INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2025, VOL. 71, NO. 3, PP. 1–6
Manuscript received April 2, 2025; revised June 2025. doi: 110.24425/ijet.2025.153623

Resilience of Named Entity Recognition models against adversarial attacks

Paweł Walkowiak

Abstract—**Adversarial Attacks are actions that aims to mislead models by introducing subtle and often imperceptible changes in model's input. Providing resilience for such kind of risk is key for all Natural Language Processing (NLP) task specific models. Current state of the art solution for one of NLP task Named Entity Recognition (NER) is usage of transformer based solutions. Previous solution where based on Conditional Random Fields (CRF).This research aims to investigate and compare the robustness of both transformer-based and CRF-based NER models against adversarial attacks. By subjecting these models to carefully crafted perturbations, we seek to understand how well they can withstand attempts to manipulate their input and compromise their performance. This comparative analysis will provide valuable insights into the strengths and weaknesses of each architecture, shedding light on the most effective strategies for enhancing the security and reliability of NER systems.**

Keywords—**Named Entity Recognition, Polish, Adversarial Attacks**

I. Introduction

AN adversarial attack involves generating new, subtly altered samples designed to cause incorrect behavior in machine learning (ML) models. These minimal often imperceptible changes exploit weaknesses in the models, leading to errors. Adversarial attacks aim to take advantage of vulnerabilities in ML models, causing them to fail in various tasks such as image classification, natural language processing (NLP) or autonomous driving. The raise in different adversarial example creation techniques has sparked concerns about the robustness and security of ML systems adversarial attacks can be classified as white-box or black-box based on the attacker's knowledge about victim model. White-box attacks allow complete access to the model's architecture, parameters, and gradients, enabling the creation of highly targeted adversarial examples. In contrast, black-box attacks provide limited or no access to the model, relying instead on its input-output behavior, which makes generating effective adversarial examples more difficult but potentially more applicable to real-world scenarios. A more detailed taxonomy of these attacks is available in [1].

Named Entity Recognition (NER) is a sequence labeling task in natural language processing (NLP), where the model identifies and classifies proper names within a given text. A recent survey of the state of the art in this area was discussed in [2]. Traditional attack methods used for classifiers cannot be directly applied to NER models, as sequence labeling produces different outputs, making it difficult to assess the success or failure of an attack using typical attack evaluation metrics. Various approaches have been proposed, including altering the boundaries of proper names or modifying the classification labels of correctly identified entities. Attacks on NER models pose significant risks, particularly because NER is often a component of larger NLP systems, such as anonymization pipelines. In these scenarios, an attacker could manipulate documents to cause confidential information to leak during the anonymization process. This risk underscores the importance of selecting the appropriate NER model for a specific application. Beyond evaluating performance in terms of speed and accuracy on clean test data, it's also crucial to consider the model's resilience to adversarial attacks. Our contribution is to investigate which NER models are more robust against various types of adversarial attacks and whether improving robustness compromises accuracy. In this work we aim to compare the behavior of Conditional Random Fields (CRF) and Transformer-based NER models, focusing on Polish language models, including the CRF based Liner2 [3] and the Transformer based WiNER[1].

II. Related Work

Adversarial examples are a well known threat to NLP classification models. One of methods that allows to craft such kind of samples is TextFooler [4]. This method is based on word level substitutions in original texts. Candidates for being substituted are obtained by measuring changes in model prediction under masking each of the text words. Candidates with highest model prediction change are chosen and substituted with their synonyms gathered with usage of GloVe [5] word vectors. Additionally to preserve text meaning, adversarial texts are filtered with usage of cosine similarity on SentenceBERT [6] embeddings.

One of the other approaches of adversarial examples creations is TextBugger, presented in the paper [7]. It relies on determining the ranking of word importance based on

The work was financed as part of the investment: ”CLARIN ERIC – European Research Infrastructure Consortium: Com- mon Language Resources and Technology Infrastructure (pe- riod: 2024-2026) funded by the Polish Ministry of Science and Higher Education (Programme: ”Support for the participation of Polish scientific teams in international research infrastructure projects”), agreement number 2024/WK/01.

P. Walkowiak, Wroclaw University of Science and Technology, Wrocław, Poland (e-mail: pawel.walkowiak@pwr.edu.pl).

[1] https://wiki.clarin-pl.eu/pl/nlpservices/list/winer

the Jacobian, calculated with usage of classification function and model's input. Based on the ranking, disturbances are introduced into important words to prepare an adversarial sample. The paper proposes five disturbance methods: insertion: which involves inserting spaces into the disturbed word, delete: removes a random letter, swap: swaps the order of two adjacent letters, Sub-C: replaces letters with visually similar ones (e.g., ”l” with ”1”), and Sub-W: replaces a word with its closest neighbor based on a pretrained GloVe. The approach proposed in the TextBugger method is characterized by high diversity of generated adversarial examples due to the randomness introduced in word modifications and the possibility of applying multiple disturbance techniques simultaneously.

Earlier research titled Breaking BERT [8] examined vulnerabilities in BERT-based models [9], focusing on named entity recognition in specific domains. The study evaluated how different BERT variants responded to various attack methods, such as replacing words within entity contexts and substituting them with similar proper nouns. To ensure semantic similarity, the authors utilized the Universal Sentence Encoder [10] (USE) with a minimum similarity threshold of 0.8, eliminating candidates that fell below this score. They introduced two evaluation metrics: the proportion of incorrectly labeled entities after an attack and partial mislabeling, which is especially relevant for attacks targeting entity contexts. These metrics provide a more intuitive extension of traditional measures for assessing the success of single-label attacks.

The SeqAttack framework [11] introduced techniques for generating and evaluating adversarial samples specifically for NER models, along with methods for adversarial training. It builds on the TextAttack library [12], which was originally designed for classification models, and provides a variety of adversarial strategies that operate at different levels, including character, word, and sentence manipulations. In their experiments, the authors focused on BERT-based models and employed untargeted attacks. They discovered that word-bugging techniques were the most effective, although word-level replacements produced more coherent adversarial examples. To assess the effectiveness of these attacks, the authors utilized several evaluation metrics: the number of samples needed to mislead the model, the percentage of modified tokens, the occurrence of grammatical errors, and the level of textual similarity. For measuring similarity between the original and adversarial samples, they used cosine similarity based on USE embeddings.

III. Adversarial Examples for NER

The methodology for creating adversarial examples in Named Entity Recognition tasks was thoroughly detailed in the Breaking BERT [8] paper, which served as a foundation for our experiments with two types of adversarial attacks: **Entity Attack** and **Entity Context Attack**. However, given our emphasis on Polish language models, we needed to modify the approach to find appropriate word synonyms, which we accomplished by utilizing pretrained, uncontextual word embeddings from the fastText [13] model, specifically its Polish vector model. Additionally, since the models we analyzed were accessed via API without providing model logits, we devised an alternative approach that involves masking words and assessing changes in proper nouns. Furthermore, rather than relying solely on the embeddings from Universal Sentence Encoder to maintain sentence similarity, we used the Sentence-BERT model with a minimum cosine similarity threshold ϵ.

The **Entity Attack** method is designed to target specific labeled entities within a text by substituting them with alternative named entities that belong to the same annotation class. The goal is to test whether the model can correctly identify and classify the replaced entity, despite the actual change in proper name. To achieve this, the method leverages, a pool of candidate labels from other dataset samples, ensuring that the replacement entity shares the same class as the original. For instance, consider the example sentence: ”My name is Michael Lee and I am doing my PhD.” where ”Michael Lee” is annotated as a living person (nam_liv_person according to KPWr categorization classes). A potential substitution candidate with the same class could be ”Smith Doe”, resulting in the modified sentence: ”My name is Smith Doe and I am doing my PhD.” Ideally, the model should recognize ”Smith Doe” as a valid replacement for ”Michael Lee” and assign it the correct class label. This approach has the flexibility to replace multiple proper names within a single sample simultaneously. However, for the purpose of these experiments, the focus is on replacing one proper name at a time to isolate the impact of the attack on the model's performance. By doing so, the researchers aim to assess the robustness of the model in handling subtle changes to the input data while maintaining accuracy in entity recognition.

The **Entity Context Attack** method targets the neighborhood of proper names in a sentence, aiming to induce partial or complete changes in the model's sequence labeling predictions. To identify suitable substitution candidates for the attack, the original methodology employs a technique of word masking, where each non-named entity word is temporarily hidden and the model's confidence scores (logits) are compared to the original prediction. The change in model prediction confidence represents the importance of masked word in model's output. However, this approach is not feasible when working with models like Liner2 and WiNER, which exhibit black-box characteristics, making their internal workings opaque. To overcome this limitation, we developed an alternative strategy to generate context aware candidates. We ran the models with individual non-named entity words masked, one at a time, and compared the model's output to the original prediction. If the masked word resulted in a change to the predicted entities, it received a score equal to the number of changed entities plus one; otherwise, it scored one. Components of proper names were assigned a score of zero, indicating they were less relevant to the model's output. By sorting the words according to their scores, we prioritized those that had the most significant impact on the model's predictions, effectively identifying the most critical words in the context surrounding the proper names. This innovative approach enables us to systematically analyze the relationships between words in the sentence and their influence on the model's entity recognition capabilities. By targeting the most influential words first, we can simulate realistic scenarios where small changes to the input data might lead to significant errors in the model's output, ultimately evaluating the robustness of the model under various contextual perturbations.

A. *Evaluation of model robustness*

Evaluation of named entity recognition robustness against adversarial examples is an complicated tasks. For our experiments we have collected a set of metrics that shows how well model stands against attacks. First of them are standard **accuracy (ACC)** and **F1-score (F1)** metrics, but with a modification where differences are calculated across all labels, including ”non-proper name” labels. Each sample's sequence of labels is combined into a single, flattened list before being inputted into the metric calculations.

For evaluating Named Entity Recognition (NER) models in our experiments, we selected three key metrics: the ratios of completely omitted proper names, partially omitted proper names, and misclassified proper names. The three metrics are evaluated for each individual data sample, and then aggregated at the dataset level using two statistical measures: mean and standard deviation. The mean provides a single, representative value for the entire dataset, while the standard deviation captures the variability or dispersion of the metric values across the samples. Evaluation metrics are described in detail below:

- The **Omitted Proper Names** metric calculates the proportion of omitted proper names out of total proper names within each dataset sample. When a sample has multiple evaluation results (such as being tested with multiple adversarial examples), the average value is used. This metric helps evaluate a model's resilience to adversarial attacks, particularly in named entity recognition tasks.
- The **Partially omitted proper names** metric measures the proportion of full proper names that were only partially identified by the model. For instance, if the correct name is ”Lower Silesia”, but the model only recognizes ”Silesia” as an entity while missing ”Lower”. This metric is partially connected with misclasification number, because error in obtaining proper name bound cant cause unplanned change in predicted class, for example ”Joe Doe” would be classified as living person, but ”Doe” matched name class.
- When a model incorrectly predicts a label for a sequence while correctly identifying the boundaries of the proper name, this discrepancy is counted towards the **NTypes** metric. Specifically, if the predicted label does not match the actual label, but the start and end positions of the proper name are accurately detected, it is considered a misclassification. The NTypes metric then aggregates these instances across all samples and calculates the average rate at which such misclassifications occur relative to the total number of actual proper names present in the data.

B. *Stages of experiments*

A visual representation of the experimental pipeline, comprising multiple stages, is illustrated in Figure 1. This pipeline outlines the sequential steps involved in conducting our experiments, which encompass the following key components:

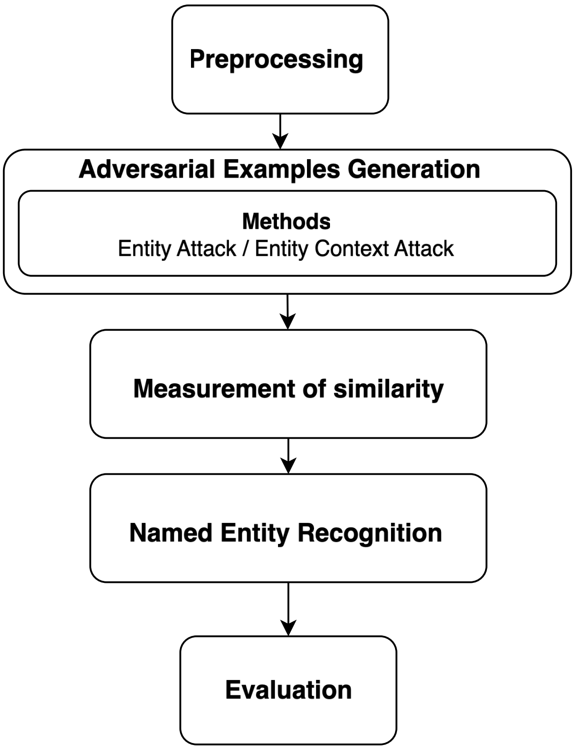
Preprocessing
Adversarial Examples Generation
Methods
Entity Attack / Entity Context Attack
Measurement of similarity
Named Entity Recognition
Evaluation

Fig. 1. Pipeline of steps in the experiments. Consists of several interconnected stages that systematically guide the progression of our research, from initial preparation to final analysis.

1) **Preprocessing** of texts from datasets, which includes POS tagging and excluding examples without any proper names.
2) **Generation of adversarial examples** with one of chosen methods either entity or conxtex attack.
3) **Measuring of similarity** to preserve high semantic similarity between adversarial examples and unchanged texts.
4) **Named Entity Recognition** with one of examined models WiNER or Liner2.
5) **Evaluation** of model's quality after attacks, to check how robust their are against each kind of adversarial examples creation method. Evaluation includes metrics such as F1-score; accuracy; omitted, partially omitted and misclassified proper names.

This structured approach allows for a systematic evaluation of our methodology, ensuring a comprehensive analysis of the results obtained.

IV. Models and Datasets

A. *Models*

Liner2 [14] is a framework designed for tackling various sequence labeling tasks. In this context, it has been applied to NER, as previously discussed in [3]. The framework assumes preprocessed text inputs with existing morphological tags. While the author suggests that incorporating morphological analysis into sequence labeling models can enhance performance, its impact on NER tasks appears limited. Liner's processing pipeline comprises three key components: a Conditional Random Fields (CRF) model trained on annotated data; a set of heuristics for merging, grouping, and filtering categories; and another set of heuristics for named entity lemmatization. The CRF model incorporates a wide range of input features, including orthographic, pattern-based, morphological, lexical, and WordNet-derived information. This study specifically focuses on utilizing Liner's fine-grained ”n82” model variant, which was trained on the KPWr [15] dataset. For Liner2 prepossessing with part of speech (POS) tagging is needed. The tool used for this task in our experiments was **MorphoDiTa**, originally presented as Czech language tagger in paper [16]. In the experiments we used its Polish version, adopted for NKJP tagset [17] and deployed in CLARIN-PL infrastructure[2]. MorphoDiTa features a comprehensive morphological dictionary that associates lemmas with their corresponding tags for in-depth analysis. The system utilizes a heuristic approach to identify patterns connected to a word's prefix and suffix, generating a range of potential lemma-tag combinations. These combinations are then refined by a POS tagger, which employs a averaged perceptron to resolve any ambiguities.

Transformer-based models have revolutionized the field of named entity recognition by leveraging their ability to capture contextual information and long-range dependencies within text. Unlike traditional models that relied heavily on handcrafted features and shallow architectures, transformers, such as BERT [9] and its variants, utilize self-attention mechanisms to process entire sequences of text simultaneously. This allows them to understand the nuances of language, including polysemy and context-specific meanings, which are crucial for accurately identifying entities like names, organizations, and locations. Finetuning these pretrained models on NER datasets has led to significant improvements in performance, enabling systems to achieve state of the art results across various languages and domains. As a result, transformer-based approaches have become the standard in NER tasks. **WiNER** is a transformer based NER model specifically designed for the Polish language. It leverages the pretrained Polish RoBERTa [18] as its foundation. Although WiNER does not require part-of-speech (POS) tagged text as input, its performance is enhanced when provided with text that adheres to the NKJP tagset standards for tokenization and sentence segmentation. Notably, both Liner2 and WiNER conform to the KPWr categorization scheme, yielding sequence labeling results across 62 distinct classes.

B. *Polish NER Datasets*

To evaluate the performance of the named entity recognition (NER) models, specifically WiNER and Liner2, two datasets were selected for testing KPWr [15] and CEN [19]. These datasets were chosen because they align with the training datasets used for the models, allowing for a fair and relevant assessment of their performance. Sizes of each dataset split are presented in table I.

- **KPWr** corpus is licensed under Creative Commons and contains a diverse range of texts from various genres, including blogs, science, and law articles. The texts are manually annotated with multiple layers of information, such as: Part-of-speech tags Predicate-argument relations Word senses Named entities This comprehensive annotation makes KPWr suitable for various sequence labeling tasks.
- **CEN** (Corpus of Economic News) consists of texts from Polish Wikipedia focused on economics, annotated with 65 categories of proper names. CEN provides an alternative domain specific resource, complementing KPWr, and is particularly useful for evaluating models on economic related named entity recognition tasks.

TABLE I
OVERVIEW OF DATASET SPLIT SIZES AND DETAILS ON TEST AND VALIDATION SETS AFTER NER FILTERING

Dataset	Train	Test		Validation	
		Base	Filtered	Base	Filtered
CEN	5,800	902	624	875	608
KPWr	9,210	4,323	2,192	4,748	2,319

V. ATTACKS RESILIENCE RESULTS

The experimental setup involves a two step process: first, text is tagged using **MorphoDiTa**, and then either **WiNER** or **Liner2** is employed for named entity recognition. An evaluation of the models performance was conducted using specific datasets and metrics, with the results presented in tables II and III. The findings reveal that WiNER consistently outperforms Liner2, achieving higher accuracy and F1-scores across both datasets, with a margin of 3-4 percentage points. A closer examination of the detailed metrics, particularly the omitted entities percentage, supports this observation, showing that Liner2 exhibits significantly higher rates (6-11 times more) compared to WiNER. However, it is noteworthy that WiNER displays unusually high percentages for partial omission and NTypes errors, suggesting potential difficulties in distinguishing between different types and parts of proper names.

A. *Attacks on Entities*

Table II presents the outcomes of an entity attack on WiNER and Liner2 models across two datasets. In this experiment, a single proper name was altered in each adversarial sample, and the replacement options were capped at five. Notably, all model-dataset combinations experienced a significant decline in accuracy and F1-score, plummeting from above 85% to around 71-76%. WiNER suffered the steepest drops, with its F1-score decreasing by 18 points on KPWr and 17 points on CEN, possibly due to its exceptional performance on unaltered samples. An examination of detailed metrics in table IV revealed WiNER's susceptibility to this kind of attack, marked by a sharp rise in Omit and Partial Omits rates. Conversely,

[2] https://services.clarin-pl.eu/dashboard

TABLE II
Results of entity attacks on Liner2 and WiNER models using KPWr and CEN datasets are presented, including a range of evaluation metrics. The accuracy and F1-score columns show the percentage change in these metrics compared to their original, unattacked values.

Dataset	Model	Non-attacked		Attacked	
		ACC [%]	F1 [%]	ACC [%]	F1 [%]
KPWr	Liner2	86	86	74 [-8]	76 [-10]
KPWr	WiNER	88	89	72 [-16]	71 [-18]
CEN	Liner2	88	88	**75** [-13]	**76** [-12]
CEN	WiNER	92	92	**75** [-17]	75 [-17]

Liner2 exhibited minimal changes in omitted proper names, albeit with a pronounced shift toward partial omissions. It's worth noting, however, that the standard deviation was remarkably high, rivaling the mean value. Furthermore, the reduction in changed entity types for both models could be attributed to the fact that many prior errors went undetected or partially detected when processing adversarial samples.

TABLE III
Results of entity attacks on Liner2 and WiNER models using KPWr and CEN datasets are presented, including a range of detailed evaluation metrics for unattacked samples

Dataset	Model	Omit [%] ↓	P. Omit [%] ↓	NTypes [%] ↓
KPWr	Liner2	33.61 ± 4.37	6.21 ± 2.66	13.38 ± 2.05
KPWr	WiNER	5.14 ± 1.00	8.41 ± 3.63	15.53 ± 3.17
CEN	Liner2	23.76 ± 3.11	6.10 ± 1.11	12.12 ± 2.09
CEN	WiNER	**1.69** ± 0.57	**4.52** ± 1.25	**8.09** ± 1.35

B. Attacks on Context

The comparison of the models F1-score and accuracy after context attack to its original unattacked level are summarized in Table V. To generate adversarial examples, one neighboring word of a proper name was swapped at a time, creating five samples per named entity. The selection of context words was informed by previous model runs that assessed word importance after deleting candidate words. As anticipated, basic metrics like accuracy and F1-score declined, with the most notable drop observed in WiNER on the KPWr dataset, where both accuracy and F1-score fell by 4 percentage points. A closer analysis presented in the table VI reveals that the transformer-based model more frequently missed parts of proper names or misclassified their types compared to unattacked samples. Similar to entity attacks, the metrics assessing omission and type changes exhibit large standard deviations, particularly for WiNER, where the standard deviation surpasses the mean value. This suggests that the transformer based model is highly susceptible to contextual changes. Moreover, the high standard deviation for WiNER indicates that certain context attack samples substantially alter the model's decision making process, leading to inconsistent outcomes.

VI. Conclusions

This article focuses on comparing two models and their architectures: Liner2, which is based on a Conditional Random Field, and WiNER, which leverages RoBERTa. The evaluation, conducted using adversarial examples generated by the entity attack method, revealed that the WiNER model is more vulnerable to such attacks than the CRF based Liner2. Both models experienced a drop in evaluation metrics, but Liner2 demonstrated better resilience when compared to its performance on an unperturbed dataset. In context based attacks, both models were less impacted, though WiNER still showed the largest accuracy decline, particularly on the KWPr dataset. The high standard deviation in WiNER's detailed metrics suggests significant distortion in its predictions for certain adversarial examples. Additionally, modifying the context of proper names led to a shift in the types of mistakes, such as partial omissions and changes in entity types. In summary, although the transformer based WiNER outperforms Liner2 on clean (unattacked) data, its performance under attack was equal to or worse than Liner2. Therefore, based on the experiments, the CRF based Liner2 appears to be more robust against the tested adversarial attacks.

TABLE IV
Results of entity attacks on Liner2 and WiNER models using KPWr and CEN datasets are presented, including a range of detailed evaluation metrics for attacked samples

Dataset	Model	Omit [%] ↓	P. Omit [%] ↓	NTypes [%] ↓
KPWr	Liner2	22.23 ± 22.61	**16.98** ± 17.54	**4.17** ± 4.29
KPWr	WiNER	8.67 ± 9.29	22.17 ± 24.05	6.45 ± 7.60
CEN	Liner2	23.85 ± 19.18	19.49 ± 15.73	5.59 ± 4.54
CEN	WiNER	**6.47** ± 8.01	20.25 ± 24.84	3.45 ± 4.45

TABLE V
Liner2 and WiNER model performance is evaluated under context attacks on KPWr and CEN datasets, with various metrics reported. Percentage changes in accuracy and F1-score relative to baseline unattacked scores are provided.

Dataset	Model	Non-attacked		Attacked	
		ACC [%]	F1 [%]	ACC [%]	F1 [%]
KPWr	Liner2	86	86	85 [-1]	84 [-2]
KPWr	WiNER	88	89	84 [-4]	85 [-4]
CEN	Liner2	88	88	87 [-1]	86 [-3]
CEN	WiNER	92	92	**90** [-2]	**91** [-1]

A. Future Works

Current attack methods predominantly focus on BERT based models, either by altering the context surrounding an entity or by modifying the entity itself. This manipulation creates discrepancies between the proper name and its contextual information, which can mislead BERT based models into making erroneous predictions. In contrast, our future research intends to pivot towards the development of innovative attack strategies that specifically leverage the distinctive features of Conditional Random Fields (CRFs).

By gaining a deeper understanding of the operational mechanisms of CRFs, we aim to craft more advanced adversarial attacks that exploit their unique characteristics. This approach will not only enhance our understanding of CRF vulnerabilities but also provide a rigorous evaluation of their robustness against adversarial inputs. We believe that by focusing on CRFs, we can uncover new insights into how these models process information and identify potential weaknesses that

TABLE VI
THE TABLE PRESENTS THE IMPACT OF CONTEXT ATTACKS ON LINER2 AND WINER MODELS USING KPWR AND CEN DATASETS, WITH DETAILED EVALUATION METRICS.

Dataset	Model	Omit [%] ↓	P. Omit [%] ↓	NTypes [%] ↓
KPWr	Liner2	28.52 ± 22.92	14.47 ± 12.07	14.97 ± 12.06
KPWr	WiNER	4.06 ± 5.50	13.50 ± 18.21	16.59 ± 21.90
CEN	Liner2	26.86 ± 19.98	12.81 ± 9.70	19.69 ± 15.08
CEN	WiNER	**2.21** ± 3.70	**9.87** ± 15.55	**13.64** ± 21.55

could be targeted in real world applications. Ultimately, our goal is to contribute to the advancement of model resilience, ensuring that CRF based systems can better withstand adversarial challenges.

REFERENCES

[1] Y. Wang, T. Sun, S. Li, X. Yuan, W. Ni, E. Hossain, and H. V. Poor, “Adversarial Attacks and Defenses in Machine Learning-Powered Networks: A Contemporary Survey,” *CoRR*, vol. abs/2303.06302, 2023.

[2] I. Keraghel, S. Morbieu, and M. Nadif, “A survey on recent advances in named entity recognition,” *CoRR*, vol. abs/2401.10825, 2024.

[3] M. Marcińczuk, J. Kocoń, and M. Oleksy, “Liner2 — a Generic Framework for Named Entity Recognition,” in *Proceedings of the 6th Workshop on Balto-Slavic Natural Language Processing*. Association for Computational Linguistics, 2017, pp. 86–91.

[4] D. Jin, Z. Jin, J. T. Zhou, and P. Szolovits, “Is BERT Really Robust? Natural Language Attack on Text Classification and Entailment,” *CoRR*, vol. abs/1907.11932, 2020. [Online]. Available: http://arxiv.org/abs/1907.11932

[5] J. Pennington, R. Socher, and C. Manning, “GloVe: Global Vectors for Word Representation,” in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1532–1543.

[6] N. Reimers and I. Gurevych, “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks,” in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, K. Inui, J. Jiang, V. Ng, and X. Wan, Eds. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 3982–3992.

[7] J. Li, S. Ji, T. Du, B. Li, and T. Wang, “Textbugger: Generating adversarial text against real-world applications,” in *26th Annual Network and Distributed System Security Symposium, NDSS 2019, San Diego, California, USA, February 24-27, 2019*. The Internet Society, 2019. [Online]. Available: https://www.ndss-symposium.org/ndss-paper/textbugger-generating-adversarial-text-against-real-world-applications/

[8] A. Dirkson, S. Verberne, and W. Kraaij, “Breaking BERT: Understanding its Vulnerabilities for Named Entity Recognition through Adversarial Attack,” *CoRR*, vol. abs/2109.11308, 2021. [Online]. Available: https://arxiv.org/abs/2109.11308

[9] J. Devlin, M.-W. Chang, K. Lee, and K. 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)*, J. Burstein, C. Doran, and T. Solorio, Eds. Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186. [Online]. Available: https://aclanthology.org/N19-1423

[10] D. Cer, Y. Yang, S.-y. Kong, N. Hua, N. Limtiaco, R. St. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, B. Strope, and R. Kurzweil, “Universal Sentence Encoder for English,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, E. Blanco and W. Lu, Eds. Brussels, Belgium: Association for Computational Linguistics, Nov. 2018, pp. 169–174.

[11] W. Simoncini and G. Spanakis, “SeqAttack: On Adversarial Attacks for Named Entity Recognition,” in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, H. Adel and S. Shi, Eds. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 308–318.

[12] J. Morris, E. Lifland, J. Y. Yoo, J. Grigsby, D. Jin, and Y. Qi, “TextAttack: A Framework for Adversarial Attacks, Data Augmentation, and Adversarial Training in NLP,” in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2020, pp. 119–126.

[13] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, “Bag of Tricks for Efficient Text Classification,” in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*. Association for Computational Linguistics, April 2017, pp. 427–431.

[14] M. Marcińczuk, J. Kocoń, and M. Janicki, “Liner2 – A Customizable Framework for Proper Names Recognition for Polish,” vol. 467, pp. 231–253, 01 2013.

[15] B. Broda, M. Marcińczuk, M. Maziarz, A. Radziszewski, and A. Wardyński, “KPWr: Towards a Free Corpus of Polish,” in *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC’12)*, N. Calzolari, K. Choukri, T. Declerck, M. U. Doğan, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, and S. Piperidis, Eds. Istanbul, Turkey: European Language Resources Association (ELRA), May 2012, pp. 3218–3222. [Online]. Available: http://www.lrec-conf.org/proceedings/lrec2012/pdf/965_Paper.pdf

[16] J. Straková, M. Straka, and J. Hajič, “Open-Source Tools for Morphology, Lemmatization, POS Tagging and Named Entity Recognition,” in *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. Baltimore, Maryland: Association for Computational Linguistics, June 2014, pp. 13–18. [Online]. Available: http://www.aclweb.org/anthology/P/P14/P14-5003.pdf

[17] A. Patejuk and A. Przepiórkowski, “ISOcat Definition of the National Corpus of Polish Tagset,” 01 2010.

[18] S. Dadas, M. Perelkiewicz, and R. Poswiata, “Pre-training Polish Transformer-Based Language Models at Scale,” in *Artificial Intelligence and Soft Computing - 19th International Conference, ICAISC 2020, Zakopane, Poland, Proceedings, Part II*, ser. Lecture Notes in Computer Science, vol. 12416. Springer, 2020, pp. 301–314.

[19] M. Marcińczuk, “CEN,” 2007, CLARIN-PL digital repository. [Online]. Available: http://hdl.handle.net/11321/6