Decoding the Obfuscated: Advanced NER Techniques on Commercial Sex Advertisement Data

Alejandro Rodriguez Perez
School of Eng. & Computer Science
Dept. of Computer Science
Baylor University
Email: Alejandro_Rodriguez4@Baylor.edu

Pablo Rivas , *Senior, IEEE*
School of Eng. & Computer Science
Dept. of Computer Science
Baylor University
Email: Pablo_Rivas@Baylor.edu

Javier Turek
Human-Robot Collaboration Lab
Intelligent Systems Research
Intel Labs
Email: Javier.Turek@Intel.com

Korn Sooksatra
School of Eng. & Computer Science
Dept. of Computer Science
Baylor University
Email: Korn_Sooksatra1@Baylor.edu

Ernesto Quevedo
School of Eng. & Computer Science
Dept. of Computer Science
Baylor University
Email: Ernesto_Quevedo1@Baylor.edu

Gisela Bichler
School of Criminology & Criminal Justice
California State University
San Bernardino
Email: gbichler@csusb.edu

Tomas Cerny
College of Engineering
Dept. of Systems and Industrial Eng.
The University of Arizona
Email: tcerny@arizona.edu

Laurie Giddens
Information Tech. & Decisions Sci. Dept.
G. Brint Ryan College of Business
University of North Texas
Email: Laurie.Giddens@unt.edu

Stacie Petter
School of Business
Wake Forest University
Email: petters@wfu.edu

***Abstract*—Detecting potential human trafficking activity within online Commercial Sex Advertisements (CSAs) presents unique challenges for Named Entity Recognition (NER) due to the complex and often obfuscated textual content. This paper thoroughly evaluates state-of-the-art language models and tokenization techniques, focusing on their efficacy in the NER task within the context of CSAs. Our results indicate that the Longformer model, equipped with byte-level BPE tokenization, outperforms other models regarding precision, recall, and the F_1 score. The study also uncovers specific areas for improvement, offering avenues for future research. Our findings have significant implications for the automated analysis and monitoring of CSAs for suspected human trafficking activity, which contributes to developing more robust and transparent online ecosystems.**

***Index Terms*—named entity recognition, transformer, natural language processing**

I. Introduction

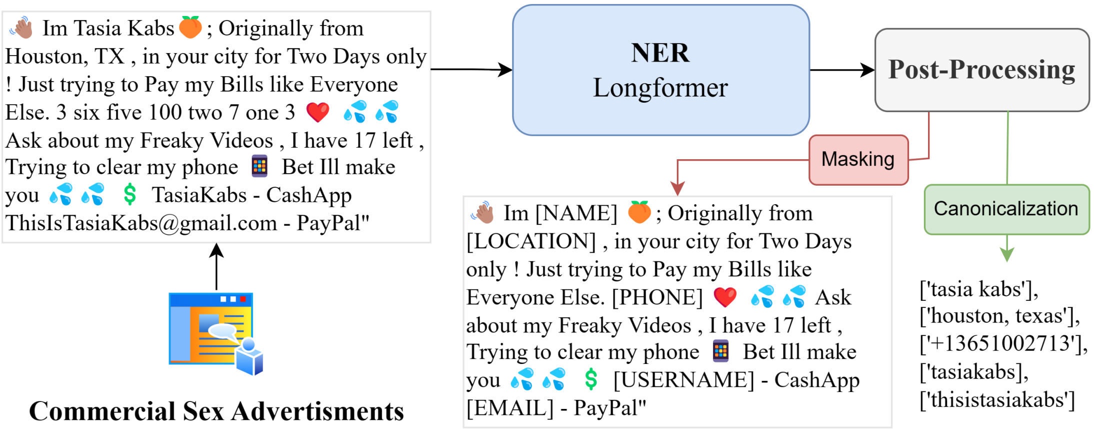
Named Entity Recognition (NER) is a crucial component in Natural Language Processing (NLP) and plays a vital role in various applications such as information extraction, machine translation, and question-answering systems [1]–[3]. However, deploying NER systems in real-world scenarios faces challenges due to noisy data, which can arise from human errors, system malfunctions, and adversarial manipulations [4], [5]. This challenge becomes particularly significant in the dynamic context of identifying suspected human trafficking activity within a large set of online Commercial Sex Advertisements (CSAs), where text is intentionally obfuscated to evade law enforcement and automated detection systems [6]. Traditional NER models perform well in controlled environments and achieve high performance on standard benchmarks [7]. However, they struggle when exposed to noise, limiting their effectiveness in high-stakes applications that require precise information extraction.

To address this gap, researchers have explored various methods to develop NER systems that are accurate and robust against noisy and adversarial data. These methods range from rule-based and dictionary-based techniques to advanced machine learning models [8], [9]. However, there is still a need for novel approaches that specifically cater to the demands posed by noisy data in CSAs.

In this paper, we propose a novel NER system (Fig. 1) tailored for the challenges posed by noisy data in CSAs. Our approach utilizes state-of-the-art Transformer-based models, fine-tuned on a curated dataset, to achieve high precision and recall in entity extraction. Our methodology not only surpasses existing approaches but also provides valuable insights into the resilience of the model against adversarial manipulations to obfuscate information, such as the encoded phone number appearing in Fig. 1 [10].

By developing a robust NER system for CSAs, we aim to enhance the accuracy and reliability of information extraction in high-stakes, real-world applications. Our research contributes to the growing body of work that addresses the challenges of noisy and adversarial data in NER systems, paving the way for more effective and reliable information extraction in various domains.

The rest of the paper is organized as follows. First, the following section explains the motivation and significance



👋🏽 Im Tasia Kabs 🍑 ; Originally from Houston, TX , in your city for Two Days only ! Just trying to Pay my Bills like Everyone Else. 3 six five 100 two 7 one 3 ❤️ 💦 💦 Ask about my Freaky Videos , I have 17 left , Trying to clear my phone 📱 Bet Ill make you 💦 💦 💲 TasiaKabs - CashApp ThisIsTasiaKabs@gmail.com - PayPal"

👋🏽 Im [NAME] 🍑 ; Originally from [LOCATION] , in your city for Two Days only ! Just trying to Pay my Bills like Everyone Else. [PHONE] ❤️ 💦 💦 Ask about my Freaky Videos , I have 17 left , Trying to clear my phone 📱 Bet Ill make you 💦 💦 💲 [USERNAME] - CashApp [EMAIL] - PayPal"

['tasia kabs'],
['houston, texas'],
['+13651002713'],
['tasiakabs],
['thisistasiakabs']

Fig. 1: Proposed pipeline for NER-based information extraction from noisy CSAs. *Note:* The content of the ad displayed in this figure is similar to the actual content of ads in our dataset; however, all entities displayed here are fictitious.

behind our work. Next, the dataset utilized and how it was obtained is detailed in Section III. Following that, Section IV elaborates on our experimental design and the different models and their charcateristics. Section V is devoted to the different tokenizers employed by these models and their relevance to the study. Finally, the results and conclusions of our investigation are presented in sections VI and VII, respectively.

II. Motivation and Significance

Some online marketplaces enable providers to post CSAs to connect with potential buyers of sexual services [11]. The purchase and sale of sex is generally illegal within much of the United States (and in many countries). While nearly all advertisers within these marketplaces want to avoid arrest for solicitation or prostitution, there is a subset of commercial sex advertisements that are related to human trafficking activity [12]. Among federal sex trafficking cases within the United States, over half of the traffickers prosecuted used the internet to identify buyers of commercial sex [13]. Therefore, when analyzing CSAs for suspected human trafficking activity, there is a need to distinguish between advertisements posted by voluntary, independent sex workers and advertisements associated with human trafficking activity. Unfortunately, many law enforcement agencies lack the human resources needed to regularly monitor CSAs for suspected sex trafficking activity and would prefer to leverage NLP to automate and enhance the detection of this type of illicit activity [14].

The extraction of specific types of information, such as personal contact details and service locations, plays a crucial role in a data processing pipeline for projects aiming to detect criminal content in adversarial settings, specifically sets of CSAs associated with sex trafficking. These extracted entities serve as key connectors for discovering underlying structural relationships that can potentially identify suspected sex trafficking activity [12].

Prior research has made progress in addressing information extraction in adversarial settings. Kejriwal et al. [4] developed a lightweight Information Extraction (IE) system that combines a high-recall entity recognizer with a classifier for refining annotations. Chambers et al. [5] specifically investigated the challenge of extracting phone numbers in CSAs and developed Recurrent Neural Network (RNN)-based models to tackle this issue. Kapoor et al. [15] addressed the Geo-tagging problem in advertisements using integer programming, while Li et al. [16] proposed a hybrid approach that combines rule-based and dictionary-based extraction techniques with contextualized language models.

While information extraction has been extensively researched, existing models primarily demonstrate their performance on standard benchmarks. However, their application in the context of CSAs presents two significant limitations. Firstly, these models are typically trained to recognize general entities and may not align with the specific entities of interest in our study. Secondly, traditional models lack the robustness required to handle the adversarial manipulations commonly found in CSAs.

The motivation for our work stems from the pressing need to develop robust and specialized NER systems that can effectively operate in the challenging environment of CSAs. Our research aims to fill the existing gaps in the literature by introducing a novel approach that is both accurate and resilient against adversarial manipulations. By developing a specialized NER system for CSAs, we aim to enhance the accuracy and reliability of information extraction in this specific domain.

III. Dataset and Annotation Methodology

To address the problem of information extraction, we adopted a supervised learning paradigm and curated a dataset of online advertisements. The CSA data utilized in this study was scraped from the Skip The Games platform from January 2022 to April 2023. Scraping focused on a purposive set of 51 urban regions, including many of the largest cities in the US. Our scraping program is written in the Python programming language. It works by navigating to every link in the gallery of results. Once inside a post, the HTML of the post webpage is parsed, and several components are identified and extracted from the webpage body: ad title, posting date, and location. We use

TABLE I: Distribution of Entity Groups in the Original Training and Test Sets

	Number of instances	
Entity group	**Training**	**Testing**
Phone Number	1088	182
Name/Nickname	604	105
Location	474	80
Onlyfans	144	24
Snapchat	110	22
Username (Other)	85	17
Instagram	68	15
Twitter	40	7
Email	18	3
Cashapp	13	2
Pornhub	5	1
Venmo	2	-
Payment (Other)	1	-
Total	**2652**	**467**

Selenium[1] for automated browser control and the Beautiful Soup[2] library to parse HTML content.

The dataset consists of a randomly-selected subset of 1,810 annotated posts, where entities and their corresponding types were meticulously labeled. The dataset was annotated using Doccano, an open-source data labeling tool that supports various tasks, including sequence labeling. The annotation tool was deployed on an Amazon Web Services' `t2-small` instance to ensure security and reliability.

To evaluate the performance of our models, we partitioned the dataset into training and test sets, following an approximate 85/15 split ratio. The frequency of each entity group class within these splits is presented in Table I.

During the dataset's splitting process, we took special measures to ensure that underrepresented classes were adequately distributed across both the training and test sets. For example, the class "Payment (Other)" appeared only once in the dataset and was allocated to the training set. Similarly, the class "Venmo" was included in the training set due to its co-occurrence with "Payment (Other)" in the same post. The class imbalance observed in the dataset is worth noting, which prompted us to re-evaluate the class categorizations.

To address the class imbalance, we consolidated several classes, including "Username (Other)," "Cashapp," "Pornhub," "Venmo," and "Payment (Other)," into a unified class called "Username (Other)." This consolidation resulted in a revised distribution of classes, as shown in Table II.

By employing a rigorous approach to dataset creation and annotation, we aim to establish a robust foundation for developing and evaluating our information extraction models.

IV. Model Architectures and Experimental Design

In the pursuit of a robust NER system capable of handling noisy data, we employed an array of state-of-the-art Transformer-based architectures, each with its unique advantages and underlying principles. Below, we delineate the salient features and theoretical underpinnings of each model.

TABLE II: Distribution of Entity Groups in the Training and Test Sets After Class Consolidation

	Number of instances	
Entity group	**Training**	**Testing**
Phone Number	1088	182
Name/Nickname	604	105
Location	474	80
Onlyfans	144	24
Snapchat	110	22
Username (Other)	106	20
Instagram	68	15
Twitter	40	7
Email	18	3
Total	**2652**	**467**

BERT: Bidirectional Encoder Representations from Transformers.
BERT is a pioneering architecture in the NLP domain, renowned for its pre-training on a colossal corpus using a Masked Language Modeling (MLM) objective. This allows BERT to capture bidirectional context, thereby enriching the semantic understanding of language. Additionally, BERT incorporates a Next Sentence Prediction (NSP) task during its pre-training phase to further enhance its language understanding capabilities [17].

ALBERT: A Lite BERT.
ALBERT serves as a computationally efficient variant of BERT, designed to maintain or even surpass the performance of its predecessor. It achieves this efficiency through parameter-sharing strategies and factorization techniques, thereby offering a resource-efficient yet powerful alternative for large-scale NLP tasks [18].

BigBird: Sparse Attention Mechanisms.
BigBird introduces a novel global attention mechanism that enables the model to process long sequences efficiently. Unlike traditional Transformer architectures, which apply self-attention uniformly across tokens, BigBird employs a sparse attention pattern, thereby reducing the computational burden for long sequences [19].

CANINE: Character-Augmented Neural Information Encoding.
CANINE is explicitly engineered for character-level sequence processing, obviating the need for tokenization. This architecture is particularly advantageous in multilingual contexts and simplifies many engineering challenges commonly encountered in NLP [20].

GPT-2: Generative Pre-trained Transformer 2.
Developed by OpenAI, GPT-2 has gained acclaim for its ability to generate coherent and contextually relevant text. It is pre-trained on an extensive text corpus, enabling it to produce human-like language across a myriad of applications [21].

Longformer: Efficient Transformer for Long Documents.
Longformer is designed to efficiently process long documents by employing a hybrid of global and local attention mechanisms. This architecture is particularly beneficial for tasks that require an understanding of extended textual content [22].

RoBERTa: Robustly Optimized BERT Approach.
RoBERTa improves upon the original BERT by implementing

[1] https://selenium-python.readthedocs.io/index.html

[2] https://pypi.org/project/beautifulsoup4/

several training optimizations, such as extended training duration, larger batch sizes, and removing the NSP objective, among others [23].

XLNet: eXtreme Multi-Label Net.
XLNet amalgamates the strengths of both autoregressive and autoencoding language models. It employs a permutation-based training objective, allowing it to capture the bidirectional context in a manner akin to BERT [24].

A. Experimental Design

We employed a rigorous 10-fold cross-validation methodology to evaluate the performance of each model. All models underwent training for 15 epochs with a batch size of 1, utilizing the AdamW optimizer [25]. The implementation and training routines were conducted using the Transformers Python library [26], and the models were trained on a Tesla V100-PCIE-16GB GPU. Table III provides a summary of the models employed.

This comprehensive approach ensures a robust evaluation of each architecture's capabilities, thereby facilitating an informed selection of the most suitable model for NER in noisy data environments.

V. Tokenizers

The choice of tokenizer is a critical factor in evaluating the suitability of various language models for NER tasks in the context of CSAs. Traditional word-based tokenization methods, which typically segment text based on spaces or punctuation, are inadequate for several reasons. Firstly, such methods are not universally applicable across languages, as some languages like Japanese and Korean do not use spaces to delimit words. Secondly, word-based tokenization can result in an unwieldy vocabulary that includes misspellings, morphological variations, and out-of-vocabulary (OOV) words, complicating the language model's training and inference processes [27].

Some alternatives to word-based tokenization have become predominant, namely, sub-word-based and character-based tokenization. The main idea behind sub-word-based tokenization is allowing tokens to be below the level of words, i.e., fragments of words or even single characters. This allows the codification of strange words as combinations of sub-word tokens [27]. Sub-word-based tokenization algorithms are described in two sub-tasks: training and encoding. Training the tokenizer takes a corpus of text and produces, at the bare minimum, a collection of tokens called vocabulary and possibly other items. Encoding is when a trained tokenizer splits a piece of text into tokens in its vocabulary.

Some methods stand out. In Byte-Pair Encoding (BPE), the training algorithm starts with a collection of all allowed characters and pairwise merges them to create new tokens. To encode a string, it assumes it has been pre-tokenized, i.e., divided into words. The training process computed the vocabulary and preserved the order in which the merges occurred. Then, encoding a word only requires reproducing the exact same merges as during training for consistency and reproducibility of the entire data pipeline [28].

Word Piece tokenization is similar to BPE in that training starts with a collection of characters and proceeds by merging them. The difference is the merging criteria. Whereas BPE only considers the frequency of the resulting pair, Word Piece considers the pairs that maximize the likelihood of the training data when added to the vocabulary. This is the same as finding a pair so that its frequency count divided by the product of the frequency counts of its parts is maximum among all pairs. Encoding a word in Word Piece is done by iteratively finding the longest substring that matches a token in the vocabulary. If, at some point, no remaining substring exists in the vocabulary, unique tokens are employed to represent them [29].

Finally, Unigram's training process is the converse of BPE and Word Piece. It starts with a vast vocabulary and trims it at each iteration until a desired size is reached. Given the current vocabulary, a unigram language model,[3] and a loss function defined in terms of the previous, the algorithm discards a certain percentage of the tokens that cause the loss to increase the least when removed from the vocabulary. The process ends with a list of tokens and a probability assigned to each single token. This allows a joint probability distribution for each possible word tokenization, and it returns the token that has a higher likelihood (or it could also return a randomized one, depending on the use case) [30].

On top of the sub-word tokenization resides the concept of Sentence Piece tokenization. All methods mentioned before required a pre-tokenization step. Sentence Piece waives that requirement, and whitespaces are considered a regular symbol within the vocabulary. Tokenization can then occur with any of the methods mentioned before. This has advantages like lossless tokenization and end-to-end sub-word segmentation [31]. In the Transformers Python library, Sentence Piece is always used with Unigram tokenization [32].

Despite the great success of sub-word-based tokenization, recent work has highlighted limitations [33]. Some research has attempted to deviate from that paradigm by proposing more radical approaches in which authors skip tokenization altogether and consider representing the input as a sequence of individual characters [20], bytes [34], or even learning tokenization as part of the network [35].

For a comprehensive comparison of the tokenizers employed in the models utilized in this study, refer to Table IV.

VI. Results

To comprehensively evaluate the various models under consideration, we focus on the post-processed NER output rather than the preliminary token classification scores. Specifically, we assess the performance of the models based on the entities that are ultimately predicted after decoding the model's output into an organized set of identified entities, each tagged with their specific type such as phone, location, etc (see Table II). This structure facilitates a more nuanced evaluation of the model's output, beyond mere token classification accuracy.

[3] A unigram language model is a model that assigns each token a probability, e.g., by dividing the frequency count of the token by the number of tokens in the training corpus.

TABLE III: Summary of Transformer-based Models Employed in the Study, Sorted by Size.

Model	Transformers Hub Name	Reference	Parameters
ALBERT	albert-base-v2	[18]	11,094,530
BERT	bert-base-cased	[17]	107,721,218
XLNet	xlnet-base-cased	[24]	116,719,874
RoBERTa	roberta-base	[23]	124,056,578
GPT-2	gpt2	[21]	124,441,346
BigBird	google/bigbird-roberta-base	[19]	127,470,338
CANINE	google/canine-c	[20]	132,084,482
Longformer	allenai/longformer-base-4096	[22]	148,070,402

TABLE IV: Comparison of Tokenizers Employed in Various Language Models

Model	Tokenizer	Vocabulary Size	Maximum Context	Truncated Sentences	Average Unknowns
ALBERT	Sent. Piece	30,000	512	19	85.54
BERT	Word Piece	28,996	512	31	91.87
BigBird	Sent. Piece	50,358	4096	0	156.43
CANINE	Characters	1,114,112	2,048	17	0.00
GPT-2	BPE	50,257	1,024	11	0.00
Longformer	BPE	50,265	4,096	0	0.00
RoBERTa	BPE	50,265	512	51	0.00
XLNet	Sent. Piece	32,000	∞	0	71.84

The decoding algorithm employed is sourced from the Transformers library’s pipeline utility, with the aggregation strategy set to ’simple,’ which has been empirically observed to yield superior results. Only entities with a prediction confidence score exceeding 0.9 were considered to enhance the precision of the evaluation.

For evaluation, we employ a modified F_1 score that considers various types of matches: correct, partial, missing, incorrect, and spurious, as delineated in prior literature [36]. In the evaluation framework, each entity identified by the pipeline is characterized by a text span, defined by start and end indices, and an associated entity type. For each sample text, a corresponding set of ’gold standard’ entities serves as the benchmark for evaluation. Within this context, we define five distinct categories of matches: 1) “Correct Matches,” which are predicted entities that perfectly align both in text span and entity type with the gold standard; 2) “Incorrect Matches,” which refer to entities that have matching text spans but differ in entity type; 3) “Partial Matches,” denoting predicted entities that partially overlap with a gold standard entity of the same type; 4) “Missing Matches,” which are entities that are present in the gold standard but were not predicted; and 5) “Spurious Matches,” which are predicted entities that do not exist in the gold standard.

To ensure equitable scoring, each entity is counted only once, and matches are computed hierarchically: correct, incorrect, partial, missing, and then spurious. Subsequently, we calculate the precision (P) and recall (R) metrics as delineated by the following equations:

$$\text{Recall} = \frac{C + \alpha P}{C + I + P + M},$$

$$\text{Precision} = \frac{C + \alpha P}{C + I + P + S}.$$

Here, C, I, P, M, and S denote the counts of correct, incorrect, partial, missing, and spurious matches, respectively. The coefficient α, where $0 \leq \alpha < 1$, modulates the weight accorded to partial matches. In our empirical analysis, we set $\alpha = 0.5$. The F_1 score is then computed conventionally:

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}.$$

Table V presents the precision, recall, and F_1 scores for each model during both the training and validation phases. These metrics are computed by averaging the results across all 10 folds, specifically focusing on the best-performing checkpoint in each fold as determined by the evaluation set.

It is noteworthy that the Longformer and XLNet models exhibit superior performance compared to other baseline models, with Longformer marginally outpacing XLNet. As delineated in Table IV, the tokenization strategies employed by Longformer and XLNet offer advantageous characteristics, such as an extended context length that obviates the need for sentence truncation within the dataset. Furthermore, Longformer’s tokenizer did not generate any OOV tokens. This can be attributed to its utilization of byte-level Byte Pair Encoding (BPE), which commences with an exhaustive vocabulary of all conceivable bytes, thereby ensuring that each byte is retained post-training and effectively eliminating the occurrence of OOV tokens.

Contrastingly, the CANINE model underperforms significantly despite adopting Unicode-based character tokenization, which inherently avoids OOV tokens. It is posited that the expansive vocabulary resulting from encoding the entire Unicode character set may contribute to its suboptimal performance in the given context.

Moreover, it is worth highlighting that GPT-2, despite employing a tokenization strategy akin to that of RoBERTa and benefiting from a larger contextual window that minimizes sentence truncation, performs markedly worse. This discrepancy can be ascribed to the auto-regressive nature of the GPT-2 architecture, which restricts its attention to preceding tokens

TABLE V: Training and Validation Results: Average results of the 10-fold cross-validation process. Standard deviation across folds is shown in parentheses.

Model	Training			Validation		
	Precision	Recall	F_1 Score	Precision	Recall	F_1 Score
ALBERT	0.98 (0.01)	0.93 (0.01)	0.95 (0.01)	0.86 (0.03)	0.80 (0.03)	0.83 (0.02)
BERT	0.93 (0.00)	0.87 (0.01)	0.90 (0.00)	0.83 (0.03)	0.71 (0.05)	0.76 (0.03)
Big Bird	0.76 (0.01)	0.70 (0.04)	0.73 (0.02)	0.68 (0.03)	0.58 (0.03)	0.63 (0.03)
CANINE	0.99 (0.00)	0.92 (0.03)	0.95 (0.02)	0.85 (0.04)	0.74 (0.06)	0.79 (0.04)
GPT-2	0.74 (0.01)	0.70 (0.02)	0.72 (0.01)	0.62 (0.03)	0.57 (0.02)	0.59 (0.02)
Longformer	0.98 (0.01)	0.96 (0.01)	0.97 (0.01)	0.87 (0.02)	0.86 (0.03)	0.86 (0.02)
RoBERTa	0.98 (0.00)	0.94 (0.00)	0.96 (0.00)	0.87 (0.03)	0.85 (0.04)	0.86 (0.02)
XLNet	0.99 (0.00)	0.95 (0.00)	0.97 (0.00)	0.86 (0.02)	0.84 (0.03)	0.85 (0.02)
Average	0.920	0.880	0.899	0.804	0.762	0.783

TABLE VI: Testing overall and per-class performance of the Longformer model trained with full training and validation data. Sorted by F_1 score.

Class	Precision	Recall	F_1 score
Twitter	1.000	1.000	1.000
Phone Number	0.941	0.973	0.957
Onlyfans	0.850	0.708	0.773
Name/Nickname	0.775	0.738	0.756
Snapchat	0.762	0.727	0.744
Location	0.750	0.708	0.728
Instagram	0.667	0.667	0.667
Email	0.400	0.667	0.500
Username (Other)	0.429	0.300	0.353
Overall (Micro)	**0.827**	**0.804**	**0.815**

for subsequent output prediction. This is a significant limitation for tasks like NER, which inherently require a bi-directional context for effective token classification.

After the validation process, the Longformer architecture was selected for further training using the whole training and validation datasets. The retrained model was then evaluated on a distinct, held-out test set. Comprehensive performance metrics, encompassing precision, recall, and F_1-score, are elaborated in Table VI. Additionally, performance was disaggregated by individual classes, and the cumulative performance was reported as the micro-averaged metrics across all classes.

Note that the model's overall performance on the test set falls short of the estimations derived from the validation set. Particularly, the classes denoted as "Username (Other)," "Instagram," and "Email" exhibit suboptimal performance. Conversely, the model demonstrates above-average efficacy when predicting entities belonging to the "Phone Number" and "Twitter" classes.

Moreover, it is noteworthy that the precision metric surpasses the recall for the chosen threshold value. This is a salient feature in the context of our study, as a higher precision contributes to creating more meaningful associations between advertisements.

Lastly, examining the learning curves for the Longformer model, as depicted in Fig. 2, reveals a plateauing trend towards the later stages of the curve. This asymptotic behavior strongly suggests that augmenting the dataset with additional data points is unlikely to yield any further improvements in model performance.

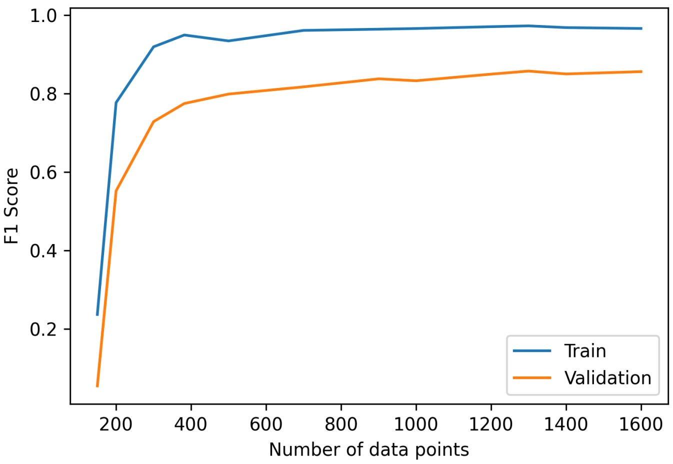


Fig. 2: Learning curves for the fine-tuned Longformer model. It is crucial to note that each data point on this plot corresponds to an individual post within the dataset, rather than a named entity.

VII. Conclusions

In this study, we have comprehensively evaluated various state-of-the-art language models and their corresponding tokenization techniques, focusing on their applicability to Named Entity Recognition (NER) in the domain of online Commercial Sex Advertisements (CSAs). The unique challenges CSAs pose, such as the intentional obfuscation of text to evade automated detection systems, necessitate a robust and adaptive NER system capable of high precision and recall.

Our results indicate that the Longformer model, with byte-level BPE tokenization, performs best overall. While the results reported in Table 5 indicate that this model performs similarly to RoBERTa in validation, the Longformer model is best in recall. Moreover, this model's ability to handle long contexts without truncation and its lack of unknown tokens make it well-suited for the complexities inherent in CSAs.

Despite the promising results, the study also revealed areas for improvement. Certain classes of entities, such as "Username (Other)", "Instagram", and "Email", exhibited suboptimal performance, warranting further investigation. Moreover, the learning curves suggest that additional data may not necessarily lead to performance gains, highlighting the need for more sophisticated techniques or model architectures to better generalize from the available data.

Our work provides valuable insights into the selection and optimization of language models for NER tasks in the challenging context when individuals seek to obfuscate their identities. In our context, sellers must share enough information about their identity to allow potential buyers to contact them for services; however, because the marketplace enables illegal activity (i.e., sale and purchase of commercial sex), the advertiser wants to avoid detection by law enforcement. A subset of individuals posting CSAs may be engaged in human trafficking, and having the ability to identify entities within CSAs is useful for identifying structures that can indicate potential human trafficking activity. The methodologies and findings presented herein advance the state of the art in NER and offer practical solutions for systematic machine learning-based labeling and the automated analysis and monitoring of CSAs,thereby offering an effective and reliable information extraction technique that can be used when individuals are seeking to obfuscate information within text.

This study, focusing on optimizing language models for NER in CSAs, forms a pivotal part of our broader ongoing research endeavors dedicated to using NLP to identify suspicious transactions in omnichannel online consumer-to-consumer marketplaces [11], [14], [37]–[39], a field where effective information extraction is key.

Ethics Statement

This study relies solely on internally scraped and curated datasets and transformer-based NER models and does not involve humans as subjects; however, the original data contains personally identifiable information that prevents the investigators from releasing the dataset to the general public. The methodology is supported by internal review board (IRB) approval at Baylor University.

Further, while we have vetted our model regarding ethical considerations, we acknowledge that it may inherit biases in the original transformer-based NER embeddings [40]–[42]. We emphasize the need for further research to mitigate these biases and are committed to methodological transparency.

Acknowledgements

This work is supported by the National Science Foundation under Grant Nos. 2210091 and 2136961. The views expressed herein are solely those of the author(s) and do not necessarily reflect those of the National Science Foundation.

References

[1] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, Łukasz Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. Corrado, M. Hughes, and J. Dean, "Google's neural machine translation system: Bridging the gap between human and machine translation," *CoRR*, vol. abs/1609.08144, 2016. [Online]. Available: http://arxiv.org/abs/1609.08144

[2] Y. Meng, Y. Zhang, J. Huang, X. Wang, Y. Zhang, H. Ji, and J. Han, "Distantly-supervised named entity recognition with noise-robust learning and language model augmented self-training," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, M.-F. Moens, X. Huang, L. Specia, and S. W.-t. Yih, Eds. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 10 367–10 378. [Online]. Available: https://aclanthology.org/2021.emnlp-main.810

[3] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," 2015.

[4] M. Kejriwal and P. Szekely, "Information extraction in illicit web domains," in *Proceedinbook title 26th International Conference on World Wide Web*, ser. WWW '17. Republic and Canton of Geneva, CHE: International World Wide Web Conferences Steering Committee, 2017, p. 997–1006. [Online]. Available: https://doi.org/10.1145/3038912.3052642

[5] N. Chambers, T. Forman, C. Griswold, K. Lu, Y. Khastgir, and S. Steckler, "Character-based models for adversarial phone extraction: Preventing human sex trafficking," in *Proceedinbook title 5th Workshop on Noisy User-generated Text (W-NUT 2019)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 48–56. [Online]. Available: https://aclanthology.org/D19-5507

[6] J. H. Zhu, "Detecting food safety risks and human tracking using interpretable machine learning methods," Ph.D. dissertation, Massachusetts Institute of Technology, 2019.

[7] S. Mayhew, G. Nitish, and D. Roth, "Robust named entity recognition with truecasing pretraining," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 05, pp. 8480–8487, Apr. 2020. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/6368

[8] E. Safranchik, S. Luo, and S. Bach, "Weakly supervised sequence tagging from noisy rules," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, pp. 5570–5578, Apr. 2020. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/6009

[9] J. Yuan and Z. He, "Adversarial dual network learning with randomized image transform for restoring attacked images," *IEEE Access*, vol. 8, pp. 22 617–22 624, 2020.

[10] K. Ren, T. Zheng, Z. Qin, and X. Liu, "Adversarial attacks and defenses in deep learning," *Engineering*, vol. 6, no. 3, pp. 346–360, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S209580991930503X

[11] L. Giddens, S. Petter, and M. H. Fullilove, "Information technology as a resource to counter domestic sex trafficking in the united states," *Information Systems Journal*, vol. 33, no. 1, pp. 8–33, 2023. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/isj.12339

[12] B. B. Keskin, G. J. Bott, and N. K. Freeman, "Cracking Sex Trafficking: Data Analysis, Pattern Recognition, and Path Prediction," *Production and Operations Management*, vol. 30, no. 4, pp. 1110–1135, April 2021. [Online]. Available: https://ideas.repec.org/a/bla/popmgt/v30y2021i4p1110-1135.html

[13] L. Lane, A. Gray, A. Rodolph, and B. Ferrigno, "Federal human trafficking report." Human Trafficking Institute, 2023.

[14] L. Giddens, S. Petter, G. Bichler, P. Rivas, M. H. Fullilove, and T. Cerny, "Navigating an interdisciplinary approach to cybercrime research," in *Proceedings of the 56th Hawaii International Conference on System Sciences*, 2023, p. 3611. [Online]. Available: https://hdl.handle.net/10125/103074

[15] R. Kapoor, M. Kejriwal, and P. Szekely, "Using contexts and constraints for improved geotagging of human trafficking webpages," in *Proceedings of the Fourth International ACM Workshop on Managing and Mining Enriched Geo-Spatial Data*, ser. GeoRich '17. New York, NY, USA: Association for Computing Machinery, 2017. [Online]. Available: https://doi.org/10.1145/3080546.3080547

[16] Y. Li, P. Nair, K. Pelrine, and R. Rabbany, "Extracting person names from user generated text: Named-entity recognition for combating human trafficking," in *Findings of the Association for Computational Linguistics: ACL 2022*, S. Muresan, P. Nakov, and A. Villavicencio, Eds. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 2854–2868. [Online]. Available: https://aclanthology.org/2022.findings-acl.225

[17] J. Devlin, M. 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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, J. Burstein, C. Doran, and

T. Solorio, Eds. Association for Computational Linguistics, 2019, pp. 4171–4186. [Online]. Available: https://doi.org/10.18653/v1/n19-1423

[18] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "ALBERT: A lite BERT for self-supervised learning of language representations," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. [Online]. Available: https://openreview.net/forum?id=H1eA7AEtvS

[19] M. Zaheer, G. Guruganesh, K. A. Dubey, J. Ainslie, C. Alberti, S. Ontanon, P. Pham, A. Ravula, Q. Wang, L. Yang, and A. Ahmed, "Big bird: Transformers for longer sequences," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 17 283–17 297. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2020/file/c8512d142a2d849725f31a9a7a361ab9-Paper.pdf

[20] J. H. Clark, D. Garrette, I. Turc, and J. Wieting, "Canine: Pre-training an Efficient Tokenization-Free Encoder for Language Representation," *Transactions of the Association for Computational Linguistics*, vol. 10, pp. 73–91, 01 2022. [Online]. Available: https://doi.org/10.1162/tacl_a_00448

[21] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[22] I. Beltagy, M. E. Peters, and A. Cohan, "Longformer: The long-document transformer," *arXiv:2004.05150*, 2020.

[23] L. Zhuang, L. Wayne, S. Ya, and Z. Jun, "A robustly optimized BERT pre-training approach with post-training," in *Proceedings of the 20th Chinese National Conference on Computational Linguistics*. Huhhot, China: Chinese Information Processing Society of China, Aug. 2021, pp. 1218–1227. [Online]. Available: https://aclanthology.org/2021.ccl-1.108

[24] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, "Xlnet: Generalized autoregressive pretraining for language understanding," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc., 2019. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2019/file/dc6a7e655d7e5840e66733e9ee67cc69-Paper.pdf

[25] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. [Online]. Available: https://openreview.net/forum?id=Bkg6RiCqY7

[26] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger, M. Drame, Q. Lhoest, and A. Rush, "Transformers: State-of-the-art natural language processing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Online: Association for Computational Linguistics, Oct. 2020, pp. 38–45. [Online]. Available: https://aclanthology.org/2020.emnlp-demos.6

[27] K. Bostrom and G. Durrett, "Byte pair encoding is suboptimal for language model pretraining," in *Findings of the Association for Computational Linguistics: EMNLP 2020*. Online: Association for Computational Linguistics, Nov. 2020, pp. 4617–4624. [Online]. Available: https://aclanthology.org/2020.findings-emnlp.414

[28] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 1715–1725. [Online]. Available: https://aclanthology.org/P16-1162

[29] M. Schuster and K. Nakajima, "Japanese and korean voice search," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 5149–5152.

[30] T. Kudo, "Subword regularization: Improving neural network translation models with multiple subword candidates," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 66–75. [Online]. Available: https://aclanthology.org/P18-1007

[31] T. Kudo and J. Richardson, "SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Brussels, Belgium: Association for Computational Linguistics, Nov. 2018, pp. 66–71. [Online]. Available: https://aclanthology.org/D18-2012

[32] Huggingface. (2023) Transformers documentation v4.33.0. [Online]. Available: https://huggingface.co/docs/transformers/tokenizer_summary#unigram

[33] S. Klein and R. Tsarfaty, "Getting the ##life out of living: How adequate are word-pieces for modelling complex morphology?" in *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*. Online: Association for Computational Linguistics, Jul. 2020, pp. 204–209. [Online]. Available: https://aclanthology.org/2020.sigmorphon-1.24

[34] L. Xue, A. Barua, N. Constant, R. Al-Rfou, S. Narang, M. Kale, A. Roberts, and C. Raffel, "ByT5: Towards a Token-Free Future with Pre-trained Byte-to-Byte Models," *Transactions of the Association for Computational Linguistics*, vol. 10, pp. 291–306, 03 2022. [Online]. Available: https://doi.org/10.1162/tacl_a_00461

[35] Y. Tay, V. Q. Tran, S. Ruder, J. Gupta, H. W. Chung, D. Bahri, Z. Qin, S. Baumgartner, C. Yu, and D. Metzler, "Charformer: Fast character transformers via gradient-based subword tokenization," in *International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=JtBRnrlOEFN

[36] A. Piad-Morffis, Y. Gutiérrez, H. Canizares-Diaz, S. Estevez-Velarde, R. Muñoz, A. Montoyo, Y. Almeida-Cruz *et al.*, "Overview of the ehealth knowledge discovery challenge at iberlef 2020," 2020.

[37] P. Rivas, G. Bichler, T. Cerny, L. Giddens, and S. Petter, "Bottleneck-based encoder-decoder architecture (bear) for learning unbiased consumer-to-consumer image representations," in *LXAI Workshop @ International Conference on Machine Learning (ICML 2022)*, 2022, pp. 1–7.

[38] A. Rodriguez Perez, K. Sooksatra, P. Rivas, E. Quevedo Caballero, J. S. Turek, G. Bichler, T. Cerny, L. Giddens, and S. Petter, "An empirical analysis towards replacing vocabulary-rigid embeddings by a vocabulary-free mechanism," in *LXAI Workshop @ International Conference on Machine Learning (ICML 2023)*, 2023, pp. 1–7.

[39] R. P. Alejandro, K. Sooksatra, P. Rivas, E. Quevedo Caballero, J. S. Turek, G. Bichler, T. Cerny, L. Giddens, and S. Petter, "Aligning word embeddings from bert to vocabulary-free representations," in *The 25th International Conference on Artificial Intelligence (ICAI 2023)*, 2023, pp. 1–8.

[40] R. Bhardwaj, N. Majumder, and S. Poria, "Investigating gender bias in bert," *Cognitive Computation*, vol. 13, no. 4, pp. 1008–1018, 2021.

[41] J. Singh, B. McCann, R. Socher, and C. Xiong, "BERT is not an interlingua and the bias of tokenization," in *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, C. Cherry, G. Durrett, G. Foster, R. Haffari, S. Khadivi, N. Peng, X. Ren, and S. Swayamdipta, Eds. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 47–55. [Online]. Available: https://aclanthology.org/D19-6106

[42] M. Mosbach, M. Andriushchenko, and D. Klakow, "On the stability of fine-tuning bert: Misconceptions, explanations, and strong baselines," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=nzpLWnVAyah