BiLSTM vs. BERT

Jan Vivo
IT University of Copenhagen
jviv@itu.dk

June 4, 2025

1 Abstract

The primary contribution of this work is to provide empirical information on the strengths and limitations of BiLSTM versus BERT for NER. These findings aim to guide researchers and practitioners in selecting appropriate models based on specific task requirements, data availability, and anticipated real-world conditions. This paper proceeds by outlining related work, detailing our methodology, presenting and analyzing the results, and finally, discussing their implications and concluding. The code used for this project can be found in our Github repository[1].

2 Introduction

Named Entity Recognition (NER) plays a vital role in various Natural Language Processing (NLP) tasks. Its purpose is to identify and categorize named entities such as locations, people, and organizations, etc. Prominent NER approaches include Bidirectional Long Short-Term Memory networks with a Conditional Random Field layer (BiLSTM) , known for strong sequential modeling and output tag dependency handling, and pre-trained Transformer models like BERT, which leverage massive unlabeled corpora and self-attention mechanisms for powerful contextual representations[2].

Although BERT often achieves state-of-the-art results due to its extensive pretraining, BiLSTM models offer a robust architecture tailored for sequence labeling, typically trained from scratch. These distinct architectural underpinnings and training paradigms motivate a direct comparison to understand their practical trade-offs. Key questions arise regarding their relative performance, efficiency when training data are scarce, and resilience to common input noise.

This project addresses these questions by investigating: *'How do BiLSTM and BERT compare in terms of performance, data efficiency, and robustness when applied to NER tasks?'* We conducted an empirical study on the WNUT_17 NER dataset, evaluating both model types using accuracy, precision, recall, and F1 score. Our comparison spans three dimensions: (1) overall performance in the entire dataset; (2) data efficiency, assessed by training on varying subsets of the training data; (3) robustness, tested by introducing typos and unseen words into the evaluation set.

3 Background

The task of Named Entity Recognition (NER) has been a central focus in NLP, with various modeling approaches evolving over time. This section briefly reviews the two architectures at the core of our comparative study: BiLSTM and BERT.

Bidirectional Long Short-Term Memory (BiLSTM) networks, often coupled with a Conditional Random Field (CRF) layer, have been a cornerstone for strong NER performance [4] BiLSTMs process input sequences from both forward and backward directions, allowing each token's representation to capture information from its entire context within the sequence. The subsequent CRF layer models dependencies between output tags, considering the labels of neighboring tokens to produce a globally optimal tag sequence, which is particularly beneficial for structured prediction tasks like NER. This architecture effectively learns features from the input and handles sequence-level tag constraints.

The birth of Transformer-based models, particularly BERT (Bidirectional Encoder Representations from Transformers), marked a paradigm shift in NLP. BERT utilizes a multi-layer bidirectional Transformer encoder that processes input sequences through self-attention mechanisms, allowing it to weigh the importance of different words when creating representations. Its core strength lies in its pre-training phase on massive unlabeled text corpora (e.g., Wikipedia, BooksCorpus). These pre-trained models can then be fine-tuned for specific downstream tasks, including NER, by adding

[1] https://github.com/Viul1488/NLP-2025

a task-specific output layer, often achieving impeccable results with significantly less task-specific labeled data compared to models trained from scratch.

While both BiLSTM and BERT have demonstrated strong capabilities for NER, they operate on different principles. BiLSTMs excel at explicit sequence modeling and tag-level dependencies, whereas BERT leverages large-scale pretraining for rich contextual understanding. Comparative studies often focus on overall performance, but a nuanced understanding of their behavior concerning data efficiency—how performance scales with varying amounts of training data—and robustness to common input perturbations (e.g., typos, unseen words, domain shifts) is less comprehensively explored in a direct head-to-head manner for NER. This study aims to contribute to this understanding by empirically evaluating these two prominent architectures across these critical dimensions.

4 Methodology

This section outlines the empirical framework established to compare the BiLSTM and BERT architectures for the task of Named Entity Recognition. We articulate the specifics of the model architectures, the datasets employed, the experimental configuration including training protocols and evaluation metrics, and the distinct dimensions, performance, data efficiency, and robustness that underlie our comparative analysis.

To achieve a truly comprehensive evaluation of Named Entity Recognition (NER) models, we strategically employ two contrasting datasets: CoNLL-2003 and WNUT 2017. This dual-dataset approach provides a broad understanding of model capabilities under various conditions.

4.1 Conll2003

The CoNLL-2003 dataset, acts as our benchmark for baseline performance in controlled environments. The dataset mostly consists of clean and formal texts and together with it's four well-defined entity types (person, organization, location, and miscellaneous), allows us to evaluate how consistently models identify well-known entities with minimal noise.

4.2 WNUT-17

In sharp contrast, the WNUT 2017 dataset is more suited for testing robustness in noisy, real-world conditions. Comprising user-generated content from platforms like Reddit and Twitter, it features six entity types (person, location, corporation, product, creative-work, and group) and emphasizes emerging or rare entities. This dataset is crucial for assessing a model's ability to generalize and perform well in informal, dynamic text, which is critical for practical applications[1].

4.3 BERT

Our first model for comparison is BERT (Bidirectional Encoder Representations from Transformers), specifically the bert-base-uncased variant. This multi-layer bidirectional Transformer encoder's strength comes from its self-supervised pre-training on BookCorpus and English Wikipedia. This pre-training uses Masked Language Modeling (MLM) to learn contextual representations by predicting masked tokens, and Next Sentence Prediction (NSP) to understand sentence relationships. Being ”uncased,” it ignores letter case and accents. The bert-base-uncased model features 12 layers, a 768-dimension hidden size, 12 attention heads, and 110 million parameters.

For the NER task, we fine-tune this pre-trained model. Input sentences are tokenized using BertTokenizerFast, which employs a 30,000-token WordPiece vocabulary and adds [CLS] and [SEP] special tokens. These tokenized sequences pass through BERT's encoder to get contextualized hidden states. A task-specific linear classification layer then maps each token's 768-dimensional output to NER tags, with a softmax activation.

Due to subword tokenization, a label alignment strategy is used: only the first subword of an original word receives its true NER label, while subsequent subwords are ignored by the loss function (assigned -100). During fine-tuning, all BERT parameters and the classification layer are jointly optimized on the labeled NER dataset[3].

4.4 BILSTM

The second model, a BiLSTM, combines the contextual sequence modeling strength of BiLSTMs with the structured prediction capabilities of a CRF layer. This architecture is trained from scratch on the task-specific NER data. Our implementation comprises:

An initial embedding layer, which transforms input tokens from the vocabulary into 100-dimensional dense vector representations. The vocabulary, including special tokens such as PAD and UNK, is constructed from the training corpus.

This is succeeded by a core BiLSTM layer, where the sequence of word embeddings is processed bidirectionally. Each LSTM direction utilizes a hidden state dimension of 50. The hidden states from both the forward and backward passes at each time step are concatenated.

The concatenated BiLSTM outputs are then fed into a linear layer that projects these representations into the NER tag space, producing emission scores for each tag at each token position.

Finally, a Conditional Random Field (CRF) layer is applied on top of these emission scores. The CRF layer learns transition probabilities between adjacent tags and considers the entire sequence of tags to find the globally optimal tag sequence. This is particularly advantageous for NER, where tag dependencies are crucial. The CRF layer utilizes special indices for proper sequence modeling and loss calculation.

The model is trained by minimizing the negative log-likelihood of the true tag sequences given the emissions and transition scores from the CRF. During inference, the Viterbi algorithm is employed within the CRF layer to decode the most probable tag sequence[5].

5 Analysis

5.1 Performance on the WNUT-17 Dataset

The WNUT-17 dataset, with its prevalence of emerging entities and often sparse annotations, presented considerable challenges for both models, particularly for the BiLSTM-CRF.

BiLSTM Performance: Our BiLSTM model, trained from scratch, struggled immensely on the WNUT-17 dataset. The learning curve for BiLSTM on WNUT-17 is presented in Figure 1.

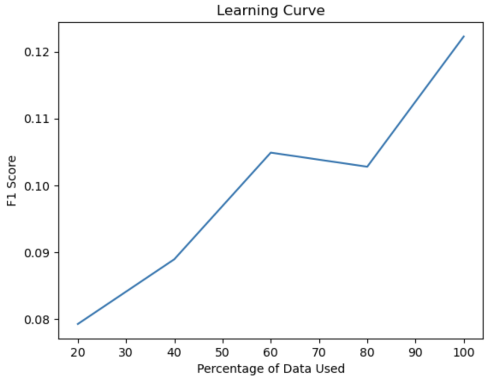

Figure 1: BiLSTM F1-Score Learning Curve on WNUT-17 Validation Set.

As shown in Figure 1, the Entity F1-scores for the BiLSTM model are exceptionally low across all data fractions. The F1-score starts at approximately 0.079 with 20% of the data, rises to around 0.09 with 40%, reaches a local peak of about 0.105 at 60%, dips slightly at 80% to around 0.103, and shows a final increase to approximately 0.124 with 100% of the training data. While there is a slight upward trend, the overall performance remains poor, indicating the model's profound difficulty in learning meaningful entity representations from this noisy dataset.

BERT Performance: In contrast, the pretrained BERT model proved to be more robust, though its performance was still modest compared to cleaner datasets. When trained on 100% of the WNUT-17 training data, BERT achieved an F1-score of 0.5915, with a precision of 0.6546, recall of 0.5395, and an accuracy of 0.9555. The learning curve for BERT on WNUT-17 is presented in Figure 2.

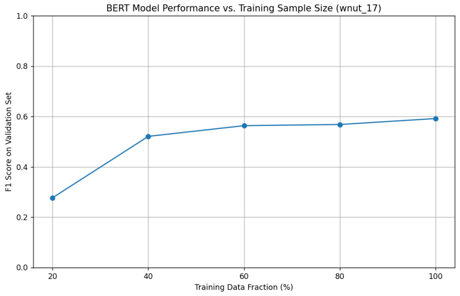

Figure 2: BERT F1-Score Learning Curve on WNUT-17 Validation Set.

The F1-score demonstrates a consistent upward trend, with diminishing returns as more training data is utilized.

5.2 Performance on the CoNLL-2003 Dataset

The CoNLL-2003 dataset, a more canonical benchmark, provided a different performance landscape. **BiLSTM Performance:** Our BiLSTM model performed decently on the CoNLL-2003 dataset. With 100% of the training data, it achieved an F1-score of 0.72. The learning curve for the BiLSTM model on CoNLL-2003 is shown in Figure 3.

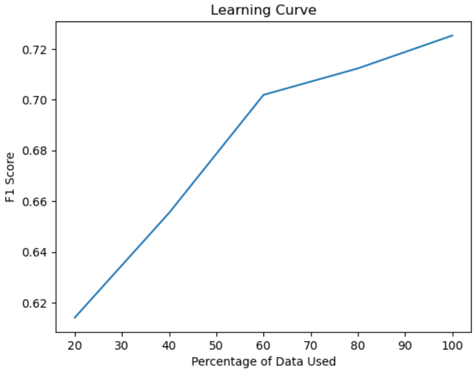

Figure 3 BiLSTM F1-Score Learning Curve on CoNLL-2003.

The model shows improvement with more data, generally plateauing after 60-80% of the data.

BERT Performance: On the CoNLL-2003 dataset, BERT demonstrated very strong performance. When trained on 100% of the training data, it achieved an F1-score (entity-level) of 0.9440. The learning curve for BERT on CoNLL-2003, presented in Figure 4, illustrates its data efficiency and high performance.

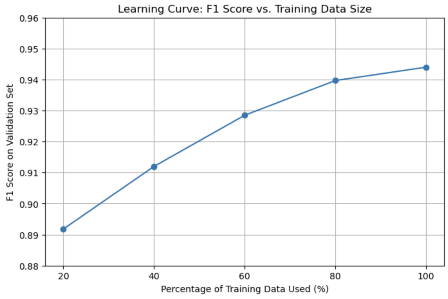

Figure 4: BERT F1-Score Learning Curve on CoNLL-2003 Validation Set.

The learning curve (Figure 5) shows that BERT quickly learns from the CoNLL-2003 data, achieving an F1-score of 0.8918 with just 20% of the training samples. The gains become more incremental with additional data, indicating high data efficiency and robust learning on this benchmark dataset, reaching a peak F1-score of 0.9440.

Comparative Discussion

- **Dataset Impact:**
  - On CoNLL-2003, a well-structured dataset, BERT achieves high F1-score of 0.9440, and BiLSTM achieves a respectable 0.72.
  - On WNUT-17, performance for both models dropped significantly. BERT managed an F1-score of 0.5915, while the BiLSTM reached only approximately 0.124 F1 with 100% data. The learning curve for BiLSTM on WNUT-17 (Figure 1) shows that even with increasing data, its ability to learn effectively is severely hampered. This starkly illustrates the difficulty WNUT-17 poses.
- **Data Efficiency Insights:**
  - BERT's learning curve on WNUT-17 (Figure 2) shows it benefits from more data, even if overall performance remains modest for this challenging dataset
  - The BiLSTM's learning curve on WNUT-17 (Figure 1) shows minimal improvement with data, never achieving a meaningful performance level. It's curve on CoNLL-2003 (Figure 3) is more typical of a traditional model, showing gains before plateauing.
  - BERT's learning curve on CoNLL-2003 (Figure 4) demonstrates exceptional data efficiency. It achieves a very high F1-score (0.8918) with only 20% of the training data and continues to improve, plateauing near its peak performance of 0.9440. This highlights BERT's ability to leverage its pre-training effectively on standard benchmark tasks.
- **Model Comparison:**
  - **BERT's Clear Advantage:** BERT consistently and substantially outperforms the BiLSTM. On CoNLL-2003, the F1-score gap is **0.9440** vs. 0.72. On WNUT-17, the difference is even more pronounced: 0.5915 for BERT versus 0̃.124 for BiLSTM.
  - **BiLSTM's Limitations:** While capable on cleaner datasets like CoNLL-2003 (Figure 3), the BiLSTM architecture trained from scratch is highly ineffective on noisy, specialized datasets like WNUT-17 (Figure 1).

  The consensus that pre-trained BERT is generally more robust and performs better is strongly supported by our findings. Its superiority is especially sharp on the challenging WNUT-17 dataset, and its data efficiency is prominent on CoNLL-2003.

Summary Table:

Dataset	Model	F1-Score
WNUT17	BERT	0.59
WNUT17	BiLSTM	0.124
CoNLL-2003	BERT	0.944
CoNLL-2003	BiLSTM	0.72

Table 1: Final Model Performance on Full Evaluation Data

6 Conclusion

This study highlights the strengths and limitations of BERT and BiLSTM for Named Entity Recognition across different data conditions. On the noisy, real-world WNUT-17 dataset, BERT significantly outperforms BiLSTM (F1-score 0.5915 vs. 0.124), thanks to its pre-trained contextual embeddings that offer strong robustness to input noise and sparse annotations. However, on cleaner and more structured datasets like CoNLL-2003, BiLSTM remains competitive, achieving reasonable performance with greater computational efficiency.

Ultimately, the choice between these models should depend on the specific application. BERT is well-suited for scenarios involving noisy or limited data, where its pre-training provides a distinct advantage. In contrast, BiLSTM offers a viable, lightweight alternative for tasks involving large amounts of clean, well-annotated text.

Future work could explore hybrid architectures that combine BERT's contextual understanding with BiLSTMs sequential modeling, or investigate domain-adaptive pretraining techniques to further enhance BERT's performance on niche areas. Additionally, assessing these models in multilingual NER settings could broaden their practical utility.

References

[1] Leon Derczynski, Eric Nichols, Marieke van Erp, and Nut Limsopatham. Results of the WNUT2017 shared task on novel and emerging entity recognition. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 140–147, Copenhagen, Denmark, 2017. Association for Computational Linguistics.

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

[3] Hugging Face. BERT Base Uncased Model. `https://huggingface.co/google-bert/bert-base-uncased`, 2024. Accessed: 23 May 2025.

[4] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, San Diego, California, June 2016. Association for Computational Linguistics.

[5] Nikolay Manchev. Named Entity Recognition (NER) Challenges and Model. `https://domino.ai/blog/named-entity-recognition-ner-challenges-and-model`, 2023. Accessed: 23 May 2025.

7 Appendix

7.1 Group contributions

- Earth Vangwithayakul: BiLSTM, Introduction, Paper formating and Analysis
- Zen Al Aabden Ammar Rehda : Abstract, Introduction, Background and Analysis
- Viktor Ulitin: BERT, Analysis and Reserch
- Jan Vivo Vivo: Conclusion, BERT and Model Testing

To be fair, everyone contributed and did their part.

Github: https://github.com/Viul1488/NLP_2025

7.2 AI usage

We utilized Gemini and Chatgpt for code snippets - Ai assistances were mainly used for fixing code-errors we ran into.