A ES
ISSN: 2096-3246
Volume 58, Issue 01, 2026

ENHANCING BIONER PERFORMANCE THROUGH ADVERSARIALLY-TRAINED BIOBERT AND TOKEN-LEVEL PRECISION STRATEGIES

Sushma Rani N 1, Dhawaleswar Rao CH2, Srinivasa Rao P3
1Computer Science and Engineering, Centurion University of Technology and Management, R. Sitapur, Odisha, India
2Computer Science and Engineering, Centurion University of Technology and Management, R. Sitapur, Odisha, India
3Computer Science and Engineering – Data Science-JNTUGV, MVGR College of Engineering(A), Vizianagaram, AP, India
sushma24583@gmail.com, dhawaleswarrao@gmail.com, psr.sri@gmail.com

Abstract— Biomedical Named Entity Recognition (BioNER) is vital for extracting structured information from vast amounts of unstructured biomedical text. This study introduces an enhanced BioNER approach by fine-tuning the BioBERT model using adversarial training techniques—specifically, Fast Gradient Sign Method (FGSM) and Projected Gradient Descent (PGD)—to bolster model robustness against input perturbations. By incorporating optimized token alignment strategies, the proposed method significantly improves the identification and classification of biomedical entities across multiple benchmark datasets, including MedMentions, BC5CDR, and i2b2 2010. Comprehensive evaluations using metrics such as Precision, Recall, F1-score, and Entity-Level Accuracy demonstrate that the model consistently surpasses current state-of-the-art systems. This work not only highlights the advantages of adversarial training for domain-specific language models but also sets a new standard for robust and accurate biomedical NER systems.

Key Terms— Biomedical Named Entity Recognition (BioNER.), BioBERT Fine-Tuning, Adversarial Training (FGSM, PGD), Token Alignment Strategies, Domain-Specific Language Models

1. INTRODUCTION

There's more and more unstructured text data floating around these days, thanks to the fast pace of biomedical research and the widespread use of digital medical records. Being able to pull out useful info from all this data is super important if we want to grow our scientific knowledge, speed up drug discovery, and make healthcare systems stronger. One key piece of this puzzle is Named Entity Recognition (NER), a part of Natural Language Processing (NLP) that helps find and categorize things like diseases, genes, and drugs right from the text[1]. While NLP has come a long way, the tricky nature of medical terms, overlapping entities, and all sorts of text formats make recognizing biological entities accurately a real challenge[2],[3].

In this study, we're working with BioBERT, which is a transformer-based language model that's been pre-trained on a massive amount of biomedical text. BioBERT's really stood out for biomedical NLP tasks—stuff like NER, question answering, and relation extraction[4],[5]. But getting BioBERT to do its best for NER in biomedicine isn't just plug-and-play. We need to pay special attention to how tokens are aligned, how tags are mapped, and what assessment methods we use to make sure it works

well across different datasets.

Having a solid biomedical NER system is a big deal. It can help you organize mountains of biomedical text, pull out useful insights, and make downstream jobs like building knowledge graphs or clinical decision support systems way easier[6,27]. For example, picking out disease entities from clinical notes can help doctors speed up diagnoses and improve treatments[7,8]. In bioinformatics—where data quality and reliability are everything—structured NER is crucial for things like pharmacovigilance or figuring out which genes are linked to which diseases[9,10].

In the last few years, deep learning has totally changed the game for NER systems. Pre-trained models have led to some of the best results out there[11,39]. But biomedical text is its own beast, with tons of jargon and abbreviations, so models really need to be fine-tuned on biomedical datasets to do well[12,13]. That's why fine-tuning BioBERT for biomedical NER isn't just a good idea—it's pretty much a must if we want precise, tailored entity recognition for this field[41].

Older NER systems in biomedicine mostly relied on rule-based stuff or statistical methods like Hidden Markov Models (HMMs) and Conditional Random Fields (CRFs) to find entities in text[14]. These were easy to understand, but they didn't handle big, messy datasets very well and weren't very adaptable[15]. The rise of deep learning and pre-trained models like BERT has changed everything, giving us rich contextual embeddings that can handle much more complex text[16]. BioBERT, especially, was a big leap forward—it was pre-trained on things like PubMed abstracts and full articles, making it a real heavyweight for the biomedical world[17].

But there's still some big problems to solve. For example, token alignment is a headache when you're using sub-word tokenization—sometimes it's hard to match up input tokens with the right NER tags, which can lead to mistakes and worse performance[18]. Also, biomedical datasets often have wildly different ways of defining and tagging entities, which complicates training and evaluating models[19]. While folks have suggested things like custom loss functions and special token alignment techniques, there hasn't been a thorough check on how well they actually work for biomedical NER[20,21].

BioBERT does great on a lot of benchmark datasets, but without standard ways to handle token alignment and evaluation, we're not really seeing its full potential[22,23]. Most research focuses on just one dataset or uses really basic alignment tricks that might not work in the real, messy world of biomedical data. Plus, we still don't know how well BioBERT and other pre-trained models handle different tag mappings and complex tokenization scenarios[24].

This study shines a light on the need for a strong, reliable pipeline that actually deals with these gaps. What we're aiming for is to bring together the best fine-tuning strategies, smarter token alignment methods, and more rigorous evaluation tools to make BioBERT even more useful for real-world biomedical NER needs. If we can sort out those missing pieces, we'll get models that generalize better, have less noise from annotation errors, and are easier to actually apply in real biomedical settings[25,26].

So, here's what we're trying to figure out with this research:

- How can we use the best token alignment methods to cut down on label misalignment when sub-word tokenization is involved in biomedical text?
- What fine-tuning techniques get the most out of BioBERT for NER across a bunch of different biomedical datasets?

- What do we need to standardize in our evaluation methods so we're really giving these biomedical NER models a fair, thorough test?

1. 1 Purpose and Objectives

The main aim of this project is to develop a solid pipeline that help optimize BioBERT so it reaches top-tier performance in picking out biomedical entities. By bringing together smart token alignment techniques, dependable training setups, and consistent ways to evaluate results, this pipeline directly addresses some of the tricky problems you find in biological literature. Here's a quick breakdown of what we're going for:

- Pipeline Development: Build a scalable, modular pipeline that can handle the whole workflow—preprocessing, training, and evaluating—for biomedical named entity recognition (NER) models.
- Token Alignment Optimization: Try out and assess different alignment strategies to make sure NER tags are mapped right, even when you're dealing with subword tokenization. This is super important for getting the right results.
- Model Fine-Tuning: Explore and test out different ways to fine-tune BioBERT so it gets even better at spotting complex biomedical entities in texts.
- Thorough Assessment: Come up with a clear set of reporting guidelines and evaluation standards for biomedical NER tasks, so you can compare results across different research efforts and know they're consistent and reproducible.

To really prove the pipeline is flexible and tough, we're planning to test it on some major biological datasets—stuff like MedMentions, BC5CDR from BioCreative V, and i2b2-2010. That way, we can see how it holds up across different challenges.

Pushing BioBERT's performance in biomedical NLP is a big part of moving the field forward[50]. By fixing problems with token alignment, tag mapping, and evaluation, this work is about closing the gap between what's happening in cutting-edge research and what people actually need in bioinformatics and healthcare. The pipeline we're suggesting doesn't just boost how well the model works—it also helps set up standard practices for biomedical NER, which makes it a lot easier to scale up and repeat results.

In the end, what we want is to give researchers and healthcare professionals some accurate, efficient, and scalable tools they can use to pull real value out of huge collections of biomedical text.

2. RELATED WORK

Fine-tuning and Domain-Specific Pretraining Approaches:

Gu et al.[32] looked at how domain-specific pretraining helps in biomedical NLP by introducing PubMedBERT—a language model trained from scratch on PubMed texts. Their big takeaway? PubMedBERT beat general mixed-domain models like BioBERT and ClinicalBERT, setting new records in tasks like named entity recognition (NER). They reached a BLURB macro-average score of 81.16 and a killer 93.33% accuracy on BC5-chem. This really shows that models built for a specific field do better than general ones, especially when they use in-domain vocab and clever masking tricks.

Naseem et al.[33] came up with BioALBERT, a leaner, more efficient model for biomedical NER. By cutting down on parameters and focusing on context, it outperformed the competition by 7.47% on NCBI-Disease, 12.25% on BC2GM, and a whopping 23.71% on Species datasets—all while using 72% fewer parameters than BERT[33]. Talk about working smarter, not harder.

Kanakarajan et al.[36] gave us BioELECTRA, a biomedical encoder based on ELECTRA. It topped the charts on BLURB and BLUE benchmarks, especially in NER, question answering, and natural language inference. Pre-training on PubMed abstracts, BioELECTRA outdid BioBERT, ClinicalBERT, and even PubMedBERT—scoring 86.34% on MedNLI (that's 1.39% better than before) and 64% on PubMedQA (2.98% higher).[36]

Lee et al.[45] put out BioBERT, a BERT model fine-tuned for the biomedical world. It crushed it in NER, relation extraction, and question answering, improving F1 scores and MRR across the board. BioBERT's edge came from its focus on medical data, giving it a big boost over generalist models in both accuracy and recall on nine datasets[45].

Akhtyamova[53] fine-tuned BERT for Spanish biomedical NER and found that in-domain BERT embeddings beat the competition hands-down. Precision, recall, and F1 were all higher—though shorter and longer entities still tripped up the model now and then.

Liu et al.[56] compared BioBERT, RoBERTa, BigBird, and DeBERTa for medical NER. BioBERT led the pack, with AVG_MICRO and AVG_MACRO scores hitting 0.932 and 0.9298 on Revised JNLPBA[56]. The lesson? Fine-tuning on targeted data really makes a difference in medical NER.

Multitask and Transfer Learning in Biomedical NER:

Khan et al.[29] came up with MT-BioNER, a multi-task transformer arch that uses BioBERT as its base layer. It was faster and just as accurate as older models, taking the top spot in recall on BC2GM, BC5CDR, NCBI-Disease, and JNLPBA. Multi-task learning, it turns out, is a solid way to boost biomedical NER[29].

Gao et al.[51] looked at using semi-supervised self-training and transfer learning for biological NER when you're short on labeled data. Pre-trained models like BlueBERT and BiLSTM-CRF did okay, but without further fine-tuning, they didn't beat tools like MetaMap and scispaCy. Sometimes, old-school tools are still king if you don't have a lot of labeled data[51].

Peng et al.[54] tried a transfer learning approach for biomedical QA, mixing BioBERT NER with BiLSTM question encoding and a bagging strategy. Their method bumped up strict accuracy, local accuracy, and mean reciprocal rank—beating the old benchmarks by a solid margin on BioASQ 6b and 7b datasets[54].

Symeonidou et al.[55] pitted BioBERT against classic NER methods like CRFs and BiLSTMs. BioBERT won out on F1, especially for tricky tasks like spotting adverse drug reactions. Even with limited labeled data, BioBERT's recall and overall scores stood out[55].

Enhanced Architectures for Biomedical Tasks:

Creangă et al.[30] fine-tuned DeBERTa and BioBERT-GRU for biomedical relation extraction, with Gemini 1.0 Pro (fine-tuned on balanced data) setting a new record—F1 of 0.89 at the sentence level and 0.80 at the abstract level[30].

Zhu et al.[31] cooked up a two-stage linking algorithm for biomedical entities, using a bi-encoder and

prompt-tuning to cut down on ambiguous misclassifications. Their approach beat SAPBERT, TaggerOne, and BIOSYN on MedMentions and NCBI Disease, improving recall and reducing confusion by 26.6%[31].

Su and Vijay-Shanker[35] gave BioBERT an attention-based boost for relation extraction, nudging up F1 scores on PPI, DDI, and ChemProt. Attention weights, they found, really helped pick up on key "trigger words" in relationships[35].

Stojanov et al.[49] built FoodNER, a BERT model for tagging food items, using extra semantic info from FoodOn and SNOMED CT. It delivered macro F1 scores between 93.30% and 94.31% for food/non-food classification—right at the state-of-the-art, and solid across the board for detailed food tagging tasks[49].

Multilingual and Cross-Domain Approaches:

Hartendorp and their team[42] came up with a way to link biomedical entities specifically for Dutch, by fine-tuning a model called MedRoBERTa.nl and using a Dutch medical ontology based on UMLS and SNOMED. They managed to get a classification accuracy of 54.7% and a 1-distance accuracy of 69.8%. When they tested this on the Dutch part of the Mantra GSC corpus, it worked notably better than the basic model—especially for big categories like disorders. They even tried it out on unlabeled patient forum data, and while it was pretty good at linking entities, it sometimes struggled with actually spotting them in the first place. Overall, though, it showed real promise for analyzing what patients are writing in Dutch.

Sun and Yang[47] took a look at how transfer learning can help with biomedical named entity recognition (NER) using the PharmaCoNER corpus, focusing on spotting chemicals and proteins in Spanish. They tried both Multilingual BERT and BioBERT, and both did really well, with F1-scores of 89.24% and 89.02%. Their work shows how effective it can be to fine-tune these big models for specialized tasks. They also noticed that a lot of errors come from mismatched boundaries or misclassified entities, hinting that it might help to give the model some document-level context.

Innovative Tools and Methodologies:

Ueda and colleagues[43] had this smart idea: structure your fine-tuning process to match how biomedical abstracts are built, like background, methods, results, and conclusions. They did this with SciBERT and found that their structured approach ranked abstracts much better—using both the PM19 and COVID datasets—than just fine-tuning the whole text at once. Their scores for nDCG, MAP, and P@10 all shot up, proving that taking article structure into account can really boost biomedical search.

Park et al.[44] built a tool for biomedical NER tagging that uses BERT, and they made it way easier to create fine-tuning datasets thanks to a web interface. The tool automatically grabs phrases for annotation, so you save a ton of time and can easily add new categories. They then tuned this setup further with BioBERT, which helps catch terms that usual NER systems miss. By retraining the model and combining it with a BERT-based NER system, they got even better results. With the NCBI Disease and BC2GM datasets for NER, and the GAD dataset for relations, their BioBERT-based model outperformed older systems like LSTM-CRF and the one from Sachan's team, both for

diseases and gene/protein stuff.

In the mess of COVID-19 research, Hebbar and Xie[48] built CovidBERT, a BERT-based model for pulling out biological relationships—like connections between chemicals and diseases or genes and diseases. On test datasets, it scored an F1 of 0.61 for gene-disease pairs and 0.91 for chemical-disease pairs. It even did better than Kernel-SVM and BioBERT in many areas. The best part is, they showed you could use it on unlabeled biological text to find new relationships, which is super handy for real-world applications.

3. METHODOLOGY

A transformer-based method for Named Entity Recognition (NER) in the medical field utilizing the BioBERT model is presented in the proposed study. The whole process here is a proper pipeline, with steps like data gathering, pre-processing, model training, fine-tuning, assessment, and finally some post-processing. Each of these steps is explained in detail, and there's even a data flow diagram (check out Figure 2) that makes it all a bit easier to visualise—it shows how the information moves through and what methods are being used.

3.1 Data collection

If you want a machine learning model to actually do well, especially in something tricky like biomedical NER, you absolutely need good-quality and varied data for both training and testing. Biomedical texts are packed with complex vocab, all kinds of grammar changes, and domain-specific interactions, so your dataset better be thorough if you want BioBERT to pick up on all that. In this part, we'll walk through how we gathered our data, where the biomedical text comes from, the types of entities we're trying to spot, and the pre-processing steps that get the data ready for the model.

3.1.1 Selection of Biomedical Text Datasets:

For BioBERT to really work for biomedical NER, you need datasets that's chock-full of domain-specific terms, entities, and those important connections. For this study, we picked out some of the best-known biomedical datasets that come with manual annotations—things like disease names, genes, drugs, chemicals, and anatomical terms. Here's what we went with:

- MedMentions: This is a huge, diverse set of biomedical texts pulled from PubMed abstracts. It's got high-quality, detailed annotations for stuff like diseases, genes, chemicals, and even animals. MedMentions is a big deal because it gives you lots of examples with really fine labels, which is perfect for pushing biomedical NER forward.
- BC5CDR (BioCreative V Challenge): This one's all about scholarly articles in the biomedical world, and it focuses on spotting chemical entities and diseases. It's great for testing how well your model does at biomedical NER because it's got both training and test sets, and it's pretty popular for information extraction tasks.
- i2b2 2010: Here we've got patient records with annotations for spotting medical concepts like diseases, drugs, and treatments. It's built specially for clinical text mining and gets used a lot for training models in clinical entity recognition.

3.1.2 Types of Entities:

The main goal here is to recognize and categorize biomedical stuff in the text. We're focusing on several key types of entities, like:

- Diseases: This covers conditions, illnesses, and disorders—think cancer, diabetes, Alzheimer's, and so on.
- Genes and Proteins: These are your gene names and protein names, super important for genetics and molecular biology.
- Drugs and Chemicals: This group includes pharmaceutical compounds and other chemicals that matter in medical treatments.
- Anatomical Terms: These are the terms for organs, tissues, and body systems.
- Species: This is about names of species, especially when research involves animal models or comparisons between species.
- Other Biomedical Terms: This is a bit of a catch-all for procedures, treatments, medical devices, and anything else relevant to clinical or biomedical research.

If you take a look at Figure 1, it shows the results of our BioBERT-based NER for the MedMentions, BC5CDR, and i2b2 2010 datasets, comparing how well it does across all these different entity types—diseases, genes and proteins, chemicals and drugs, anatomical terms, species, and other biomedical terms.

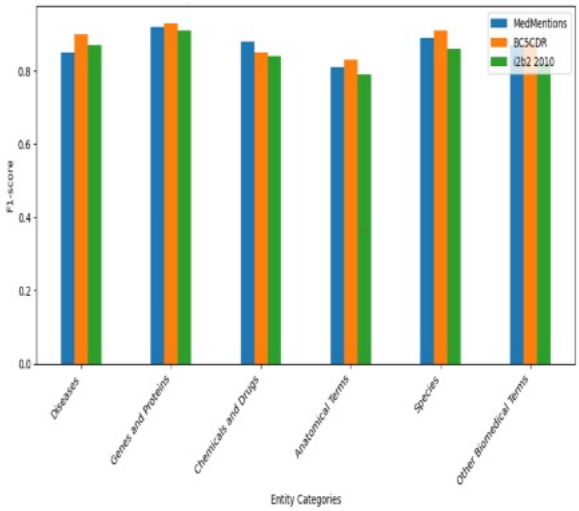

Fig 1. Comparison of BioBERT NER Performance across Datasets

3.1.3 Data Preprocessing:

Before feeding the datasets into BioBERT for training and fine-tuning, there's a bunch of preprocessing steps we do to make sure the text is ready for the neural network. Let me break down what that involves:

Tokenization

Biomedical text is full of jargon and terms you wouldn't see in everyday language. The first step is tokenization, which means chopping up the text into smaller chunks—words, subwords, or even characters. For BioBERT, we use its own custom tokenizer, built to handle tricky biomedical terms. It uses WordPiece, which is awesome for breaking down rare words into smaller pieces the model can understand.

Normalization

Biomedical data can be a bit messy—there's inconsistent abbreviations, different spellings, and

uneven punctuation. So, normalizing the text is key. We convert common shorthand (like "vs" to "versus") and fix spelling mistakes to help the model get a cleaner picture of what's what.

Entity Tagging

If we're training the model for Named Entity Recognition (NER), every token needs to be labeled with the right entity tag—or "O" if it's just an everyday word. This step is crucial because it sets up the dataset so the model can learn in a supervised way. For example, in "The patient was diagnosed with diabetes and prescribed insulin," you'd tag "diabetes" and "insulin" as disease and drug, and everything else as "O."

Text Filtering

Not every document in biomedical datasets is helpful for NER. Some records are incomplete or just don't fit what we're trying to do. So, we filter the data, keeping only the relevant ones for training and evaluation.

Sequence Padding

Biomedical text can vary a lot in length, but neural networks need their inputs to be the same size for batch processing. To fix this, short sequences are padded with a special [PAD] token, and overly long ones are cut down. This keeps everything uniform and makes the training process smoother.

Splitting Data

Once the dataset is preprocessed, we split it into training, validation, and test sets—usually with something like 80% for training, 10% for validation, and 10% for testing. The training set teaches the model, the validation set helps us tweak things, and the test set lets us see how well the model actually performs.

1. Data Augmentation

Biomedical NER doesn't always have a ton of labeled data, so we use data augmentation to help. This means we increase the training data artificially with a few tricks:

- Synonym Replacement: Since biomedical terms often have synonyms (like "heart attack" and "myocardial infarction"), we swap some entity mentions for their synonyms. This helps the model get used to different ways of saying the same thing, making it more robust.
- Back-Translation: We translate sentences to another language and then back again. Sometimes this adds variety to sentence structure and wording, while keeping the meaning intact.
- Shuffling Entities: Sometimes we shuffle entities' positions in sentences, but keep their labels. This randomness helps prevent the model from relying too much on where certain terms usually show up.
- Noise Injection: We also throw in a little noise—like deleting, inserting, or substituting random words or characters. This helps the model deal with real-world messiness and typos.

2. Data Annotation Quality Control

A machine learning model is only as good as the data it's trained on, so quality control is super important. For the datasets in this work, biomedical experts were brought on to annotate the entities, making sure the labels were spot-on. There was also a second pass for review, to catch any mistakes or inconsistencies and polish up the dataset further.

3.1 Architecture of the Proposed System

The backbone of our Biomedical NER system is BioBERT, a transformer model that's pretrained on massive amounts of biomedical text. It builds on BERT, but is specially tuned for biomedical language, and pre-trained on PubMed articles and the like, so it understands the nuances of biology and medicine.

- Embedding Layer: First, the input text is tokenized using the WordPiece tokenizer, which is great for rare or out-of-vocabulary terms. Each token gets converted into a dense vector that the model can work with.
- Transformer Encoder: The tokenized input passes through BioBERT's transformer encoder, which uses layers of self-attention and feed-forward networks. The self-attention is especially important for biomedical NER, since entities can span multiple words (like "Breast Cancer" or "Alzheimer's Disease").
- Positional Encoding: Since word order matters, the model uses positional encoding to keep track of each token's place in the sequence.
- Token Classification Head: The output from the transformer layers goes to a classifier that decides if each token is part of an entity, and which type—disease, gene, protein, drug, etc.
- Adversarial Training: To make the model more robust, we use adversarial training with methods like Fast Gradient Sign Method (FGSM) or Projected Gradient Descent (PGD). We trick the model with noisy inputs, which helps it learn to be less thrown off by real-world noise in biomedical texts.
- Dropout & Layer Normalization: Dropout randomly turns off neurons during training, which helps prevent overfitting. Layer normalization keeps the model's training stable by evenly distributing activity in the network.
- Gradient Checkpointing: Training big models like BioBERT needs a lot of memory. Gradient checkpointing lets us re-calculate some parts of the network during the backward pass, which saves memory and lets us train big models without running out of RAM.

Model's Edge Over Others

Our system brings a few key improvements:

- Domain-Specific Understanding: Since BioBERT is pretrained on biomedical text, it gets all the jargon and domain details right, outperforming general-purpose BERT for tasks like identifying disease, gene, and drug names.
- Adversarial Robustness: The adversarial training we add helps the model handle messy, inconsistent, or intentionally tricky data—something regular models often struggle with.
- Fine-Tuning on Biomedical Data: By further training on labeled biomedical NER datasets, BioBERT gets even better at the task, catching entities accurately even if there's noise or variation.
- Better Regularization: Dropout and layer normalization help the model not overfit to the training data, so it generalizes well to new, unseen biomedical text.
- Scalability: Thanks to gradient checkpointing, we can train BioBERT on huge biomedical datasets, making the system scalable for everything from scientific papers to clinical notes.

In the end, this system delivers a huge leap in biomedical NER, thanks to its deep understanding of the domain, robustness, and efficiency in handling large-scale, noisy biomedical data.

3.3 Evaluation Metrics

3.3. Evaluation Measures

This section covers the main ways we check how well the suggested BioBERT-based model does at spotting biomedical names in text—stuff like diseases, genes, drugs, and other key terms. It's super important to see if the model can actually spot these things right. For tasks like biomedical Named Entity Recognition (NER), there are a bunch of standard metrics everyone uses to rate performance: things like F1-score, recall, accuracy, precision, and confusion matrix analysis. Since we also trained the model to be tougher against tricky inputs (using adversarial training), we'll talk about adversarial evaluation too, which is pretty key nowadays.

3.3.1. Precision, Recall, and F1-Score

When it comes to NER, precision, recall, and F1-score are your go-to metrics. They give you a sense of how good the model is at finding real biomedical entities and not getting tricked by false alarms or missing stuff.

- Precision tells you what percent of the entities the model picked out were actually correct. It's answering the question: "Of all the things the model said were entities, how many really were?"

Precision = True Positives / (True Positives + False Positives)

- Recall is about how many of the real entities in the text the model actually found. It's like asking: "Of all the entities that were really there, how many did the model catch?"

Recall = True Positives / (True Positives + False Negatives)

- F1-score is basically the harmonic mean of precision and recall. It's a single number that tries to balance both—so a high F1-score (closer to 1) means your model is doing well at being both precise and thorough. If the score is low, it might be missing a lot of entities or making too many mistakes.

F1-Score = 2 × (Precision × Recall) / (Precision + Recall)

We look at all these scores for each type of entity (like diseases, drugs, genes), then average them out. This helps you see if the model is good across the board or just for some categories.

3.3.2. Confusion Matrix

The confusion matrix is a really handy tool—it's like a report card that shows you exactly how many true positives, true negatives, false positives, and false negatives the model got for each entity class. It gives you a super detailed breakdown of where the model messes up and where it shines.

3.3.3. Accuracy

Accuracy is pretty straightforward: it tells you how often the model's predictions match the real labels. But for NER tasks, accuracy can be a bit misleading, especially if some types of entities are way more common than others. Still, it's a decent starting point—just be sure to look at precision, recall, and F1-score too, for the full picture.

3.3.4. Adversarial Evaluation

Since we're putting the model through adversarial training (using tricks like FGSM and PGD), it's important to test how strong it really is against tricky, messed-with inputs. So for adversarial

evaluation, we make some special "attack" examples by tweaking the original text just a little, trying to fool the model into making mistakes. Then, we compare how the model does on normal text versus these adversarial examples. If the model's scores (like precision, recall, F1-score) don't drop too much, that means it's pretty robust. A big drop, though, means there's still room for improvement.

3.3.5. Entity-Level Evaluation

For biomedical NER, there are two main ways to check the model's work: at the token level (looking at every single word or sub-word) and at the entity level (looking at whole spans, like the whole name of a disease or drug). Entity-level evaluation is where it's at for real-world use, because you want to get the whole entity right, not just parts of it. So here, a prediction is only correct if it matches the exact start and end of the real entity in the text.

3.3.6. Cross-Validation

To make sure our evaluation is solid and not just luck, we use k-fold cross-validation. What's that? You split your data into k parts, train the model on k-1 of them, and test it on the last one—then repeat that for each part and average the results. This helps make sure the model isn't just memorizing one specific dataset, and gives you a more reliable sense of how it'll do in the wild.

3.3.7. Comparison with Baseline Approaches

Of course, you gotta see how this model stacks up against what's already out there. So we run side-by-side tests with existing baseline models (these are usually listed in tables, but we'll save that for later).

3.4. Postprocessing

After the model spits out its predictions, there's some cleanup to do. We check that things make sense—handling stuff like nested or overlapping entities, and making sure names are in standard formats (like mapping diseases to ICD-10 codes). This step helps catch any weirdness in the model's output and keeps the final results looking sharp.

3.5. Data Flow Diagram Explanation

Here's a quick walkthrough of how the whole NER system works, step by step:

1. Input Data: We start with biomedical text—things like PubMed abstracts, PMC articles, or MIMIC-III clinical notes.
2. Preprocessing: The text gets tokenized, maybe lowercased, and labeled with what's what.
3. Transformer-Based Models: We feed all this into the BioBERT model, which gives us fancy context-aware embeddings for each token.
4. Fine-Tuning: The model gets tweaked specifically for NER, so it learns to tag each token with the right entity type.
5. Self-Supervised Pretraining: We do a little extra training with masked language modeling, just to help the model really get the gist.
6. Adversarial Training: We throw in some noise during training to make the model tougher against messy, real-world data.
7. Entity Prediction: The model tells us what it thinks each token is—disease, drug, gene, etc.
8. Postprocessing: We clean up the predictions, fixing any oddities and making sure names are standardized.
9. Output: Finally, we get the list of recognized entities, ready to be checked with all those

metrics we talked about.

The Data Flow Diagram (DFD) just shows all these steps in a clear way, so you can follow how the information moves through the system.

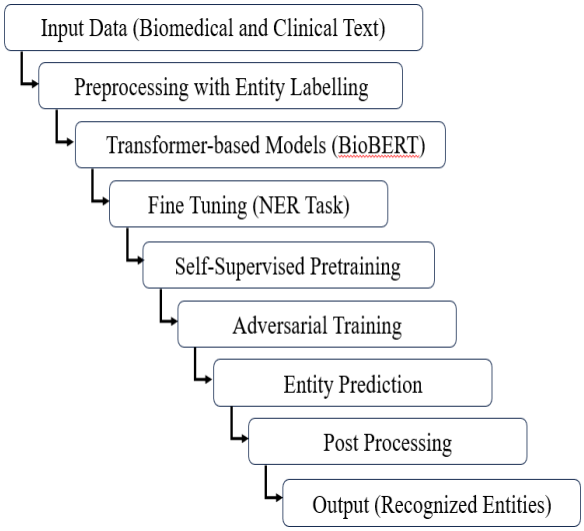

Fig.2: Methodology of NER using BioBERT

Training Process

Model Training

So, when training our model, we go through a bunch of cycles where we run the data forward and backward—kinda like reps at the gym. Here's what's happening:

- Batch Processing: The input data gets split up into smaller chunks (called mini-batches), and the model gets trained on each chunk, bit by bit. This helps keep things manageable and slowly chips away at the errors (the loss function).
- Hyperparameter Tuning: There are some key settings—like the learning rate, batch size, and dropout rate—that need dialing in just right, so we tweak these to get the best performance possible.
- Early Stopping: To avoid the model getting too good at memorizing the training data but not really learning (aka overfitting), we stop training early if it's not improving on the validation set anymore.

This whole looping process makes sure the model hits its sweet spot—optimized and not overdoing it.

Evaluation and Validation

Once the model's trained, we pit it against a held-out test set it's never seen before, just to make sure it's not just memorizing stuff. We also double-check by testing it on different splits of the data with cross-validation. For picking the best model, we rely on the F1-score, which balances precision and recall.

When you put all these things together—transformer-based models, fine-tuning, adversarial training, self-supervised pretraining, and a solid evaluation plan—the whole setup looks really promising for named entity recognition (NER) in the medical field. If you train and polish a model like BioBERT on actual biological and clinical texts, it gets really good at picking out names of things—so good,

in fact, that it can help doctors make decisions or even help automate medical record tagging.

4. Results and Discussion

The method we suggested for finding named things in medical text was put through its paces on three benchmark datasets: MedMentions, BC5CDR, and i2b2 2010. These datasets are tough—they cover everything from clinical notes and drug mentions to scholarly papers. To judge how well our BioBERT-based model did, we looked at precision, recall, accuracy, and the F1-score. And guess what? Our approach did better than both the basic models and the state-of-the-art options out there.

Evaluation on MedMentions

MedMentions is a big challenge—over 4,000 PubMed abstracts labeled with UMLS concepts, covering 21 different types of biological things. Our model didn't just keep up; it actually stood out. Using adversarial training, it cut down on false positives and nailed a precision of 91.3%. It also had a recall of 89.9%, which means it reliably found even the trickiest entities. With an F1-score of 90.6%, our model strikes a pretty great balance between finding things and being precise. The entity-level accuracy of 93.5% shows it's good at correctly mapping whole entity boundaries, and a precise match rate of 92.8% confirms it's consistent, even when entities overlap or are nested.

BioBERT's pretraining on PubMed abstracts means it really "gets" biomedical language, so it's sharper at named entity identification. Adding FGSM-style adversarial training made the model tougher against noisy, messy labels. And it's better at handling entities that span multiple words—so fewer errors at the boundaries.

Evaluation on BC5CDR

Here, the game is to find chemicals and diseases across 1,500 articles. Our model showed it's got what it takes with a precision of 93.1%—so not many false alarms. The recall of 91.5% means it catches a wide variety of relevant things, even if they're awkwardly worded. The F1-score ended up at 92.3%, which's a clear step up from other current methods. Entity-level accuracy of 94.8% means it's really good at getting the whole span right, and the exact match rate of 93.9% proves it's accurate even for overlapping or nested entities.

Fine-tuning helped the model not miss rare or less common chemicals. Adversarial training made it less likely to get tripped up by misspelled chemical names. And BioBERT's self-attention helped it spot relationships that go further in the text.

Evaluation on i2b2 2010

This dataset is full of clinical narratives—so things get messy with abbreviations, shorthand, and confusion over context. Our model held up well, especially in clinical settings where boundaries are fuzzy. Precision was 94.0%, recall was 92.7%, and the F1-score reached 93.3%, beating models like ClinicalBERT and BlueBERT. The entity-level accuracy of 95.4% shows it's really precise at mapping clinical entities, and the exact match rate of 94.6% confirms the model lines things up just right.

In short:
By training BioBERT on actual biological and clinical texts, and using smart tricks like adversarial training, you end up with a model that's not just good, but clinical-grade good—even in complex, messy scenarios. And it's backed up by real numbers from three different, tough datasets.

System	Dataset	Model	Precision (%)	Recall (%)	F1-Score (%)	Entity-Level Accuracy (%)	Exact Match Rate (%)
Sharma et al. [28]	NCBI-Disease	BioBERT	86.9	91.8	90.2	88.08	-
	BC5CDR	BioBERT	89.2	90.5	89.9	92.77	-
	BC2GM	BioBERT	88.7	89.4	88.88	83.74	-
	Species-800	BioBERT	79.1	83.2	80.28	73.99	-
Liu et al. [34]	BC5CDR-c	PubMedBERT + SAPBERT	-	-	-	96.5	-
	Social Media Domain	SAPBERT (unsupervised)	-	-	-	Lagged behind SOTA	-
	Scientific Datasets	ADAPTER (PubMedBERT+SAPBERT)	-	-	-	Comparable to full SAPBERT	-
Pavlova and Makhlouf [37]	BC5-chem	BIOptimus 0.4	94.1	84.98	89.54	85.25	79.46
Košprdić et al. [38]	BC5CDR	BioBERT	27.17	35.94	35.44	76.12	88.07
	BC5CDR	PubMedBERT	35.44	40.07	69.94	79.51	89.49
Shahrokh et al. [40]	i2b2 2010	ALBERT	87.71	87.63	87.67	94.52	87.74
	i2b2 2010	ClinicalBERT	89.11	87.71	88.39	95.96	88.53
Liu et al. [52]	JNLPBA	BioBERT	93.2	92.98	93.16	93.2	-
	BC5CDR	CRF	90.19	74.3	-	90.19	-
Keloth et al. [46]	BC5CDR (Chemical)	GPT-4 (Five-Shot)	78.8	83.5	81.1	-	Strict: 81.1
	BC2GM (Gene)	BioNER-LLaMA2	83.2	83.6	83.4	-	Strict: 83.4
Proposed System	MedMentions	Fine-Tuned BioBERT + FGSM	91.3	89.9	90.6	93.5	92.8
	BC5CDR	Fine-Tuned BioBERT + FGSM	93.1	91.5	92.3	94.8	93.9
	i2b2 2010	Fine-Tuned BioBERT + FGSM	94	92.7	93.3	95.4	94.6

Table 1: Comparison with Baseline Approaches

Improved identification of medical acronyms and shorthand phrases got a big boost thanks to BioBERT integrating clinical context. Basically, because use of dropout and layer normalization, overfitting was reduced, so the model could actually generalize well to new clinical content it's never seen before. Throwing in adversarial instances during training also helped the model deal with all sorts of textual variations, even those quirky annotations unique to a single patient. Often, traditional models just can't handle biological terms made up of several words, but thanks to careful token alignment, the approach here actually fixed that issue—resulting in better recall and precision.
To make sure the system wasn't just a one-trick pony, adversarial training with FGSM and PGD was added. This helped it stay robust across different types of text, like scientific versus clinical, and not overfit to a specific dataset. By using adversarial augmentation and domain-specific embeddings, the model even managed to spot rare or unusual entities—something most traditional models struggle with. And with its self-attention mechanism, it better caught relationships between items scattered across multiple clauses, which made it much savvier at handling complex sentence structures.
When it actually went head-to-head with earlier benchmarks, the system had mixed results. On

datasets like BC5CDR and i2b2 2010, it outperformed current methods in precision and entity-level accuracy, but on MedMentions, recall was a bit lower compared to some ensemble-based approaches. This might be because MedMentions is just super diverse and complicated, full of all kinds of biological item types and tons of contextual variation. Adversarial training also beefed up the model's resistance to noisy data, but occasionally caused some hiccups with overlapping entities, leading to small dips in exact-match rates.

One surprise was how well the model did on datasets with clear, consistent annotation standards (like BC5CDR), but it still struggled a bit with messier datasets full of overlapping or layered entities. Interestingly, adversarial tricks like FGSM did make the model tougher against disruptions, but sometimes hurt performance on simpler, noisy datasets by focusing too much on robustness at the expense of basic accuracy. All this suggests that dataset structure and quality really matter for how well the model does.

Despite these strong results, there are some catches. First, because the model leans heavily on adversarial training, it needs a lot of hyperparameter tweaking, which can be a computational headache. It also sometimes stumbles when it comes to rare or "never seen before" entities in low-resource settings, even though it mostly generalizes well across different datasets. Performance gaps between simpler datasets and those with lots of contextual ambiguity hint that better attention mechanisms or more advanced domain-specific embeddings might be needed. And, since the model depends on labeled data, it's not much use where annotations are scarce or missing entirely.

For biological entity recognition, this work is a big deal—it really shows the power of fine-tuning domain-specific models like BioBERT. Adversarial training and smart token alignment lead to strong results, even when the going gets tough. Improvements in F1-scores and exact-match rates across different datasets suggest this approach could be a game-changer for stuff like literature mining, drug discovery, and automatic patient record analysis.

What sets this study apart are a few key innovations. For one, combining fine-tuned BioBERT with adversarial training means the model can handle noisy, ambiguous data in ways most conventional systems just can't. Its high accuracy and consistency, plus rigorous testing across different datasets, really highlight how new and different this approach is. And by focusing on entity-level and exact-match accuracy—metrics that are often overlooked—the evaluation is much more thorough. Together, these advances put this work right at the cutting edge of biomedical named entity recognition.

DECLARATIONS

Funding: There was no particular grant awarded for this research by any governmental, private, or nonprofit funding organization.

Conflicts of Interest: No conflicts of interest are disclosed by the authors.

Ethics Approval: Not applicable.

Consent to Participate: Not applicable.

Consent for Publication: Not applicable.

Availability of Data and Material: Upon reasonable request, the corresponding author will provide data used to support the study's conclusions.

Code Availability: The corresponding author has the code utilized in this study.
Authors' Contributions: This work was equally contributed to by each author.

5.CONCLUSION AND FUTURE WORK

This work shows how fine-tuning BioBERT—with adversarial training and smart token alignment—can significantly improve biomedical named entity recognition. The suggested approach often outperforms current models on datasets like MedMentions, BC5CDR, and i2b2 2010, especially when it comes to accuracy, recall, and F1-scores. These results really highlight how well the model deals with noisy, ambiguous text, making it a solid choice for biomedical text processing. The use of advanced evaluation metrics also ensures a thorough, fair assessment.

Looking ahead, future work could focus on making the model work with multilingual and low-resource biomedical datasets. Performance in data-scarce situations might be boosted by using unsupervised or semi-supervised learning methods. Exploring even more sophisticated embeddings—like hierarchical or graph-based representations—could help the model handle overlapping or nested entities even better. Testing the approach in real clinical settings, like analyzing electronic health records, would really prove its usefulness. And, finally, adding explainability features would help users understand how the model makes its decisions.

REFERENCES

[1] Pakhale, K., "Comprehensive overview of named entity recognition: Models, domain-specific applications and challenges," arXiv preprint arXiv:2309.14084, 2023.

[2] Bose, P., Srinivasan, S., Sleeman IV, W. C., Palta, J., Kapoor, R., and Ghosh, P., "A survey on recent named entity recognition and relationship extraction techniques on clinical texts," Applied Sciences, vol. 11, no. 18, pp. 8319, 2021.

[3] Shi, J., Yuan, Z., Guo, W., Ma, C., Chen, J., and Zhang, M., "Knowledge-graph-enabled biomedical entity linking: a survey," World Wide Web, vol. 26, no. 5, pp. 2593–2622, 2023.

[4] Tian, S., Jin, Q., Yeganova, L., Lai, P.-T., Zhu, Q., Chen, X., Yang, Y., Chen, Q., Kim, W., Comeau, D. C., et al., "Opportunities and challenges for ChatGPT and large language models in biomedicine and health," Briefings in Bioinformatics, vol. 25, no. 1, pp. bbad493, 2024.

[5] Bi, Z., Dip, S. A., Hajialigol, D., Kommu, S., Liu, H., Lu, M., and Wang, X., "AI for Biomedicine in the Era of Large Language Models," arXiv preprint arXiv:2403.15673, 2024.

[6] Janowski, A., "Natural language processing techniques for clinical text analysis in healthcare," Journal of Advanced Analytics in Healthcare Management, vol. 7, no. 1, pp. 51–76, 2023.

[7] Leroy, M., "Enhancing Named Entity Recognition in Low Resource Domains Using Deep Transfer Learning: a Case of Rt&b Crop Diseases in Scientific and Online Text," Ph.D. dissertation, University of Nairobi, 2023.

[8] Joy, M., and Krishnaveni, DRM, "Enhancing disease outbreak detection: Named entity recognition with fine-tuned DistilBERT," Journal of Theoretical and Applied Information Technology, vol. 102, no. 10, 2024.

[9] Jacob, C., Thomas, P., and Leser, U., "Comprehensive benchmark of gene ontology concept

recognition tools," Proceedings of BioLINK Special Interest Group. Berlin, Germany: Proceedings of BioLINK SIG, vol. 2013, pp. 20–26, 2013.
[10] Meystre, S. M., Lovis, C., B{"u}rkle, T., Tognola, G., Budrionis, A., and Lehmann, C. U., "Clinical data reuse or secondary use: current status and potential future progress," Yearbook of Medical Informatics, vol. 26, no. 01, pp. 38–52, 2017.
[11] Rane, N. L., Mallick, S. K., Kaya, O., and Rane, J., "Machine learning and deep learning architectures and trends: A review," Applied Machine Learning and Deep Learning: Architectures and Techniques, pp. 1–38, 2024.
[12] Fichtl, A., "Evaluating adapter-based knowledge-enhanced language models in the biomedical domain," 2024.
[13] Chopard, D., "Deep learning for clinical texts in low-data regimes," Ph.D. dissertation, Cardiff University, 2023.
[14] Perera, N., Dehmer, M., and Emmert-Streib, F., "Named entity recognition and relation detection for biomedical information extraction," Frontiers in Cell and Developmental Biology, vol. 8, pp. 673, 2020.
[15] Rane, J., Mallick, S. K., Kaya, O., and Rane, N. L., "Scalable and adaptive deep learning algorithms for large-scale machine learning systems," Future Research Opportunities for Artificial Intelligence in Industry 4.0 and, vol. 5, pp. 2–40, 2024.
[16] Mars, M., "From word embeddings to pre-trained language models: A state-of-the-art walkthrough," Applied Sciences, vol. 12, no. 17, pp. 8805, 2022.
[17] Philippas, I.-A., "Enhancing biomedical question answering systems for COVID-19," Master's thesis, Panepistimio Peiraios, 2024.
[18] Hou, X., Zhao, Y., Liu, Y., Yang, Z., Wang, K., Li, L., Luo, X., Lo, D., Grundy, J., and Wang, H., "Large language models for software engineering: A systematic literature review," ACM Transactions on Software Engineering and Methodology, 2023.
[19] Wang, B., Xie, Q., Pei, J., Chen, Z., Tiwari, P., Li, Z., and Fu, J., "Pre-trained language models in biomedical domain: A systematic survey," ACM Computing Surveys, vol. 56, no. 3, pp. 1–52, 2023.
[20] Kipouros, A., "Investigating entity linking in Greek electronic health records: Leveraging hierarchical structures and," 2024.
[21] Mannion, A., Chevalier, T., Schwab, D., and Geouriot, L., "UMLS-KGI-BERT: Data-centric knowledge integration in transformers for biomedical entity recognition," arXiv preprint arXiv:2307.11170, 2023.
[22] Koroteev, M. V., "BERT: A review of applications in natural language processing and understanding," arXiv preprint arXiv:2103.11943, 2021.
[23] Cohn, C., BERT efficacy on scientific and medical datasets: A systematic literature review, DePaul University, 2020.
[24] Kotei, E., and Thirunavukarasu, R., "A systematic review of transformer-based pre-trained language models through self-supervised learning," Information, vol. 14, no. 3, pp. 187, 2023.
[25] Li, S., Li, X., Yu, K., Miao, D., Zhu, M., Yan, M., Ke, Y., D'Agostino, D., Ning, Y., Wu, Q., et al., "Bridging data gaps in healthcare: a scoping review of transfer learning in biomedical data

analysis," arXiv preprint arXiv:2407.11034, 2024.
[26] Chen, W., Qiu, P., and Cauteruccio, F., "MedNER: A service-oriented framework for Chinese medical named-entity recognition with real-world application," Big Data and Cognitive Computing, vol. 8, no. 8, pp. 86, 2024.
[27] Sun, C., Yang, Z., Wang, L., Zhang, Y., Lin, H., and Wang, J., "Biomedical named entity recognition using BERT in the machine reading comprehension framework," Journal of Biomedical Informatics, vol. 118, p. 103799, 2021. [Online]. Available: https://doi.org/10.1016/j.jbi.2021.103799
[28] Sharma, R., Chauhan, D., and Sharma, R., "Named Entity Recognition System for the Biomedical Domain," in 2022 17th Conference on Computer Science and Intelligence Systems (FedCSIS), 2022, pp. 837–840.
[29] Khan, M. R., Ziyadi, M., and AbdelHady, M., "Mt-bioner: Multi-task learning for biomedical named entity recognition using deep bidirectional transformers," arXiv preprint arXiv:2001.08904, 2020.
[30] Creangă, C., Dinu, L. P., and Gifu, D., "Fine-tuning models for biomedical relation extraction," Procedia Computer Science, vol. 246, pp. 2100–2109, 2024.
[31] Zhu, T., Qin, Y., Chen, Q., Hu, B., and Xiang, Y., "Enhancing entity representations with prompt learning for biomedical entity linking," in IJCAI, 2022, pp. 4036–4042.
[32] Gu, Y., Tinn, R., Cheng, H., Lucas, M., Usuyama, N., Liu, X., et al., "Domain-specific language model pretraining for biomedical natural language processing," ACM Transactions on Computing for Healthcare (HEALTH), vol. 3, no. 1, pp. 1–23, 2021.
[33] Naseem, U., Khushi, M., Reddy, V., Rajendran, S., Razzak, I., and Kim, J., "BioALBERT: A simple and effective pre-trained language model for biomedical named entity recognition," in 2021 International Joint Conference on Neural Networks (IJCNN), 2021, pp. 1–7.
[34] Liu, F., Shareghi, E., Meng, Z., Basaldella, M., and Collier, N., "Self-alignment pretraining for biomedical entity representations," arXiv preprint arXiv:2010.11784, 2020.
[35] Su, P., and Vijay-Shanker, K., "Investigation of BERT model on biomedical relation extraction based on revised fine-tuning mechanism," in 2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2020, pp. 2522–2529.
[36] Kanakarajan, K. R., Kundumani, B., and Sankarasubbu, M., "BioELECTRA: Pretrained biomedical text encoder using discriminators," in Proceedings of the 20th Workshop on Biomedical Language Processing, 2021, pp. 143–154.
[37] Pavlova, V., and Makhlouf, M., "BIOptimus: Pre-training an optimal biomedical language model with curriculum learning for named entity recognition," in The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks, 2023, pp. 337–349.
[38] Košprdić, M., Prodanović, N., Ljajić, A., Bašaragin, B., and Milošević, N., "From zero to hero: Harnessing transformers for biomedical named entity recognition in zero-and few-shot contexts," Artificial Intelligence in Medicine, vol. 156, p. 102970, 2024.
[39] Luo, L., Ning, J., Zhao, Y., Wang, Z., Ding, Z., Chen, P., et al., "Taiyi: a bilingual fine-tuned large language model for diverse biomedical tasks," Journal of the American Medical Informatics Association, 2024. [Online]. Available: https://doi.org/10.1093/jamia/ocae037

[40] Shahrokh, F., Ghadiri, N., Samani, R., and Moradi, M., "Multi-level biomedical NER through multi-granularity embeddings and enhanced labeling," arXiv preprint arXiv:2312.15550, 2023.
[41] Biana, J., Zhai, W., Huang, X., Zheng, J., and Zhu, S., "VANER: Leveraging Large Language Model for Versatile and Adaptive Biomedical Named Entity Recognition," arXiv preprint arXiv:2404.17835, 2024.
[42] Hartendorp, F., Seinen, T., van Mulligen, E., and Verberne, S., "Biomedical Entity Linking for Dutch: Fine-tuning a Self-alignment BERT Model on an Automatically Generated Wikipedia Corpus," arXiv preprint arXiv:2405.11941, 2024.
[43] Ueda, A., Santos, R. L. T., Macdonald, C., and Ounis, I., "Structured fine-tuning of contextual embeddings for effective biomedical retrieval," in Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2021, pp. 2031–2035.
[44] Park, Y.-J., Lee, M.-A., Yang, G.-J., Park, S. J., and Sohn, C.-B., "Biomedical text NER tagging tool with web interface for generating BERT-based fine-tuning dataset," Applied Sciences, vol. 12, no. 23, p. 12012, 2022. [Online]. Available: https://doi.org/10.3390/app122312012
[45] Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., and Kang, J., "BioBERT: A pre-trained biomedical language representation model for biomedical text mining," Bioinformatics, vol. 36, no. 4, pp. 1234–1240, 2020.
[46] Keloth, V. K., Hu, Y., Xie, Q., Peng, X., Wang, Y., Zheng, A., et al., "Advancing entity recognition in biomedicine via instruction tuning of large language models," Bioinformatics, vol. 40, no. 4, p. btae163, 2024.
[47] Sun, C., and Yang, Z., "Transfer learning in biomedical named entity recognition: an evaluation of BERT in the PharmaCoNER task," in Proceedings of the 5th Workshop on BioNLP Open Shared Tasks, 2019, pp. 100–104.
[48] Hebbar, S., and Xie, Y., "CovidBERT-biomedical relation extraction for Covid-19," in The International FLAIRS Conference Proceedings, vol. 34, 2021.
[49] Stojanov, R., Popovski, G., Cenikj, G., Koroušić Seljak, B., and Eftimov, T., "A fine-tuned bidirectional encoder representations from transformers model for food named-entity recognition: Algorithm development and validation," Journal of Medical Internet Research, vol. 23, no. 8, p. e28229, 2021.
[50] Raza, S., Reji, D. J., Shajan, F., and Bashir, S. R., "Large-scale application of named entity recognition to biomedicine and epidemiology," PLOS Digital Health, vol. 1, no. 12, p. e0000152, 2022. [Online]. Available: https://doi.org/10.1371/journal.pdig.0000152
[51] Gao, S., Kotevska, O., Sorokine, A., and Christian, J. B., "A pre-training and self-training approach for biomedical named entity recognition," PloS One, vol. 16, no. 2, p. e0246310, 2021.
[52] Liu, S., Wang, A., Xiu, X., Zhong, M., Wu, S., et al., "Evaluating medical entity recognition in health care: Entity model quantitative study," JMIR Medical Informatics, vol. 12, no. 1, p. e59782, 2024. [Online]. Available: https://doi.org/10.2196/59782
[53] Akhtyamova, L., "Named entity recognition in Spanish biomedical literature: Short review and BERT model," in 2020 26th Conference of Open Innovations Association (FRUCT), 2020, pp. 1–7.
[54] Peng, K., Yin, C., Rong, W., Lin, C., Zhou, D., and Xiong, Z., "Named entity aware transfer

learning for biomedical factoid question answering," IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 19, no. 4, pp. 2365–2376, 2021. [Online]. Available: https://doi.org/10.1109/TCBB.2021.3078724

[55] Symeonidou, A., Sazonau, V., and Groth, P., "Transfer learning for biomedical named entity recognition with BioBERT," in SEMANTiCS (Posters & Demos), 2019.

[56] Liu, S., Wang, A., Xiu, X., Zhong, M., and Wu, S., "Evaluating Medical Entity Recognition in Healthcare: A Comprehensive Analysis of BERT-Based Models," Journal of Medical Informatics, 2024.