieri et al., 2023; Dmonte et al., 2025). These tasks can be broadly categorized into two types based on their primary objective: (i) Primary span identification tasks, where identifying and classifying spans is the end goal itself, such as Named Entity Recognition (NER) (Seow et al., 2025), chunking (Qu et al., 2025), and keyphrase detection (Kong et al., 2023); and (ii) Explainability-oriented span identification tasks, where spans serve as rationales to explain or interpret predictions from a classification task, such as Toxic Spans Detection (TSD) (Pavlopoulos et al., 2021), which identifies the specific text fragments that make a comment toxic, and aspect term extraction in Aspect-Based Sentiment Analysis (ABSA) (Hua et al., 2024), which highlights which product features drive sentiment judgments (Minh et al., 2022). Both of these task types have been widely used in language understanding benchmarks (Hardalov et al., 2023).

Traditionally, encoder-only transformer models such as BERT (Devlin et al., 2019), XLM-RoBERTa (Höfer and Mottahedin, 2023), and DeBERTa (Martin et al., 2022) have achieved state-of-the-art performance on span identification tasks. The standard approach involves adding a token classification head to a pretrained encoder and fine-tuning on task-specific data (Devlin et al., 2019; Kaur et al., 2024). To further improve performance, Conditional Random Field (CRF) layers (Lafferty et al., 2001) are commonly added on top of transformer encoders to model label dependencies and enforce sequence-level constraints (Papay et al., 2020). More sophisticated approaches, such as Relation Aware Collaborative Learning (Chen and Qian, 2020), have been proposed to handle complex relationships between different span types. Models like UniLM and Encoder-Decoder based models like T5 have also been explored for these tasks, where task specific masking strategy can be used in the former while the latter uses an encoder for the task input and a decoder to generate the output. Unlike these models, our approach changes only the partial causal mask, allowing both causal and bidirectional attention computation simultaneously. Additionally, our approach uses a pretrained model, unlike these models that need to be pretrained.

2.2 LLMs and Token Spans Identification

Large Language Models have shown improved performance on many NLP tasks (Brown et al., 2020). Given these successes, recent work has explored applying LLMs to span identification tasks (Han et al., 2023; Popovič and Färber, 2024; Morand et al., 2026; Dmonte et al., 2025). Most existing approaches rely on prompting strategies or instruction-based fine-tuning (Wang et al., 2025). For example, Han et al. (2023) leveraged GPT for four tasks, namely NER, Relation Extraction, Entity Extraction, and ABSA. Similarly, Dmonte et al. (2025) used several LLMs for identifying token spans for simple and complex text. They explored several techniques, and the experiments show that LLMs consistently struggle to replicate the performance of encoder-only models on most tasks.

The main limitation of decoder-only LLMs on semantic understanding tasks lies in their autoregressive text processing. Unlike encoder-only models, decoder-only models do not have access to future tokens, hence losing critical information from the future tokens. To improve the LLM performance on language understanding tasks such as NER, ABSA, etc, several approaches have been proposed that allow access to the future tokens (Li et al., 2023; Dukić and Šnajder, 2024; BehnamGhader et al., 2024; Li and Li, 2024; Lee et al., 2025; Muennighoff et al., 2025; Springer et al., 2025; Katz et al., 2025).

Existing solutions can be broadly categorized into prompting-based methods and architectural modifications. Prompting-based approaches attempt to compensate for the lack of bidirectional context by manipulating the input, for example, by repeating the input text within the prompt to expose the model indirectly to future information (Springer et al., 2025). In contrast, architectural approaches explicitly modify the attention mechanism by converting some or all attention layers from uni-directional to bidirectional. Notably, Li et al. (2023) transformed all attention layers of the LLaMA model into bidirectional ones and fine-tuned the model using a token classification objective, achieving improved performance on NER. Similarly, Dukić and Šnajder (2024) enabled access to future tokens in a subset of attention layers. To better preserve generative capabilities, Li and Li (2024) proposed modifying only a limited number of layers, striking a balance between bidirectional understanding and autoregressive generation.

Similarly, Katz et al. (2025), proposed a segment-based masking approach that allows bidirectional attention computations for user and system prompts while preserving autoregressive generation for chat-based tasks.

While these methods introduce bidirectional attention by altering large portions of the model architecture and, in some cases, changing the original pretraining objective, our approach takes a different direction. We propose selectively granting access to future tokens for specific tokens, while retaining the causal language modeling objective of the pretrained model. This design aims to enhance semantic understanding without sacrificing the generative strengths that define decoder-only LLMs.

3 SUNDER - Proposed Approach

The LLMs consist of a causal mask (CM) that is applied to the dot product of Query (Q) and Key (K) vectors before computing the softmax. This restricts the model from attending to the future tokens during attention computation. The causal mask for the decoder-only model is defined as a triangular matrix, where the lower triangle is set to 0 while the upper triangle is set to $-\infty$ to prevent access to the future tokens during attention computation.

$$CM_{i,j} = \begin{cases} 0, & \text{if } j \leq i \\ -\infty, & \text{otherwise} \end{cases}$$

During the attention computation, before computing the *softmax*, CM is added to the product of Q and K.

$$Attn(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}} + CM)V$$

As shown in Li et al. (2023), removing the causal mask improves the LLM’s performance on text understanding and classification tasks. Unlike their approach, where they remove the causal mask for the entire prompt (instruction + input), we only remove the causal mask for the main input in the prompt. This allows the model to attend to the future tokens of the input, which can help in understanding the context.

In order to indicate what part of the input the causal mask is to be removed from, we introduce two special delimiters: *<start>* to indicate the start of the input and *<end>* to indicate the end of the input. The causal mask is changed such that it allows all the tokens between the *<start>* and *<end>*

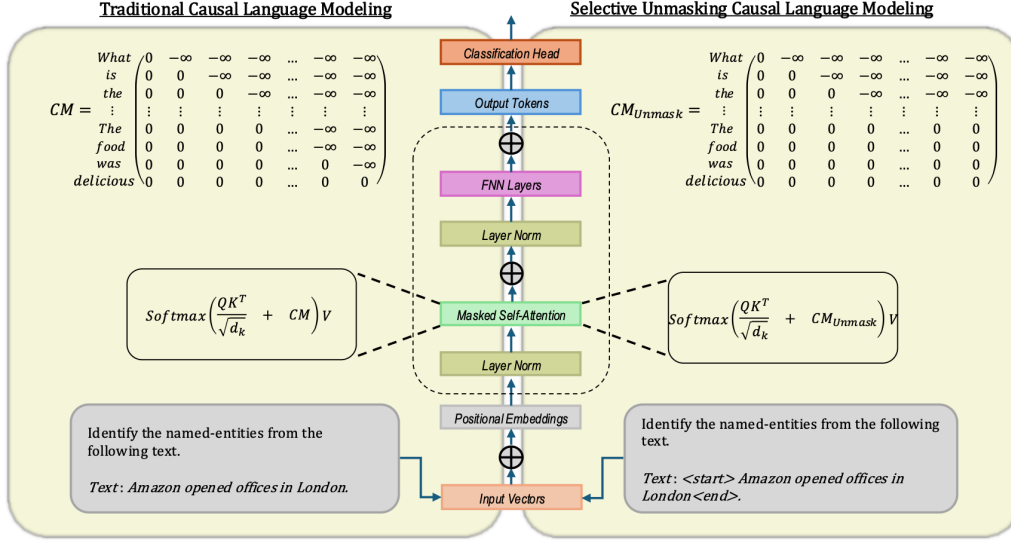


Figure 2: SUNDER: The proposed selective unmasking approach compared to traditional causal language modeling.

tokens to attend to future tokens, i.e., the CM matrix values for all these tokens are changed from $-\infty$ to 0. The updated causal mask is as follows:

$$UCM_{i,j} = \begin{cases} 0, & \text{if } (j \leq i) \text{ or} \\ & (i \geq ind_s \text{ and } j \leq ind_e) \\ -\infty, & \text{otherwise} \end{cases}$$

where, ind_s is the $\langle start \rangle$ index and ind_e is the $\langle end \rangle$ index. Here, UCM represents the unmasked causal mask. We initialize the $\langle start \rangle$ and the $\langle end \rangle$ token embeddings with the $\langle bos \rangle$ and $\langle eos \rangle$ token embeddings respectively, to avoid random initialization of input and output the embedding vectors. Additional details of our approach are provided in Appendix A.1.

4 Experimental Setup

4.1 Tasks and Datasets

We evaluate SUNDER on three token span identification tasks using four available English datasets presented in this section. The number of instances used for training, development, and testing for each task are shown in Table 1.

Named-Entity Recognition (NER) We use the dataset from the CoNLL 2003 shared task (CoNLL) (Sang and De Meulder, 2003). The dataset consists of examples in English and German annotated with four named entity labels, *person*, *location*, *organization*, and *miscellaneous names*. We only use the English examples for our experiments. Additionally, we use English examples from the

WikiNEuRal (Wiki) (Tedeschi et al., 2021) dataset that consists of examples in nine languages.

Definition Extraction (DEF) We use the DeftEval shared task dataset (Spala et al., 2020). We use the token spans identification dataset consisting of 13 unique labels for definitions and terms.

Aspect-based Sentiment Analysis (ABSA) We use the dataset from SemEval 2014 Task 4 (Pontiki et al., 2014). Since this dataset consists only of the annotated aspect terms, we use the dataset with extended annotations for the opinion terms annotated by Wang et al. (2016).

4.2 Baselines and Evaluation

In our experiments, we use three large decoder-only models, with vanilla instruction-tuning serving as the primary baseline.

Recent large decoder-only models, that process text from left-to-right, have shown exceptional capabilities on several NLP tasks, including generation and understanding. For our experiments, we mainly use three models: Llama-3.1-8B, Mistral-7B-v0.3, and Qwen3-8B. We primarily use these small models for our experiments; however, we also considered larger models like Llama-3.1-70B for ablations, showing that SUNDER works with larger models as well. The vanilla instruction-tuning is used as the primary baseline. The base versions are instruction-tuned on task-specific training data using the prompts shown in Tables 7.

We compare SUNDER with four additional baselines. Following Li et al. (2023) and Dukić and

Task	Example Instance	Span Types	Reference	Train	Dev	Test	Total
CoNLL	Sebastian Thrun PERSON started at Google ORGANIZATION in 2007 MISCELLANEOUS	Person, Location, Organization, Miscellaneous	Sang and De Meulder (2003)	3,000	300	300	3,600
DEF	The scientific method TERM is a method of research with defined steps DEFINITION	Term, Alias Term, Ordered Term, Referential Term, Definition, Secondary Definition, Ordered Definition, Referential Definition, Qualifier	Spala et al. (2020)	6,969	332	353	7,654
ABSA	The pasta ASPECT was delicious OPINION but service ASPECT was slow OPINION	Aspect, Opinion	Wang et al. (2016)	1,902	447	673	3,022

Table 1: Overview of token span identification datasets we experimented with annotated examples and data splits.

Šnajder (2024), we remove the causal mask and replace it with a bidirectional mask. Similar to Li et al. (2023), we remove the causal mask from all the attention layers for one baseline, while for the other, we remove the causal mask only for a subset of layers. Specifically, the causal mask is removed for all the layers between layers 8 and 24. These layers are selected based on the best configuration results from Dukić and Šnajder (2024). For both these unmasking strategies we fine-tune the LLMs using standard instruction tuning and for token classification. We also use fine-tuned RoBERTa and Flan-T5 models as baselines.

Training runs for 1000 steps with a learning rate of $1e-4$, using the AdamW optimizer and a cosine learning rate scheduler. We set the training batch size to 2 with gradient accumulation over 8 steps, and use a maximum sequence length of 512 tokens, which accommodates most sequences in our datasets. To reduce computational requirements for LLMs, we apply LoRA with rank 16, alpha value of 32, and dropout of 0.1. Same hyperparameters are used to instruction-tune the LLMs with SUNDER, with prompts shown in Table 8. All models were trained on two A100 80GB GPUs.

Finally, to evaluate the performance of the models, we compute the micro F1 score using sequeval², considering the exact match of the extracted spans.

5 Results

Table 2 shows the performance of Llama, Mistral, and Qwen models on the three tasks. Specifically, we compare the performance of our proposed approach with vanilla instruction tuning of the LLM as the baseline. We run the experiments five times and report the average evaluation scores. For all models and evaluation measures, the standard deviation was negligible.

As shown, for all the evaluated tasks, SUNDER

outperforms the vanilla instruction tuning approach. Especially on the ABSA task, SUNDER outperforms vanilla instruction tuning by upto 8.5 points with the Mistral and Qwen models, while our approach outperforms vanilla instruction tuning by 17.3 points on the Llama model.

Approach	Model	CoNLL	Wiki	DEF	ABSA
Encoder	RoBERTa	0.865	0.862	0.654	0.868
Encoder-Decoder	Flan-T5	0.897	0.818	0.595	0.791
UM-All-IT	Llama-8B	0.000	0.000	0.212	0.208
	Mistral-7B	0.862	0.744	0.657	0.647
	Qwen3-8B	0.000	0.023	0.000	0.015
UM-LG-IT	Llama-8B	0.860	0.824	0.689	0.694
	Mistral-7B	0.880	0.770	0.682	0.696
	Qwen3-8B	0.870	0.771	0.610	0.610
UM-All-TC	Llama-8B	0.803	0.876	0.536	0.808
	Mistral-7B	0.894	0.884	0.627	0.858
	Qwen3-8B	0.848	0.863	0.630	0.812
UM-LG-TC	Llama-8B	0.871	0.817	0.478	0.826
	Mistral-7B	0.859	0.832	0.558	0.849
	Qwen3-8B	0.790	0.813	0.456	0.823
Vanilla-TC	Llama-8B	0.653	0.744	0.452	0.732
	Mistral-7B	0.712	0.788	0.463	0.749
	Qwen3-8B	0.658	0.752	0.431	0.720
Vanilla	Llama-8B	0.875	0.839	0.684	0.690
	Mistral-7B	0.888	0.801	0.699	0.740
	Qwen3-8B	0.875	0.797	0.682	0.700
SUNDER	Llama-8B	0.899	0.881	0.708	0.863
	Mistral-7B	0.898	0.866	0.699	0.825
	Qwen3-8B	0.888	0.872	0.690	0.778

Table 2: Results (Micro F1) comparison between fine-tuning LLM (Traditional approach) and SUNDER. TC=Token Classification and IT=Instruction Tuned. Best scores for each task are displayed in bold.

On CoNLL, between the vanilla instruction-tuned and SUNDER models, we see a performance improvement of upto 2 points across the three models. For Wiki dataset, we see that SUNDER outperforms all vanilla tuned models by upto 7.5 points. Specifically, there is a substantial performance improvement with the Qwen model of 7.5 points,

²<https://github.com/chakki-works/sequeval>

while for Llama and Mistral models, we see gains of 4.2 and 6.5 points respectively. We also see a performance improvement on the DEF compared to the vanilla instruction tuning with gains of upto 2.5 points.

Overall, across all the tasks, we see the performance of LLMs improve with SUNDER compared to the vanilla approach. Among the evaluated LLMs, Llama performed the best on all four tasks, followed by the Mistral model. The results demonstrate the effectiveness of introducing bidirectionality in LLMs.

SUNDER also outperforms the other baselines by up to 26 points across all tasks. This shows that removing the entire causal mask from the attention layers may introduce a mismatch with the pretraining objective and the models may learn inaccurate representations. This is notably demonstrated in the UM-All-IT approach, where models like Llama and Qwen3 do not generate any output or produce irrelevant tokens for some tasks like NER. The results further demonstrate that selectively allowing bidirectionality during attention computation while maintaining the pretraining objective helps with text understanding compared to removing the entire causal mask.

6 Analysis and Discussion

In this section, we analyze the instruction-tuned models and further discuss the results of additional ablations conducted to evaluate the generalizability of our proposed approach in different LLM settings and across model sizes. This section is organized as follows: we first analyze the per-label performance of the instruction-tuned models in Section 6.1. The performance of in-context learning (ICL) is discussed in Section 6.2, followed by a discussion on the performance across different model sizes in Section 6.3.

6.1 Per Label Performance

From Section 5 we see that SUNDER improves the performance of LLMs on all four datasets. In this section, we look at how the performance changes across the individual labels for each dataset. For this analysis, we only consider Llama-8B model. We see similar performance improvements on the Mistral-7B and Qwen-7B models. Figure 4 show the per-label performance for each dataset. Looking at CoNLL, we see that the F1-score for *Miscellaneous entities* improves the most followed by

Organization, *Person*, and *Location*. For Wiki, *Location* and *Organization* slightly underperforms vanilla LLM. Among the four named-entities, *Miscellaneous entities* is more context dependent.

The DEF task has ten labels, of which six labels have limited examples in the dataset, leading to an F1-score of 0. Hence, for this analysis, we only consider the four labels that have sufficient examples in the dataset. From Figure 4, we see that on three of the four labels considered, SUNDER outperforms traditional instruction tuning. Notably, we see a 30 points increase in F1-score for the model identifying the *Referential-terms*. This observation is similar to NER, where the performance of LLMs improve identifying the token spans that are mostly context dependent. Similarly, for the ABSA task, we see that SUNDER outperforms traditional instruction-tuned models, identifying both *aspect* and *opinion* spans, with about 11 points of performance improvement on *opinion* spans. Of the two span types, opinion spans are the most context dependent as they can be expressed implicitly and may require surrounding words to be inferred. The per label performance of the LLMs across tasks suggests that adding bidirectionality improves the understanding capabilities of the LLMs and improves the performance of these models on context dependent span identification. Further analysis of how the attention computations vary across vanilla instruction-tuned models and SUNDER tuned models is provided in Appendix A.4.

6.2 In-Context Learning

Approach	CoNLL	Wiki	DEF	ABSA
0-shot _{Vanilla}	0.508	0.193	0.119	0.566
0-shot _{SUNDER}	0.492	0.195	0.124	0.561
3-shot _{Vanilla}	0.661	0.601	0.427	0.641
3-shot _{SUNDER}	0.680	0.565	0.436	0.651
5-shot _{Vanilla}	0.699	0.632	0.437	0.663
5-shot _{SUNDER}	0.679	0.633	0.457	0.674
10-shot _{Vanilla}	0.710	0.662	0.494	0.658
10-shot _{SUNDER}	0.692	0.680	0.500	0.691

Table 3: Results (Micro F1) comparison between the ICL approaches with and without SUNDER. We use the Llama-8B model. Best scores are displayed in bold.

To evaluate the generalizability of SUNDER across different LLM approaches, we evaluate the Llama-8b model in in-context learning (ICL) settings.

Specifically, we evaluate this model in zero and few-shot settings. Unlike the instruction tuning approach, where we use the Llama-8b base model, for the ICL experiments, we use the Instruct model, which is instruction-tuned on a large general domain instruction dataset. For few-shot experiments, we provide the input prompt with 3, 5, and 10 examples. The examples are selected using a semantic similarity search strategy. Here, the top-n semantically similar examples to the input from the training dataset are used. We also add the `<start>` and `<end>` tokens to the input of each example to indicate that the input can access future tokens for attention computation.

As seen in Table 3, SUNDER outperforms vanilla ICL on three of the four datasets. In zero-shot, our approach marginally outperforms the vanilla approach on the Wiki and DEF datasets, whereas, in few-shot on average, SUNDER outperforms vanilla few-shot for Wiki, DEF, and ABSA, with 10-shot achieving the highest F1 score on these three datasets. Compared to ICL, instruction-tuning with SUNDER outperforms vanilla instruction-tuning on all datasets with 1-8 point overall gains. This performance disparity can be attributed to the attention computations. The models are pretrained autoregressively with a causal language model objective, wherein each token attends only to its preceding context. Consequently, the attention parameters and learned representations are optimized assuming strictly left-to-right information flow. Introducing bidirectional attention without additional fine-tuning may result in distribution shift between the pretraining and inference attention patterns, potentially leading to inaccurate representations. Another contributing factor to the lower performance gains for ICL with SUNDER may be the introduction of the `<start>` and `<end>` tokens. Since these tokens are initialized with the `<bos>` and `<eos>` embeddings respectively, including them in the prompt may lead to inaccurate attention patterns and even degraded performance as the generated output may contain irrelevant tokens. Additionally, analysis of the outputs indicates that with SUNDER, the model occasionally generates partial spans or a single span is split into multiple sub-spans for some tasks like CoNLL, which affects the F1-score. To improve the performance of the models in ICL setting with SUNDER, these models need to be retrained with partial unmasking to allow bidirectional understanding. We aim to explore this in future work.

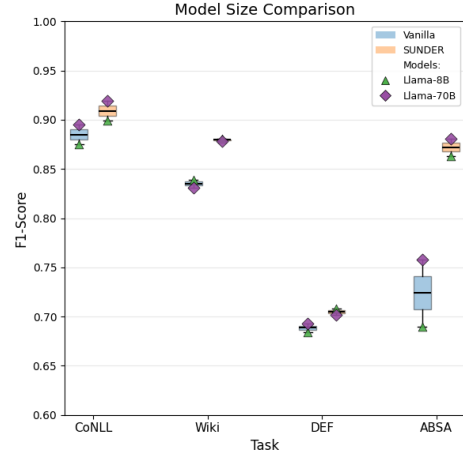


Figure 3: Model size comparison across model families.

6.3 Comparison Across Model Sizes

Dmonte et al. (2025) show that for the task of span identification, the LLM size only slightly improves the performance. To evaluate if SUNDER performs better with larger models, we instruction-tune the larger LLMs and compare the F1-scores to their smaller counterparts. Similar to (Dmonte et al., 2025), we experiment with Llama model family, due to availability of larger models. Due to lack of comparably sized model variants, Mistral and Qwen3 models were excluded. We use the Llama-3.1-70B model. We use the same training configuration and hyperparameters as described in Section 4.2.

Figure 3 shows the performance of different models with vanilla instruction tuning and instruction tuning with SUNDER. As seen, with the vanilla instruction tuning approach, on all four datasets, the larger Llama model outperforms its smaller counterpart. As with our approach, we observe a similar trend on three of the four datasets. This observation aligns with the findings in (Dmonte et al., 2025) that larger models don’t significantly improve the performance on span identification tasks. Dukić and Šnajder (2024) in their work, also perform a similar experiment with model scale. However, they observe that removing the causal mask degrades the performance of the larger models compared to their smaller counterparts. This shows that removing the entire causal mask affects the model’s performance. Unlike their approach, we only remove the causal mask for the main input. From Figure 3, we see that our approach is size agnostic and it outperforms both small and large vanilla instruction-tuned models. The results

Language	Example Instance	English Translation	Train	Dev	Test	Total
DE	Oktober 1938 in Viareggio LOCATION, begann seine Spielerkarriere beim FC Bologna ORGANIZATION.	In October 1938 in Viareggio LOCATION, he began his playing career at FC Bologna ORGANIZATION.	3,000	300	300	3,600
ES	Juega como centrocampista y su actual equipo es el Club Deportivo Castellón ORGANIZATION de la Segunda División de España MISCELLANEOUS.	He plays as a midfielder and his current team is Club Deportivo Castellón ORGANIZATION, of the Spanish Second Division MISCELLANEOUS.	3,000	300	300	3,600
FR	“Nous Autres” MISCELLANEOUS inspira à George Orwell PERSON le contexte de son roman “1984”.	“We” MISCELLANEOUS inspired George Orwell PERSON in developing the context for his novel “1984”.	3,000	300	300	3,600
IT	Nel 2011 torna in TV nella fiction di Rai 1 ORGANIZATION “Che Dio ci aiuti” MISCELLANEOUS, al fianco di Elena Sofia Ricci PERSON.	In 2011, she returned to television in the Rai 1 ORGANIZATION series “Che Dio ci aiuti” MISCELLANEOUS, alongside Elena Sofia Ricci PERSON.	3,000	300	300	3,600
HI	भारत LOCATION में 29 लाख बच्चे खसरे के टीके की पहली खुराक से वंचित: यूनिसेफ ORGANIZATION	In India LOCATION, 2.9 million children missed the first dose of the measles vaccine: UNICEF ORGANIZATION.	3,500	300	300	4,100
SI	9. MISCELLANEOUS ගල්කිස්සා LOCATION මහේස්ත්‍රාත් අධිකරණයේ ORGANIZATION වැසිකිළිය අලුත්වැඩියා කිරීම	9. MISCELLANEOUS Renovation of the toilet at the Mount Lavinia LOCATION Magistrate’s Court ORGANIZATION.	2,410	300	300	3,010

Table 4: Overview of the multilingual named entity recognition datasets, with annotated example instances, corresponding English translations, and data splits. Entity annotations are retained in the English translations.

further demonstrate that introducing partial bidirectionality improves the model’s understanding.

7 Generalizability Analysis

7.1 Insights on Multilingual NER

Languages, Dataset, and Models To evaluate SUNDER’s performance on languages other than English, we carry out a multilingual NER experiments on four high-resource languages (German, French, Italian, and Spanish) from WikiNeural (Tedeschi et al., 2021) as described in Section 4.1. Additionally, we also experiment with Hindi and Sinhala, which are considered medium and low-resource languages, respectively (Ranasinghe et al., 2025a,b). For Hindi, we use the Naamapadam (Mhaske et al., 2023) dataset, while for Sinhala, we use the multiNER (Ranathunga et al., 2025) dataset. Table 4 provides dataset statistics for each language. We explore the Llama-3.1-8B and Qwen3-8B models due to their multilingual capabilities. Similar to our English experiments, we compare the performance of our approach with traditional instruction-tuning.

Results Table 5 shows the performance of the models. Comparing the LLMs, Qwen outperforms the Llama model on three of the six languages. This can be attributed to the enhanced multilingual support and high quality training data (Yang et al., 2025). Between vanilla instruction-tuning and SUNDER, we see improved performance on all six languages. On five of the six languages, we see

an average gain of 2.8 points with up to 16 points of performance gains on Hindi.

Model	ES	DE	FR	IT	HI	SI
Qwen3-8B _{vanilla}	0.938	0.928	0.914	0.897	0.627	0.635
Llama-8B _{vanilla}	0.931	0.930	0.925	0.900	0.755	0.625
Qwen3-8B _{SUNDER}	0.942	0.942	0.929	0.930	0.787	0.644
Llama-8B _{SUNDER}	0.953	0.931	0.952	0.894	0.774	0.666

Table 5: Results on Multilingual NER. We evaluate six languages, German (DE), Spanish (ES), French (FR), Italian (IT), Hindi (HI) and Sinhala (SI). Best scores for each language are displayed in bold.

Analysis of the outputs indicates that with SUNDER, the model generates partial spans for some entities. Additionally, between the two models, we see lower performance in identifying some entities. The disparities in the performance of different models can be due to the language specific capabilities of these models. The bidirectional attention may affect the learned representation differently for different languages and models. Overall, the results show the generalizability of SUNDER in multilingual setting. The observations also suggest that for non-English languages, SUNDER may work better with certain models while it may achieve comparable performance with others. The performance of the models in non-English languages can be further improved by using language-specific LLMs.

7.2 Insights on SQUAD

Dataset and Models To evaluate the performance of SUNDER in tasks other than token spans

identification, we experiment with the SQUAD (Rajpurkar et al., 2016) QA dataset. For these tasks, we enclose both the questions and the context text within the `<start>` and `<end>` tokens. Similar to the token spans experiments, we use the Llama-8B, Mistral-7B, and Qwen3-8B models for these experiments.

Approach	Llama-8B	Mistral-7B	Qwen3-8B
Vanilla	0.743	0.758	0.685
SUNDER	0.824	0.818	0.822

Table 6: Results on the SQUAD dataset. Best scores are displayed in bold.

Results Table 6 shows the F1-scores of the three models for vanilla instruction-tuning and SUNDER. Across the models, Mistral outperforms both Llama and Qwen models, while for SUNDER, Llama achieves the best F1-score. Comparing vanilla instruction-tuning and SUNDER, we see that SUNDER improves the performance of all three models by up to 14 points demonstrating the SUNDER’s generalizability across different text understanding tasks, showing consistent gains across models.

8 Conclusion

In this paper, we introduce SUNDER, a novel approach that enables partial bidirectionality in decoder-only models by selectively modifying the causal mask to allow specific tokens to attend to future context during attention computation. Unlike previous approaches that remove the causal mask entirely or modify all attention layers, SUNDER selectively unmask only the input text while preserving the autoregressive property during output generation. For English, we validate our approach on four token span identification datasets: CoNLL, Wiki, DEF, and ABSA. We instruction-tune three LLM families, Llama, Qwen, and Mistral, using both traditional instruction tuning (with causal masking) and SUNDER.

We revisit the research question posed in the introduction: *Does selectively removing the causal mask from decoder-only models bring performance improvement to vanilla LLMs?* SUNDER delivers F1-score improvements across all tasks over vanilla instruction tuning. We further perform several ablations to evaluate if the performance gains translate across different LLM approaches and model sizes. Experimental results indicate that SUNDER

also show performance gains on larger models while we see performance improvement in ICL setting on three of the four evaluated datasets. Finally, we also show that SUNDER works in multilingual setting on some of the multilingual LLMs.

Future work will extend SUNDER to enhance its performance in ICL settings through (a) exploring complementary training strategies; (b) carrying out experiments on additional model architectures; and (c) exploring a wider set of languages, in particular, low-resource languages.

Limitations

We evaluate open-source models on SUNDER, since the model details for other proprietary models like GPT-5 are unavailable. Furthermore, we evaluate three model families. While we pose that this is enough to empirically demonstrate the effectiveness of the proposal approached, it would be interesting to see how SUNDER would perform with other models.

Ethical Considerations

All datasets used in this work are publicly available with established research usage permissions. All datasets have been previously published and evaluated by the research community. We acknowledge that these datasets may contain biases reflective of their source domains and the time periods in which they were collected. Models trained on these datasets may inadvertently learn and reproduce such biases.

Furthermore, the computational requirements of our approach (requiring multiple A100 GPUs for training) may limit accessibility for researchers with fewer resources. We have released our code to promote reproducibility, but we acknowledge that computational barriers remain.

Finally, researchers applying SUNDER to new datasets and/or domains are encouraged to evaluate potential biases in their specific contexts and consider the ethical implications in their intended applications, particularly in sensitive domains such as healthcare and law.

Acknowledgements

Tharindu Ranasinghe is funded by COST Action CA23147 GOBLIN – Global Network on Large-Scale, Cross-domain and Multilingual Open Knowledge Graphs, supported by COST (European Cooperation in Science and Technology).

References

- Anum Afzal, Florian Matthes, and Alexander Fabbri. 2025. DA-pred: Performance prediction for text summarization under domain-shift and instruct-tuning. In *EMNLP*.
- Parishad BehnamGhader, Vaibhav Adlakha, Marius Mosbach, Dzmitry Bahdanau, Nicolas Chapados, and Siva Reddy. 2024. Llm2vec: Large language models are secretly powerful text encoders. In *COLM*.
- Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. In *NeurIPS*.
- Zhuang Chen and Tieyun Qian. 2020. Relation-aware collaborative learning for unified aspect-based sentiment analysis. In *ACL*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*.
- Alphaeus Dmonte, Roland Oruche, Tharindu Ranasinghe, Marcos Zampieri, and Prasad Calyam. 2025. Exploring the performance of large language models on subjective span identification tasks. In *IJCNLP-AACL*.
- David Dukić and Jan Šnajder. 2024. Looking right is sometimes right: Investigating the capabilities of decoder-only llms for sequence labeling. In *ACL Findings*.
- Ridong Han, Tao Peng, Chaohao Yang, Benyou Wang, Lu Liu, and Xiang Wan. 2023. Is information extraction solved by chatgpt? an analysis of performance, evaluation criteria, robustness and errors. *arXiv*.
- Momchil Hardalov, Pepa Atanasova, Todor Mihaylov, Galia Angelova, Kiril Simov, Petya Osenova, Veselin Stoyanov, Ivan Koychev, Preslav Nakov, and Dragomir Radev. 2023. bgglue: A bulgarian general language understanding evaluation benchmark. In *ACL*.
- Antonia Höfer and Mina Mottahedin. 2023. Minanto at semeval-2023 task 2: Fine-tuning xlm-roberta for named entity recognition on english data. In *SemEval*.
- Yan Cathy Hua, Paul Denny, Jörg Wicker, and Katerina Taskova. 2024. A systematic review of aspect-based sentiment analysis: domains, methods, and trends. *Artificial Intelligence Review*.
- Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. IndicNLP Suite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages. In *EMNLP Findings*.
- Shahar Katz, Liran Ringel, Yaniv Romano, and Lior Wolf. 2025. Segment-based attention masking for gpts. In *ACL*.
- Navjeet Kaur, Ashish Saha, Makul Swami, Muskan Singh, and Ravi Dalal. 2024. Bert-ner: A transformer-based approach for named entity recognition. In *ICCCNT*.
- Aobo Kong, Shiwan Zhao, Hao Chen, Qicheng Li, Yong Qin, Ruiqi Sun, and Xiaoyan Bai. 2023. PromptRank: Unsupervised keyphrase extraction using prompt. In *ACL*.
- John D Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*.
- Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2025. Nv-embed: Improved techniques for training llms as generalist embedding models. In *ICLR*.
- Yuanyuan Lei and Ruihong Huang. 2025. Multi-document summarization through multi-document event relation graph reasoning in LLMs: a case study in framing bias mitigation. In *ACL*.
- Xianming Li and Jing Li. 2024. Bellm: Backward dependency enhanced large language model for sentence embeddings. In *NAACL*.
- Zongxi Li, Xianming Li, Yuzhang Liu, Haoran Xie, Jing Li, Fu-lee Wang, Qing Li, and Xiaoqin Zhong. 2023. Label supervised llama finetuning. *arXiv*.
- Caleb Martin, Huichen Yang, and William Hsu. 2022. Kddie at semeval-2022 task 11: Using deberta for named entity recognition. In *SemEval*.
- Arnav Mhaske, Harshit Kedia, Sumanth Doddapaneni, Mitesh M Khapra, Pratyush Kumar, Rudra Murthy, and Anoop Kunchukuttan. 2023. Naamapadam: A large-scale named entity annotated data for indic languages. In *ACL*.
- Dang Minh, H Xiang Wang, Y Fen Li, and Tan N Nguyen. 2022. Explainable artificial intelligence: a comprehensive review. *Artificial Intelligence Review*.
- Fengran Mo, Yifan Gao, Chuan Meng, Xin Liu, Zhuofeng Wu, Kelong Mao, Zhengyang Wang, Pei Chen, Zheng Li, Xian Li, and 1 others. 2025. Uni-Conv: Unifying retrieval and response generation for large language models in conversations. In *ACL*.
- Victor Morand, Nadi Tomeh, Josiane Mothe, and Benjamin Piwowarski. 2026. Tommer-efficient entity mention detection from large language models. *ACL*.
- Niklas Muennighoff, SU Hongjin, Liang Wang, Nan Yang, Furu Wei, Tao Yu, Amanpreet Singh, and Douwe Kiela. 2025. Generative representational instruction tuning. In *ICLR*.

- Roland Oruche, Sai Keerthana Goruganthu, Rithika Akula, Xiyao Cheng, Ashraf Md Goni, Bruce W Shibo, Kerk Kee, Marcos Zampieri, and Prasad Calyam. 2025. A survey on the recent advancements in human-centered dialog systems. *ACM Computing Surveys*, 57(10):1–36.
- Roland Oruche, Marcos Zampieri, and Prasad Calyam. 2024. Deep contrastive active learning for out-of-domain filtering in dialog systems. In *DSAA*.
- Sean Papay, Roman Klinger, and Sebastian Padó. 2020. Dissecting span identification tasks with performance prediction. In *EMNLP*.
- John Pavlopoulos, Jeffrey Sorensen, Léo Laugier, and Ion Androutsopoulos. 2021. SemEval-2021 task 5: Toxic spans detection. In *SemEval*.
- Alistair Plum, Felicia Körner, Anne-Marie Lutgen, Laura Bernardy, Fred Philippy, Emilia Milano, Nils Rehlinger, Cedric Lothritz, Tharindu Ranasinghe, Barbara Plank, and Christoph Purschke. 2026. ItzGLUE: Luxembourgish general language understanding evaluation. In *ACL Findings*.
- Soham Poddar, Biswajit Paul, Moumita Basu, and Saptarshi Ghosh. 2024. Icp 2024 competition on multilingual claim-span identification. In *ICPR*.
- Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 task 4: Aspect based sentiment analysis. In *SemEval*.
- Nicholas Popović and Michael Färber. 2024. Embedded named entity recognition using probing classifiers. In *EMNLP*.
- Renyi Qu, Ruixuan Tu, and Forrest Sheng Bao. 2025. Is semantic chunking worth the computational cost? In *NAACL Findings*.
- Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *EMNLP*.
- Tharindu Ranasinghe, Hansi Hettiarachchi, Constantin Orasan, and Ruslan Mitkov. 2025a. MUSTS: Multilingual semantic textual similarity benchmark. In *ACL*.
- Tharindu Ranasinghe, Hansi Hettiarachchi, Nadeesha Chathurangi Naradde Vidana Pathirana, Damith Premasiri, Lasitha Uyangodage, Isuri Nanomi Arachchige, Alistair Plum, Paul Rayson, and Ruslan Mitkov. 2025b. Sinhala encoder-only language models and evaluation. In *ACL*.
- Tharindu Ranasinghe and Marcos Zampieri. 2021. Mudes: Multilingual detection of offensive spans. In *NAACL*.
- Surangika Ranathunga, Asanka Ranasinghe, Janaka Shamal, Ayodya Dandeniya, Rashmi Galappaththi, and Malithi Samaraweera. 2025. A multi-way parallel named entity annotated corpus for english, tamil and sinhala. *NLP Journal*.
- Erik Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *HLT-NAACL*.
- Wei Liang Seow, Iti Chaturvedi, Amber Hogarth, Rui Mao, and Erik Cambria. 2025. A review of named entity recognition: from learning methods to modelling paradigms and tasks. *Artificial Intelligence Review*.
- Sasha Spala, Nicholas Miller, Franck Dernoncourt, and Carl Dockhorn. 2020. Semeval-2020 task 6: Definition extraction from free text with the defct corpus. In *SemEval*.
- Jacob Mitchell Springer, Suhas Kotha, Daniel Fried, Graham Neubig, and Aditi Raghunathan. 2025. Repetition improves language model embeddings. In *ICLR*.
- Simone Tedeschi, Valentino Maiorca, Niccolò Campolungo, Francesco Cecconi, and Roberto Navigli. 2021. Wikineural: Combined neural and knowledge-based silver data creation for multilingual ner. In *EMNLP Findings*.
- Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. In *NeurIPS*.
- Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *BlackboxNLP*.
- Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, Guoyin Wang, and Chen Guo. 2025. Gpt-ner: Named entity recognition via large language models. In *NAACL Findings*.
- Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2016. Recursive neural conditional random fields for aspect-based sentiment analysis. In *EMNLP*.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Ly, and 1 others. 2025. Qwen3 technical report. *arXiv*.
- Haoran Yang, Yumeng Zhang, Jiaqi Xu, Hongyuan Lu, Pheng-Ann Heng, and Wai Lam. 2024. Unveiling the generalization power of fine-tuned large language models. In *NAACL*.
- Marcos Zampieri, Skye Morgan, Kai North, Tharindu Ranasinghe, Austin Simmons, Paridhi Khandelwal, Sara Rosenthal, and Preslav Nakov. 2023. Target-based offensive language identification. In *ACL*.

A Appendix

A.1 Additional Details

In this section we provide further details on how SUNDER differs from the traditional causal language modeling. As described in 3, the only difference between traditional causal language modeling and SUNDER is in the causal mask used during attention computation. The causal mask used in traditional causal language modeling restricts the tokens in the input sequence from attending to the future tokens. In contrast, SUNDER changes this causal mask such that the tokens in the main input can attend to the future tokens of the input.³ This change in the causal mask changes the representation of the hidden states. In the transformer architecture, the output hidden state representations of a specific self-attention layer are propagated through the subsequent layers in the attention block. The hidden state representation of the final token from the final self-attention layer is then used to predict the next token in the sequence. These representations generated with the traditional causal mask depend only on the context to the left of each token. Conversely, with SUNDER, the representations change as the causal mask is changed to include future context for some tokens. This is because the attention scores computed in each self-attention layer with SUNDER are different compared to the attention scores computed with traditional causal mask. Although the entire input sequence is available to the model at the last token for both vanilla causal language modeling and SUNDER, the change in attention scores propagated through the self-attention layers and subsequently the hidden state representation of the final token generates different outputs.

A.2 Task Prompts

Table 7 shows the task-specific prompts used in our experiments. The prompt contains a task-specific instruction, where, the task definition along with any additional terminology. These instructions are followed by the output format and the main input. It is to be noted that we use a similar prompt for traditional instruction-tuning. Only the `<start>` and `<end>` tokens are excluded. Moreover, the same prompts are used for ICL experiments. For few-shot experiments, examples are included before the input.

³The input here indicates only the text from which the spans are to be identified and not the entire input prompt with the instructions.

A.3 Training and Inference Analysis

SUNDER outperforms traditional instruction tuning on most tasks, but it may also incur additional computational cost. Hence we compare the training and inference cost of the traditional instruction tuning the LLMs and instruction tuning using SUNDER. Every 100 steps in traditional instruction tuning are completed in about 3 minutes, while it takes on an average 5.5 minutes with SUNDER. This additional training time can be attributed to the additional computations required with SUNDER. During inference, the output per instance is generated in about 1.2 seconds, while with our approach, the output is generated in about 1.6 seconds. This shows a slight increase in inference time. This suggest that while SUNDER gives performance gains, there is computational overhead that needs to be considered with this approach.

A.4 Attention Score Analysis

To better understand the effect of SUNDER on the models attention computation, we visualize the attention weights of the main input tokens across three self-attention layers for vanilla instruction-tuned and SUNDER tuned models. Specifically, we use the fine-tuned Llama models for NER. We visualize the layers 8, 16, and 24 of the models. As seen in Figure 5, for SUNDER, we see a distinctive triangular attention pattern within the main input span, where the tokens attend to the future tokens as well. While there is no such inherent pattern visible for the vanilla instruction-tuned model. Notably, tokens such as "New" attend to future tokens "York" and "City" that are required to precisely identify "New York City" as a location entity. The bidirectional attention patterns are more pronounced in middle layers 8 and 16, and weakens in the latter layers. This suggest that selective unmasking predominantly changes the token representations in the middle layers that are responsible for semantic understanding.

To further quantify how SUNDER changes hidden representations relative to the vanilla instruction-tuned models, we compute layer-wise CKA similarity between the two models on 100 held-out NER examples, measuring the hidden state similarity of the main input tokens. We plot the similarity scores and standard deviation across the hidden layers as shown in Figure 6. Layer 0 shows a lower initial similarity indicating the effects of the introduced by the delimiter `<start>` and

`<end>` tokens. We further observe an increase in similarity through the early layers, with the maximum similarity obtained around the middle layers. This indicates that both vanilla and SUNDER tuned models build similar representations in these initial layers despite the causal mask difference. The similarity declines steadily beyond layer 16, with the lowest similarity obtained at the final layer, indicating the representational differences between vanilla and SUNDER models. These observations are consistent with the findings from the attention visualizations, suggesting that the representations progressively diverge across the layers with the bidirectional attention patterns accumulating through the middle layers and producing representational difference in the latter layers. The performance improvements observed with SUNDER can be attributed to the representational difference in the final layers which inform the output generation.

Dataset	Prompt
NER	For the given text, identify the named entities. The named entities can belong to one of the following categories: person, location, organization, and miscellaneous name. Output Format: <entity category>: <entity span> Input: {input text} Response:
ABSA	For the given text, identify the aspects and opinions. • Aspect: The entities to which sentiments are tied to. • Opinion: The sentiment words or phrases. Output Format: Aspect n: <nth aspect> Opinion n: <nth opinion> Input: {input text} Response:
DEF	For the given text, identify the term and definition spans. The span can belong to one of the following categories: qualifier, definition, ordered definition, referential term, term frag, alias term frag, definition frag, alias term, secondary definition, term, referential definition, or ordered term. A span can belong to only one these categories. The text may or may not have any terms or definitions. Output Format: <term-definition category>: <term-definition span> Input: {input text} Response:

Table 7: The task-specific prompts used to instruction-tune the LLMs with the Traditional Approach (Causal LM).

Dataset	Prompt
NER	For the given text, identify the named entities. The named entities can belong to one of the following categories: person, location, organization, and miscellaneous name. Output Format: <entity category>: <entity span> Input: <start>{input text}<end> Response:
ABSA	For the given text, identify the aspects and opinions. • Aspect: The entities to which sentiments are tied to. • Opinion: The sentiment words or phrases. Output Format: Aspect n: <nth aspect> Opinion n: <nth opinion> Input: <start>{input text}<end> Response:
DEF	For the given text, identify the term and definition spans. The span can belong to one of the following categories: qualifier, definition, ordered definition, referential term, term frag, alias term frag, definition frag, alias term, secondary definition, term, referential definition, or ordered term. A span can belong to only one these categories. The text may or may not have any terms or definitions. Output Format: <term-definition category>: <term-definition span> Input: <start>{input text}<end> Response:

Table 8: The task-specific prompts used to instruction-tune the LLMs using SUNDER.

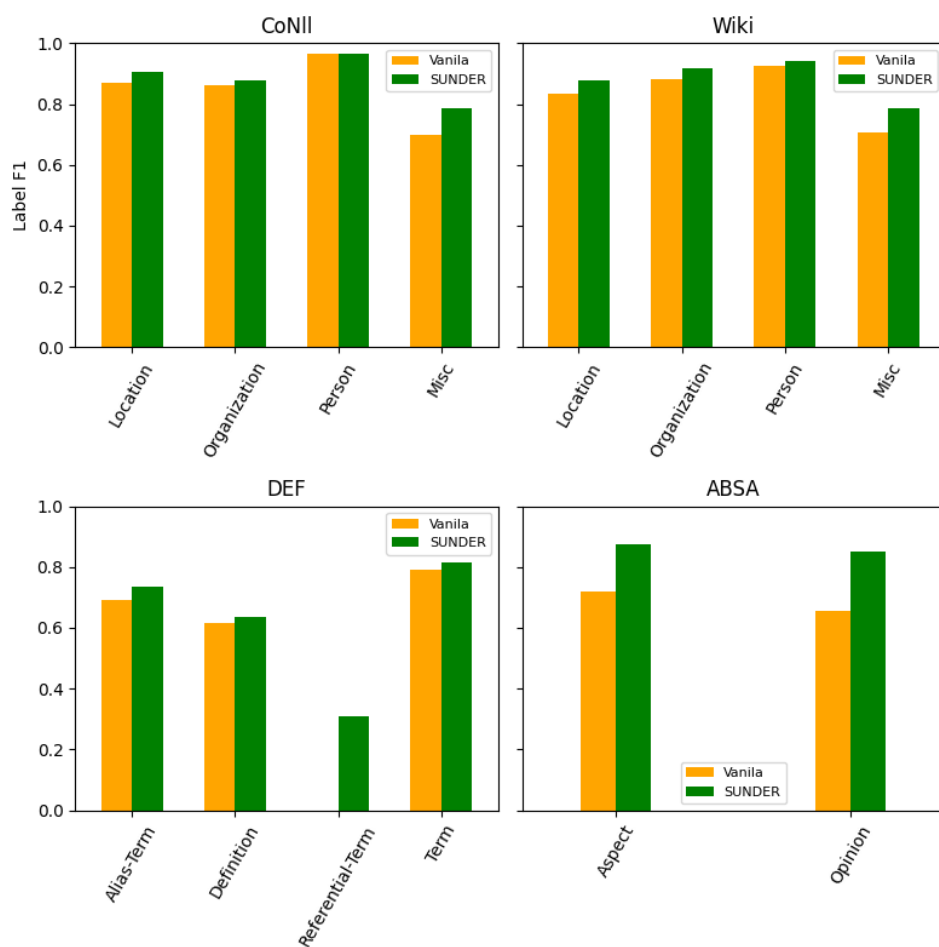


Figure 4: Per label F1-score comparison between traditional fine tuning and SUNDER on Llama-8B model.

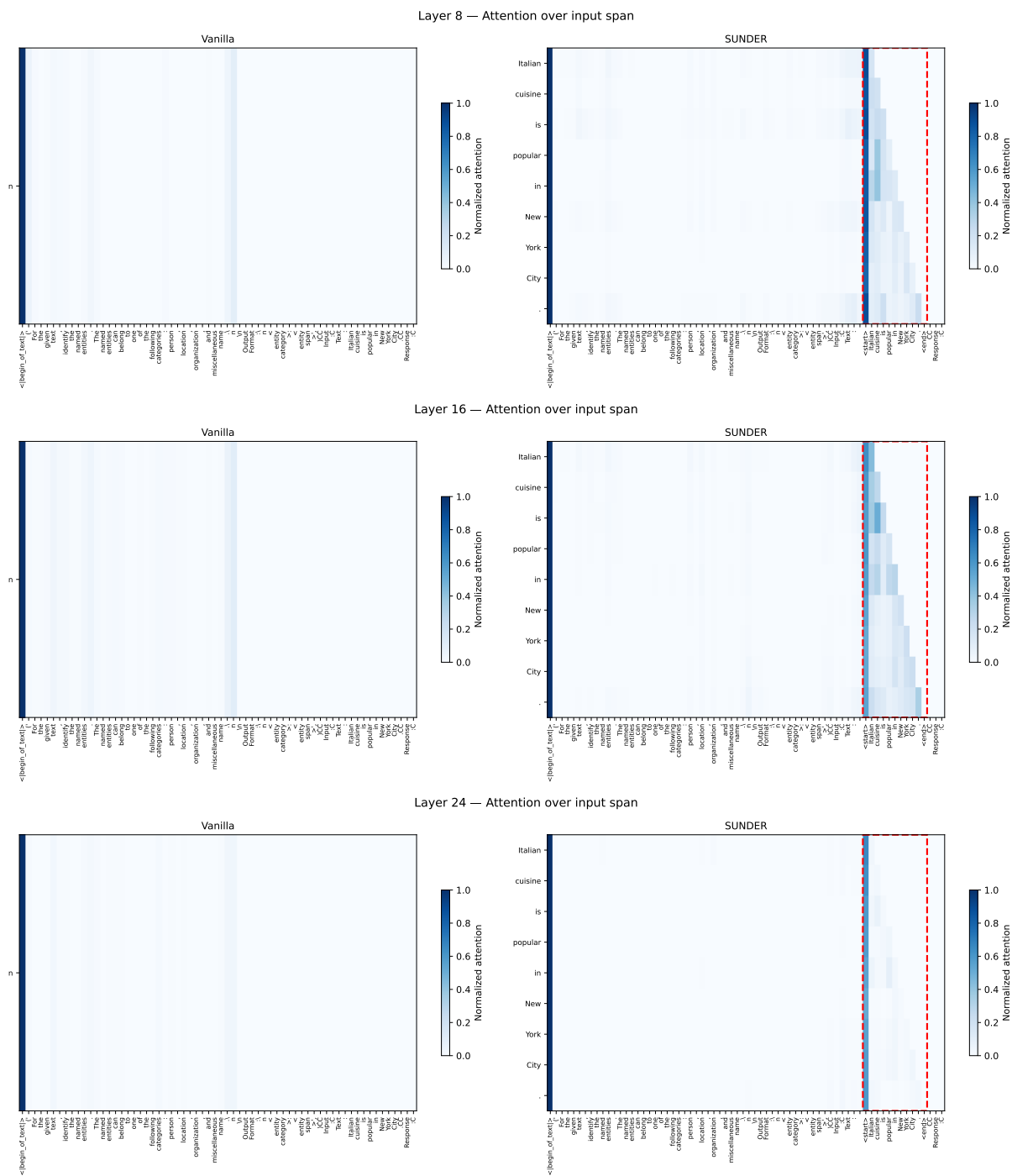


Figure 5: Attention weights comparison between vanilla instruction LLM and SUNDER tuned LLM.

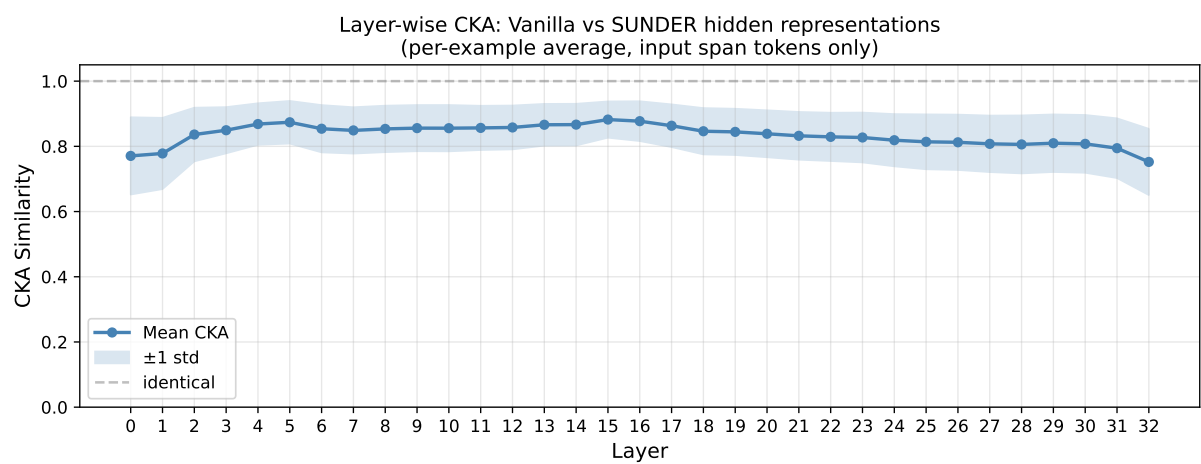


Figure 6: CKA similarity between the vanilla instruction LLM and SUNDER tuned LLM.