Token-Level PII Detection with Symbolic, Sequential, and Transformer-Based Ensemble Models

Hessah Abdullah Alshamrani[1], Mona Alnahari[2]
Department of Cyber Security, Taif University, Taif, Saudi Arabia[1]
Department of Human-Computer Interaction, Taif University, Taif, Saudi Arabia[2]

Abstract*—The rapid increase in unstructured digital information has led to an urgent demand for effective systems for safeguarding Personally Identifiable Information (PII) across multiple sectors and application domains. Existing single-model approaches frequently fail to resolve entity-type ambiguity in unstructured text, particularly when a token's PII status is context-dependent rather than syntactically predictable. This study presents EnsemblePII, a weighted voting ensemble that combines rule-based patterns, dictionary matching, Conditional Random Fields, Bi-LSTM sequence models, and transformer-based token classification. The ensemble applies a class-specific weighted voting strategy in which each model's per-entity influence is proportional to its per-class F1-score on a held-out validation set. The approach is assessed by using the ai4privacy/pii-masking-43k data set and the Mendeley Synthetic Financial Documents data set. EnsemblePII achieves a weighted F1 of 0.9749 on the general-purpose HF test set, marginally exceeding the strongest individual component (DistilBERT, F1 = 0.9744) on the in-distribution evaluation, and outperforming a published entity-span-level hybrid baseline by over 6 percentage points on the multi-class token-level task. On the Mendeley financial test set, Bi-LSTM and DistilBERT achieve F1 scores of 0.9959 and 0.9703, respectively, while the ensemble records 0.8433, revealing calibration sensitivity to entity classes absent from the weight calibration corpus. The results indicate that ensemble-based PII detection can improve in-distribution performance, but stable cross-domain generalization requires domain-aware calibration of ensemble weights; the DistilBERT component achieves the highest average F1 (0.9724) across both test sets, underscoring the value of transformer-based models for cross-domain PII detection.

Keywords*—Personally identifiable information; PII detection; weighted voting ensemble; named entity recognition; BIO tagging; transformer models; data privacy

I. Introduction

The expansion of online services has led to the collection, storage, and sharing of more personal data within and between organizations. There is a significant amount of sensitive information contained in emails, reports, contracts, logs, medical records, financial documents, and other unstructured text data. This presents major privacy and security concerns, particularly when Personally Identifiable Information (PII) is not adequately identified, secured, or deidentified. PII is any direct or indirect identifier that can be used to identify, contact, locate or distinguish an individual, such as names, addresses, phone numbers, email addresses, national identification numbers, financial data, and other sensitive attributes [1, 2]. The safeguarding of this information is a significant concern for organisations due to growing regulatory and legislative attention and the spread of data protection regulations around the world [3].

Identifying PII in unstructured text is difficult. Unstructured text holds sensitive information in different formats, informal language, inconsistent spelling and writing styles, as well as in different contexts. For instance, a word may be a name, an organisation, or a common word depending on the text in which it is used. Likewise, identifiers like phone numbers, emails, URLs, and account numbers may be present in full, truncated, obscured, or noisy representations. These factors make manual review a tedious task and justify the need for automated PII detection tools that are effective, scalable, and domain-general [1].

Rule-based methods, including regular expression matching and pattern-based heuristics, represent the earliest and most widely deployed approaches to structured PII detection [4]. These techniques are effective when target entities exhibit structured, syntactically predictable patterns, such as email addresses, phone numbers, or national identification numbers. They are also interpretable, efficient, and easy to deploy. However, rule-based methods are not effective in dealing with ambiguous entities, unstructured text, domain-specific variations, and context-specific PII [4]. These methods may be supplemented with dictionary or keyword-matching approaches that compare tokens with handcrafted lists of names, places, and organizations; however, their effectiveness depends on dictionary coverage and declines for unseen, uncommon, or misspelled entities.

Given the inherent limitations of symbolic approaches, machine learning and deep learning have gained popularity. Conditional Random Fields (CRF) offer a sequential approach to token labeling that accounts for label dependencies and includes contextual, lexical, and orthographic features [5]. Recurrent neural networks, particularly Long Short-Term Memory (LSTM) networks, can model these dependencies and are commonly used for modeling sequences in text [6]. Recently, transformer models like BERT [7] and its variants have performed well on NLP tasks, due to their ability to capture contextual features via self-attention [8]. These are especially important for detecting PII entities, which are sensitive to their context in the sentence [9].

Yet, single detection techniques still show inconsistent results per entity type and domain. Rule-based systems work well for identifiers, and not so well for names or organisations. Lexicon-based approaches may identify known entities, but not

new ones. CRF models work well for sequence-labeling applications, but require a lot of annotated data and hand-designed features. LSTM-based models are good for capturing sequential dependencies and tend not to be good for capturing long-range context. Transformer models have a strong understanding of context, but they are more computationally expensive and may still be influenced by dataset-specific tokenization effects [4, 6]. Therefore, it is not enough to trust one particular model to provide strong and generalizable PII detection results.

To address these challenges, this study proposes EnsemblePII, a weighted voting approach for robust PII detection in text. The framework combines five diverse detection components: a rule-based model, a dictionary-based model, a Conditional Random Field sequence tagger, a Bidirectional Long Short-Term Memory sequence model, and a transformer-based token classifier. Each component independently processes the input text and produces token-level predictions. These predictions are then merged using a weighted voting aggregator, where the influence of each model is determined by its validation performance for each PII category. This approach enables the framework to combine the strengths of symbolic, statistical, sequential, and contextual models while reducing the limitations of individual methods.

The main contributions of this study are threefold. First, it develops a class-specific weighted voting strategy for combining heterogeneous PII detection models at the token level. Second, it evaluates the proposed framework against its individual components using a unified BIO tagging scheme across seven canonical PII categories. Third, it investigates the effect of domain variation by testing the models on both a general-purpose PII dataset and a domain-specific financial dataset, highlighting practical challenges such as tokenization noise, entity imbalance, missing entity classes, and domain-specific weight calibration.

The rest of this study is structured as follows. Section II discusses previous research on rule-based, machine learning, deep learning, and hybrid methods for PII detection. The EnsemblePII framework and the voting mechanism are presented in Section III. Section IV presents details on datasets, the preprocessing pipeline, implementation, and evaluation metrics. Experimental results are presented and discussed in Section V. Section VI compares the proposed approach with other hybrid PII detection systems. The conclusion of the study is stated in Section VII. Future research directions are presented in Section VIII.

II. Literature Review

Prior research on PII detection in NLP can be grouped into rule-based, machine-learning, deep-learning, transformer-based, and hybrid approaches. Building on these approaches, several studies have explored machine learning and deep learning techniques to enhance the detection and protection of personally identifiable information.

Study [4] conducted a systematic comparative evaluation of machine learning approaches for automated PII detection in unstructured text, assessing rule-based methods, classical classifiers (including SVM), and deep learning models such as BERT. The research evaluated rule-based methods, classical classification techniques (including SVM), and deep learning models (such as Transformers/BERT). The analysis showed that deep learning models, such as BERT, have high accuracy and recall, surpassing the classical methods. However, the authors observed that these better-performing models require substantial computing power and a lot of data.

Study [9] developed an artificial intelligence (AI) model that uses natural language processing (NLP) approaches to identify sensitive personally identifiable information (PII) in unstructured text. The authors applied a pre-trained bidirectional encoder representation from the transformers (BERT) model, re-trained on a balanced dataset including both sensitive and non-sensitive texts, and used cross-validation to obtain accurate results. The results from experiments showed an accuracy of 99.558% and a 7.5% improvement over the previous model, indicating the success of combining NLP with deep learning models for the automatic detection of sensitive data for privacy protection. The research also suggested extending the experiments to more sensitive areas (such as healthcare and finance) and building scalable and privacy-focused versions of BERT for real-world applications.

Study [10] proposed a hybrid approach to detect and anonymize Personally Identifiable Information (PII) in financial records using rule-based Natural Language Processing (NLP) and machine learning. This research sought to protect PII such as name, credit card number, and email addresses from data breaches and ensure compliance with privacy regulations such as GDPR and CCPA. They created a varied synthetic dataset with the Faker library and performed detailed annotations to train a spaCy-based Named Entity Recognition (NER) model. The model proved to be effective with a precision of 94.7%, a recall of 89.4%, an F1-score of 91.1%, and an accuracy of 89.4% on synthetic data, and similar accuracy on real financial reports (an average of 93%). This solution addressed issues of semantic consistency, dealing with unstructured data, and speed. But there were some drawbacks with respect to ambiguous entities and biases in synthetic data, and improvements for multilingual and privacy-preserving capabilities can be made in future research. Their research provides a scalable, reliable, and compliant approach to safeguard sensitive financial data.

Study [11] proposed a rapid system for Personally Identifiable Information (PII) discovery with automatic consent discovery. The study highlighted the importance of protecting personal data in line with international standards like GDPR and PDPA. In the study, the authors introduced "PII RapidDiscover," which uses smart search algorithms and natural language processing, incorporating an enhanced Presidio library for English and Thai documents. It automatically detects PII and links with consent management processes. Through practical experiments, the system's effectiveness and efficiency were shown to be superior to current systems, revealing a holistic approach to improving data security, governance, and legal consent obligations.

Study [12] carried out a systematic evaluation study on hybrid approaches for multilingual Personally Identifiable Information (PII) detection, in which a new hybrid framework, RECAP, was developed. The research sought to solve the

problem of PII detection in low-resource languages, which are linguistically diverse and have little to no official data annotation. The framework combines deterministic regular expressions for static, rule-governed PII patterns with context-aware large language models for NER cases requiring semantic interpretation. The RECAP approach has a modular design that enables support for more than 300 entity types in a three-stage PII refinement process without the need for retraining the model for each language. The academic experiments indicate a substantial advantage for the RECAP approach, where it is 82% better than fine-tuned NER models and 17% better than Zero-shot LLMs in the Weighted F1-score. These results support the effectiveness of the hybrid approach in offering a Scalable and Adaptable framework for PII detection in 13 languages for privacy-compliant applications across languages.

Study [13] proposed a new model for cross-domain PII (Personally Identifiable Information) detection, de-identification, and re-identification using semantic features and ensemble machine learning. Their EESD-PII model combines class-balancing methods, including SMOTE, SMOTE-BL, and SMOTE-ENN, with ensemble classifiers such as Bagging, Boosting, AdaBoost, Random Forest, and Extra Trees, using Word2Vec CBOW representations for contextual text features. To tackle the problems of traditional dictionary-based approaches, as well as cross-domain generalization, the model is trained using multi-domain data from financial, medical, and identification records. The reported results indicate that EESD-PII outperformed existing methods, achieving an accuracy of 99.77% and maintaining performance on imbalanced and heterogeneous data sources. The work is significant in the field of privacy protection within modern data management systems. Table I summarizes the most relevant prior studies on PII detection by comparing their model types, datasets, and reported evaluation metrics.

TABLE I. SUMMARY OF RELATED WORKS

Ref	Year	Type of model				Dataset		Evaluation criteria		
		R-B	ML	DL	Hybrid	1 Dataset	2 Datasets	Best F1-score	Best Recall	Best Precision
[4]	2025	✓	✓	✓	✓	✓	✕	97.9%	97.2%	98.6%
[9]	2024	✕	✕	✓	✕	✓	✕	N/A	N/A	N/A
[10]	2025	✓	✓	✕	✓	✕	✓	91.1%	89.4%	94.7%
[11]	2023	✓	✓	✓	✓	✓	✕	97.39%	97.18%	97.63%
[12]	2025	✓	✕	✓	✓	✓	✕	0.657	0.605	0.729
[13]	2024	✕	✓	✕	✓	✓	✕	99.71%	99.63%,	99.81%
Our Proposed Model		✓	✓	✓	✓	✕	✓	97.49%	97.03%	97.95%

Many hybrid approaches for PII detection have been developed, but they continue to struggle with detecting different forms of PII, unstructured text, and contextual patterns. The majority of existing approaches combine only two methods, such as using rule-based detection combined with machine learning or NLP processing combined with a deep learning model. Although these methods outperform stand-alone systems, they still struggle to detect different types of PII in various text structures. Most current studies also use small data sets, synthetic data, or domain-specific data, making it hard to verify if their models are robust across different domains. Furthermore, some research only provides a subset of evaluation measures, making it harder to compare precision, recall, F1-score, and overall system performance.

To overcome these challenges, this study proposes a novel hybrid model that integrates rule-based patterns, dictionary matching, machine learning, deep learning, and transformer-based contextual classification models in an ensemble approach. This helps broaden the detection scope, as each model can contribute to the overall detection based on their strengths: rule-based approaches support structured identifiers, dictionary matching supports known entities, sequence models support contextual information, and transformer-based models support semantic information. Through a weighted voting approach, the proposed model seeks to enhance consistency and stability in predictions for different PII categories. The proposed model is also evaluated on two datasets to support its generalizability, making the framework more appropriate for detecting PII in various types of unstructured text.

III. ENSEMBLEPII FRAMEWORK

EnsemblePII is a weighted voting ensemble framework that combines five PII detection models working together on the same input text. Each model focuses on a different aspect of PII recognition, and their predictions are combined through a class-specific Weighted Voting Aggregator.

A. System Overview

This method proposes the development and evaluation of a comprehensive PII detection system that is built to balance performance with efficiency. EnsemblePII is a modular ensemble architecture that integrates five heterogeneous PII detection components under a unified per-class F1-weighted voting aggregation scheme. The framework integrates five complementary detection components: rule-based regex pattern matching, dictionary-based gazetteer matching, a Conditional Random Field (CRF) sequence tagger, a Bidirectional LSTM-CRF (Bi-LSTM-CRF) sequence model, and a DistilBERT transformer fine-tuned for token classification. The core of this method is the individual development and evaluation of five different PII detection models. Every model deals with the task from a different perspective, providing unique strengths and weaknesses of the models with respect to accuracy, contextual understanding, and computational cost.

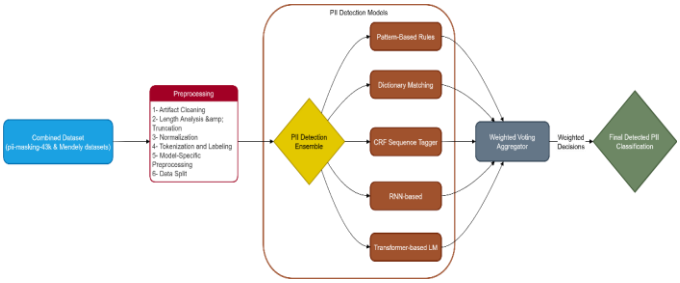

Fig. 1. EnsemblePII system architecture.

Fig. 1 shows the parallel input of the raw text to the five independent detection models (Pattern Rules, Dictionary Matching, CRF, RNN-based, and Transformer-based), each separately.

The results of the five models are combined by the "Weighted Voting Aggregator" that analyzes them with the help of category-specific weights for the final list of classified PII tokens.

B. Component Models

The EnsemblePII framework consists of five complementary detection components, combining rule-based, dictionary-based, statistical, sequential deep learning, and transformer-based approaches for PII detection [4].

1) Rule-based pattern matching (Regex): This component uses regular expressions (regex) to detect the most common PII patterns (such as phone numbers and social security numbers) [14]. The component uses a deterministic rule-based model to target structured PII that appears in consistent, well-defined formats. A large set of regex rules, which are designed to identify structured entities, is matched against the input text. Rules are divided into two main groups: general patterns apply to entities like IP addresses that follow common patterns in their representations; region-specific patterns apply to entities like national identification numbers, which need to be specially created patterns. The technique is able to detect high precision in a well-defined data type.

However, it tends to return false positives, and it does not work well with unstructured or irregular text [4]. Regex-based systems do not consider semantic context, thereby often dealing with poor semantic resolution and issues of surface variation. For example, a pattern that matches only numbers wouldn't correctly "red flag" a number that isn't sensitive.

2) Dictionary matching: This component uses gazetteers and lexical dictionaries of common names, locations, etc., to identify PII [4]. This technique is usually used in conjunction with others in hybrid PII detection systems [10]. Some PII detectors for network traffic, for instance, merge structured pattern matching techniques (e.g., regular expressions) and dictionary/lookup approaches. The technique works well on names of known entities, like first names, last names, and city names, added to the known list. However, its usefulness is highly dependent on the completeness of the underlying dictionaries it uses: it may not recognize entities not included in the dictionaries it uses (e.g., uncommon or misspelled names) and it cannot distinguish between entities that are not included in the dictionary that can be recognized as a person name from those that serve another grammatical role (e.g., common tokens such as Mark may be recognized as a person name in contexts where they do not play that role).

3) Conditional Random Field (CRF) sequence tagger: This component relies on a statistical sequence model that has been trained for PII span recognition based on the presence of contextual features in the surrounding text. It is specifically based on Conditional Random Field (CRF), a structured prediction framework, which is known to be a good baseline in sequence labelling problems [5]. In this setting, data anonymization is formulated as a sequence labeling task, where CRF models assign labels to tokens to identify and categorize personal data entities (such as names, addresses, or dates) and map them to anonymized tags [15]. To improve PII detection, the CRF component models dependencies between adjacent labels within the prediction pipeline. Sequence labeling is one of the tasks where the objective is to convert an input token sequence into an output label sequence of the same length. The CRF model is a stand-alone structured prediction model. It labels tokens by modelling the conditional probability of the whole label sequence conditional on the input token sequence. The CRF represents a classifier that takes into account dependencies between consecutive labels through the use of transition scores, such as B→I and O→I, with the transition B→I being favored over O→I. This is a global optimization over the label sequence to prevent structurally invalid sequences of labels (e.g., I-PER tag immediately after an O tag). This work adds a densely featured CRF model with token lexical features, character n-grams, token shapes, context window features, and gazetteer flags, which allows it to predict with informed decisions without the need for a neural architecture.

4) Bidirectional LSTM (Bi-LSTM) sequence tagger: This component uses a sophisticated recurrent neural network – the Bidirectional LSTM (Bi-LSTM) – to help it get more context. The Bi-LSTM captures both forward and backward information for each sequence by processing them in both directions. This bidirectional result in a more complete contextual representation than a vanilla unidirectional LSTM. The architecture is comprised of cells that can store, update, and reset their internal states as they go through the sequence. The BiLSTM model is merged with a CRF layer to create a hybrid Bi-LSTM-CRF architecture, which is used in various applications [16]. To enhance the Bi-LSTM model, this work adds a CRF layer to it, resulting in the Bi-LSTM-CRF model. In all tables in this study, this model is evaluated and reported as "Bi-LSTM", but it can be understood to be this Bi-LSTM-CRF architecture. The Bi-LSTM module takes in the sequence of tokens, processes them from both directions, and captures more complex context-dependent patterns [16]. Finally, the CRF layer is stacked on top of the Bi-LSTM output, which models the dependencies between adjacent labels and ensures

that only valid label transitions (B→I, I→O, etc.) are produced, giving globally consistent output sequences. It possesses both rich contextual representation and structured output constraint, and thus works well for sequence labeling tasks like PII detection. The hybrid model between two of the above core techniques (recurrent networks and CRF) is shown in Fig. 2.

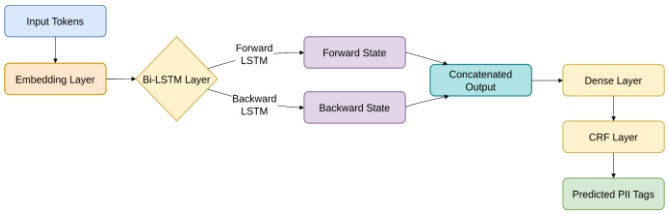

Fig. 2. Bi-LSTM-CRF architecture.

It illustrates the information that passes through the first embedding layer to the two bidirectional LSTMs layers that process the sequence from both directions to get the context, and to the last CRF layer that outputs the tagging decision, together with the learned tag-transition constraints (e.g., an I-NAME tag must follow a B-NAME tag).

5) Transformer-based token classification (DistilBERT): This component uses a transformer-based architecture fine-tuned for token classification to capture contextual representations relevant to PII detection. The design is based on modern NLP models, which are based on the transformer architecture. Transformers are a family of neural network architectures based on self-attention and have achieved strong performance in language representation and language understanding tasks [9]. Unlike models that process tokens strictly in sequence, transformers perform computation in a small, fixed number of steps by applying self-attention to model interactions between all words in a sentence, regardless of their relative positions [8]. The self-attention mechanism enables transformers to find long-distance dependencies, which is important for PII identification tasks in which the meaning of a token is derivable from its relations with other words in the text, rather than just its close neighbors or position.

In this work, the DistilBERT model [17] is fine-tuned for token classification, with a linear classification head to output the final classification of 15 BIO tag classes.

C. Weighted Voting Aggregator

For token Processing, the raw text is processed in parallel by all five models (Pattern Rules, Dictionary Matching, CRF, RNN-based, and Transformer-based). Each model identifies and classifies sensitive tokens, producing its own outputs [13]. Finally, for the Weighted Aggregation, the outputs are merged via the aggregator, which changes the impact of each model based on its evaluation results and category-specific performance. For example, if a phone number token is recognized by both the Regex model and the CRF, the Regex vote may receive greater influence for that class. In the case of a "Name" in an ambiguous context, a Transformer's vote would be given the greatest weight. The aggregator calculates the weighted votes for each token and outputs a final set of classified PII tokens.

It is important to note that per-class F1 weights are derived exclusively from the HF validation set. For entity classes absent from this set, specifically ORG and PHONE, no discriminative calibration signal is available, and all five models are assigned uniform weights for these classes. This constitutes a known methodological limitation: the ensemble weight calibration is domain-specific to the HF corpus, and its cross-domain applicability is not guaranteed.

In this work, the Mendeley validation set is held out exclusively for domain-transfer evaluation; using it during ensemble weight calibration would introduce data leakage into the cross-domain test, and consequently, the HF validation set alone is used as the calibration corpus.

A domain-adaptive weighting strategy, for example, calibrating weights on a held-out mixed-domain validation set or employing meta-learning over validation sets from multiple domains, represents a critical direction for future work.

IV. Materials and Methods

A. Datasets

Two publicly available synthetic datasets were used to train and evaluate the EnsemblePII framework [18, 19].

Dataset A - HF PII43k (ai4privacy/pii-masking-43k): 42,759 rows of text templates filled in various domains. This dataset was filtered, and 42,644 clean rows were kept, with 80% (34,118 samples) used for training, 10% (4,261 samples) for validation, and 10% (4,265 samples) for testing.

Dataset B - Mendeley Synthetic Financial Documents Dataset: 45,000 rows of synthetically generated financial documents rich with PII. From this dataset, 80% (36,004 samples) were used for training, 10% (4,496 samples) for validation, and 10% (4,500 samples) for testing.

TABLE II. Dataset Split Details

Split	Source Dataset	Size	Proportion	Purpose
HF Train	HF PII43k	34,118	80% of HF	Model training (combined)
HF Validation	HF PII43k	4,261	10% of HF	Per-class weight calibration for Ensemble
HF Test	HF PII43k	4,265	10% of HF	Primary held-out evaluation
Mendeley Train	Mendeley	36,004	80% of Mendeley	Model training (combined)
Mendeley Validation	Mendeley	4,496	10% of Mendeley	Internal validation
Mendeley Test	Mendeley	4,500	10% of Mendeley	Domain-transfer evaluation
Combined Training	**Both**	**70,122**	—	**Actual training corpus**

The combined training corpus consists of 70,122 samples (34,118 from HF + 36,004 from Mendeley), which are shuffled and provided together to all trainable models. The evaluation results on two disjoint test sets – HF test set (4265 samples) and Mendeley test set (4500 samples) are reported. Based on this setting, we can evaluate generalization in both domain-specific

and domain-agnostic settings. Table II presents the dataset split configuration used for training, validation, and testing across the HF PII43k and Mendeley financial datasets.

Fig. 3 shows the frequency of the most common PII tag types in the pii-masking-43k dataset. FULLNAME, EMAIL, and NAME are the most common labels, with CITY, URL, and NUMBER following, and the remaining labels, such as USERNAME, PASSWORD, and different types of ADDRESS and ACCOUNT labels, are relatively rare. This means that the data is highly person- and contact-centric, and in fact contains PII.

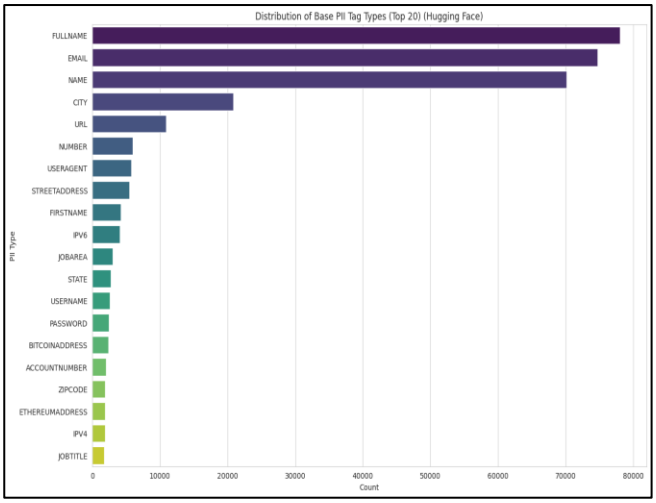

Fig. 3. PII tag distribution in HF dataset.

The frequency of each type of PII tag in the Mendeley data set is presented in Fig. 4. Name, company, and SSN are the most common types, showing that the most frequently occurring types are person or organisation identifiers. Other categories present in the Bar Plot include address, phone, credit card, URL, and email, with a variety of structured and unstructured PII tags. A more balanced distribution across PII types may improve the ensemble model's ability to generalize across sensitive information categories.

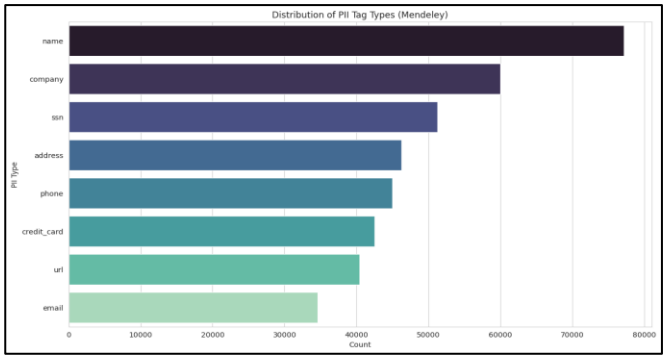

Fig. 4. PII tag distribution in the Mendeley dataset.

B. Unified Label Schema

Because the two source datasets use different entity naming conventions, a unified label schema was constructed to enable joint training. Seven canonical PII entity classes were defined. Table III shows the mapping process used to convert the original dataset-specific entity labels into the unified seven-class PII schema adopted in this study.

TABLE III. LABEL MAPPING SCHEMA

Unified Class	Maps From (HF Dataset)	Maps From (Mendeley)
PER	Name, Firstname, Lastname, Fullname, Displayname	Name, Person
LOC	City, Street address, State, Country, Zipcode, County, Secondary address	Address, Location
ORG	Company, Organization, Department	Company, Org
EMAIL	Email	Email
PHONE	Phone, Phonenumber, Tel	Phone
ID	SSN, Creditcardnumber, Passport, License, IBAN, Bitcoinaddress, Accountnumber, Pin, Vehiclevin, BBAN, Maskednumber, IPV4, IPV6, MAC	Ssn, Creditcard, ID
URL	URL, Domain, Useragent	URL

The entity tags were transformed into BIO format, resulting in a total of 15 labels: O, B-PER, I-PER, B-LOC, I-LOC, B-ORG, I-ORG, B-EMAIL, I-EMAIL, B-PHONE, I-PHONE, B-ID, I-ID, B-URL, I-URL [20].

C. Implementation Setup

For the implementation of the proposed model, a virtual machine environment was used on Google Colaboratory. The accelerators comprise NVIDIA L4 Tensor Core GPUs equipped with 24 GB of GDDR6 memory.

The host CPU back end has about 51 GB of RAM, 107 GB of disk space, and is running Ubuntu Linux (64-bit). For all model training and evaluations, Python 3 was used with Hugging Face Datasets, Scikit-learn, TensorFlow/PyTorch, and the Python re module for regex patterns.

Key libraries used: Hugging Face Datasets [21], Scikit-learn [22], TensorFlow/PyTorch [23], and the Python re module for regex patterns. Table IV summarizes the main hyperparameters and training settings used for each model in the proposed EnsemblePII framework.

TABLE IV. HYPERPARAMETER SUMMARY ACROSS ALL MODELS

Model	Key Hyperparameters	Training Epochs / Iterations
Regex	Deterministic (no training)	N/A
Dictionary	Gazetteer-based (no training)	N/A
CRF	Feature window ±2 tokens; L2 regularization; gazetteer features	Convergence-based (sklearn-crfsuite default)
Bi-LSTM	Hidden dim: 256×2 (bidirectional); Char-CNN embeddings; Word shape features; Adam optimizer; LR = 0.001; batch size = 32	12 epochs
DistilBERT	distilbert-base-uncased; max_length=512; warmup_steps; AdamW; batch size = 32	5 epochs (with early stopping)
Ensemble	Per-class F1 weights from HF Validation set; weighted majority vote per token	N/A (inference only)

To ensure reproducibility, all experiments were conducted with a fixed random seed of 42. Training times on the NVIDIA L4 GPU were approximately 4.2 hours for Bi-LSTM (12 epochs) and 1.8 hours for DistilBERT (5 epochs). Token-level alignment across heterogeneous models was achieved through a unified whitespace tokenisation pre-processing step applied before model-specific encoding, ensuring consistent token boundaries across all five component models.

The Bi-LSTM model includes CNN embeddings for characters and words, and features that focus on words' shape, providing orthographic information that helps identify structured PII like email addresses, identifiers, and phone numbers. The fine-tuned DistilBERT transformer is initialized from the distilbert-base-uncased checkpoint and is then fine-tuned for token classification using a linear classification head with 15 BIO tag classes.

D. Evaluation Metrics

For comparative purposes between various models, their performances are measured in terms of accuracy, precision, recall, and F1-score. These metrics are critical in the characterization of the effectiveness of these models in the detection of PII.

Accuracy: Accuracy represents the overall proportion of correctly classified tokens. However, in PII detection tasks, it can be deceptive, since non-PII tokens greatly outnumber PII tokens.

$$Accuracy = \frac{TN+TP}{TN+TP+FP+FN} \quad (1)$$

where, TP stands for True Positives, TN for True Negatives, FP for False Positives, and FN for False Negatives.

Precision: This is the proportion of the correctly classified tokens predicted as PII, which is the reliability of the positive predictions made by the model.

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

Recall: This is the proportion of actual PII tokens predicted by the model. In privacy-preserving applications, a high recall is especially crucial to ensure that most of the sensitive information is detected.

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

F1-Score: This is the harmonic mean of precision and recall, providing a single balanced measure that accounts for both error types.

$$F1\ socre = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (4)$$

All models are evaluated using the same set of evaluation metrics, calculated over the BIO-tagged labels. The main metric is the weighted F1-score, which addresses class imbalance by weighting each entity class's F1 by its frequency in the test set. Additionally, per-entity precision, recall, and F1 scores are reported for seven standard PII classes: PER (persons), LOC (locations), ORG (organizations), EMAIL, PHONE, ID (identifiers and structured codes), and URL. A prediction is correct only if both the entity boundary and type match the ground truth. This approach is strict, compared to token-level accuracy, and reflects the real-world need to locate PII correctly [24].

V. Experimental Results

A. Test Set Definitions

HF Test Set (4,265 samples): This is a subset from the pii-masking-43k dataset, it includes diverse, general-purpose PII found in conversational and template-based text across multiple fields.

Mendeley Test Set (4,500 samples): This test set is drawn from the Mendeley Synthetic Financial Documents dataset and provides domain-specific financial text, including audit reports, transaction confirmations, and invoices, with PII entities embedded in realistic financial data.

The two test sets enable evaluation of the system's performance across both general-purpose and financial-domain PII detection scenarios.

B. Baseline Model Evaluation

1) Rule-based regex model: The regex model is a pattern-matching baseline model identifying structured PII tokens with the help of a set of tailored regular expressions. No training data is required, and it is simply directly matched to the patterns. Email addresses are detected as top priority; email addresses have very strict syntactic rules followed by URLs, phone numbers, and identifiers. If the model is only based on patterns on the surface, it cannot model context, it cannot resolve ambiguity, not the named entity — a person, place or organization.

The lowest weighted F1 score across all evaluated models is recorded by the Regex model on the HF test set, at 0.1837. A major drawback is that if this model is applied to LOC entities (F1 = 0.000), locations don't have a consistent formatting for pattern-based recognition. The F1 score for the PER class is only 0.2899, which is moderate. This is facilitated by the ability of the regex model to recognize word sequences with capitalized words as candidate names. Similarly, URL has a low F1 score of 0.1048, with many URLs in the HF data set partially masked and thus not matching the regex.

Notably, the EMAIL class, a natural strength for a regex-based model, is also difficult for the model (F1 = 0.0188) on the HF test set. This is due to partially masked email addresses in the dataset (e.g., "##@example.com") that do not fit the standard regex patterns for complete addresses. This observation also accounts for the fact that the model based on the regular expression does much better on the Mendeley test set (weighted F1 = 0.4801) than on the other sets, where email addresses are presented in their full form (EMAIL F1 = 0.9320).

The difference between the results obtained on the HF and Mendeley sets is quite significant: 0.1837 vs 0.4801, which gives us an insight into the fact that the cleaner Mendeley text favors pattern-based models more than the noisier HF text. These results support the decision to include the Regex model as a component of the ensemble rather than as a standalone model [25].

2) Dictionary with regex model: The Dictionary+Regex model adds on entity gazetteers, or a list of known entities from the training data, to the regex baseline. Three gazetteers are built: PER (70,478 person-name tokens), LOC (42,735 location tokens), and ORG (38,111 organization tokens). For structured entities such as EMAIL, PHONE, URL, and ID, the model uses the same regex patterns as in Model 1. Named entities (PER, LOC, ORG) match each input token on the respective gazetteer.

Compared to the pure regex baseline, the Dictionary+Regex model achieves a slight improvement on the HF test set (F1: 0.2635 vs. 0.1837) and its main strength lies in the performance on LOC (F1 = 0.4340 vs. 0.0000). The LOC gazetteer, built from 42,735 training tokens, achieves high precision (0.7566) but moderate recall (0.3042), indicating that many locations in the test sets do not appear in the training data; this is expected in open-world named entity recognition.

These results show the limits of the symbolic methods. Both the Regex and Dictionary models serve as lower-bound baselines, ensuring the need for statistical and deep learning models. Fig. 5 compares the weighted F1-scores of the Regex and Dictionary+Regex baseline models on both test sets.

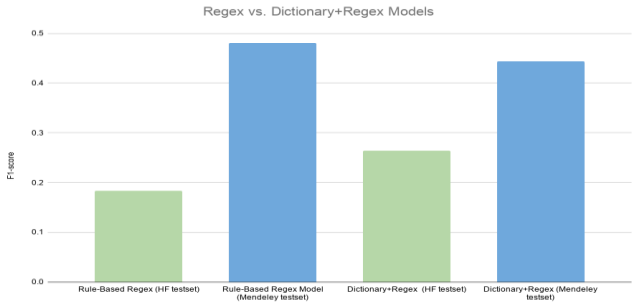

Fig. 5. Weighted F1-score comparison between the rule-based regex and dictionary+regex models on both test sets.

The HF gain (from 0.1837 to 0.2635) contrasts with the drop on Mendeley (from 0.4801 to 0.4432), highlighting the vocabulary-dependency of gazetteer methods.

3) CRF sequence tagger: The Conditional Random Field model shows the transition from symbolic to statistical sequence analysis. It is trained on 70,122 samples, using a different set of features. These features include token-level lexical features, character n-grams, token shapes, context window features, and gazetteer flags for all seven entity classes. The gazetteers for CRF feature includes 7,439 PER, 67,587 LOC, 17,420 ORG, 29,351 EMAIL, 38,156 PHONE, 75,912 ID, and 23,932 URL tokens, all compiled from the combined training data.

The CRF model shows a great improvement over both baselines, achieving an F1 of 0.9469 on the HF Test Set and 0.9639 on the Mendeley Test Set. This represents an approximately 0.68 F1 improvement over the Dictionary model on HF and highlights the effectiveness of statistical feature engineering for PII detection. Five out of seven entity classes achieve F1 scores above 0.84 on the HF Test Set, with EMAIL (0.9868), URL (0.9956), and PER (0.9496) standing out.

The two entity classes ORG and PHONE are completely missing from the HF dataset. As a result, there are no examples of these classes in both the HF Test Set and the HF Validation Set, and all models give an F1 of 0.000 for these classes on HF. This is not a model constraint, but rather a characteristic of the data: with those same models, PHONE is detected with F1 = 0.9463 on the Mendeley Test Set.

However, on the Mendeley Test Set, PHONE achieves F1 = 0.9463, confirming that the issue relates to data quality rather than a model limitation [15].

4) Bi-LSTM sequence tagger: The Bi-LSTM model performs sequence labeling using three embedding channels: word embeddings with a vocabulary size of 265,960, character-level CNN embeddings with 94 unique characters, and word-shape embeddings representing 194 unique shape patterns [26]. These are combined and passed through a two-layer bidirectional LSTM network with a hidden size of 256 per direction, outputting 512-dimensional representations for each token. The model has 35,682,085 trainable parameters, and it is trained for 12 epochs using the Adam optimizer with a learning rate of 0.001 and a batch size of 32 [27]. Class weights are applied during training, too, to deal with the imbalance between the 'O' (non-PII) class and the rare PII entity classes. Fig. 6 shows the Bi-LSTM training loss over 12 epochs, demonstrating the convergence pattern of the model during training.

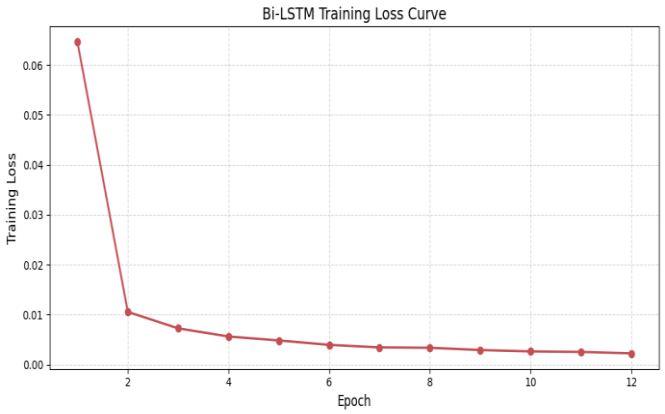

Fig. 6. Bi-LSTM training loss curve over 12 epochs.

Training loss decreases substantially within the first two epochs and continues to decline monotonically thereafter, indicating stable convergence without oscillation or divergence.

The Bi-LSTM model's results reveal an obvious performance discrepancy. On the Mendeley Test Set, it achieves a near-perfect weighted F1 of 0.9959, the highest score of any single model in this study. Four entity classes (ID, PHONE, LOC, URL) achieve near-perfect F1, and even the complex EMAIL class achieves F1 = 0.9991. This strong performance on Mendeley shows the model's ability to learn the consistent patterns of the synthetic financial text, which follows clear template structures.

However, on the HF Test Set, the Bi-LSTM records an F1 of only 0.8433, which is lower than the CRF's 0.9469. This gap is

due to two reasons. First, the HF dataset's tokenization masking again makes it harder to detect EMAIL and URL (F1 = 0.5968 and 0.5590, respectively) because partially masked tokens damage the character patterns the Bi-LSTM needs.

The contrast between 0.8433 (HF) and 0.9959 (Mendeley) shows that the Bi-LSTM is highly sensitive to domain changes. It performs well in well-structured domains but struggles with tokenization noise [28].

5) Baseline summary: Table V compares the weighted F1-scores of the baseline models on the HF and Mendeley test sets, highlighting the relative strengths and weaknesses of each approach.

TABLE V. BASELINE MODEL SUMMARY: F1-SCORE

Model	HF Test F1	Mendeley Test F1	Avg. F1	Key Strength	Key Weakness
Regex	0.1837	0.4801	0.3319	High precision on structured entities (Mendeley)	Cannot detect named entities; breaks on masked tokens
Dictionary+Regex	0.2635	0.4432	0.3534	Adds LOC detection via gazetteer	LOC gazetteer fails on domain shift; ORG/PHONE zero
CRF	0.9469	0.9639	0.9554	Strong cross-domain consistency	ORG/PHONE fails on HF due to masking
Bi-LSTM	0.8433	0.9959	0.9196	Near-perfect on Mendeley	Large performance gap under domain shift

The CRF model stands out as the most robust baseline with an average F1 of 0.9554, achieving strong results across both test sets. The Bi-LSTM reaches the highest single-dataset score on Mendeley (0.9959) but shows weaker performance under domain shift. Both regex and dictionary matching models highlight the limitations of pattern-based approaches, especially in the presence of tokenization noise (in the HF dataset). Their results justify their position as lower-weight contributors in the ensemble system.

C. Transformer Component Evaluation (DistilBERT)

Before evaluating the full ensemble, we discuss the fifth and most powerful individual model, the fine-tuned DistilBERT model. DistilBERT is fine-tuned for token classification using all 70,122 combined training samples [29]. The fine-tuning process was conducted for 5 epochs, with early stopping based on validation F1. The training history is shown in Table VI. Fig. 7 presents the DistilBERT fine-tuning curves, showing changes in training loss and validation F1-score across the five training epochs. Table VI reports the DistilBERT fine-tuning history across five epochs, including training loss, validation loss, and validation F1-score.

TABLE VI. DISTILBERT FINE-TUNING HISTORY 5 EPOCHS

Epoch	Training Loss	Validation Loss	Validation F1
1	0.103743	0.014234	0.9673
2	0.002088	0.010111	0.9757
3	0.001085	0.010821	0.9769
4	0.000555	0.008429	0.9808
5	0.000284	0.007503	0.9804

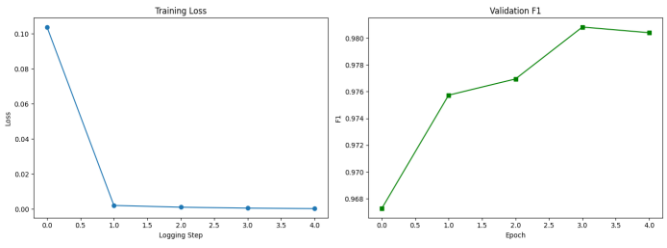

Fig. 7. DistilBERT fine-tuning curves: training loss (left) and validation F1 (right).

The training loss drops sharply within the first epoch. The validation F1 rises from 0.967 to a peak of 0.9808 at epoch 4, showing stable convergence over 5 epochs of fine-tuning.

The DistilBERT Transformer achieves the second-highest F1 on the HF Test Set (0.9744). It scores perfectly on EMAIL and URL, and performs well for PER (0.9781) and ID (0.9632). On the Mendeley Test Set, it reaches a strong F1 of 0.9703, detecting all seven entity classes with no zero scores. Most classes achieve precision close to 1.000, meaning that the model's positive predictions are highly reliable, even though it has lower recall for LOC (0.9224), PER (0.9388), and URL (0.9213).

The main takeaway is that the Transformer's strength lies in its accurate contextual understanding. It almost does not produce false positives, but it may miss some true entities, resulting in lower recall scores. This focus on precision balances the CRF's recall-oriented approach.

D. EnsemblePII System Results

The EnsemblePII system combines predictions from all five models using a dynamic Weighted Voting Aggregator. All five models predict BIO labels for each token in the input text, in the style of voting. Each vote is given a score according to the F1 per entity score of the model on the HF validation set (4261 samples). For example, a transformer has a vote weight of 0.9838, whereas a regex model has a vote weight of 0.3028. For instance, in the case of a PER token, the weight of the vote cast by the transformer is 0.9838, but the weight of the vote cast by the regex model is only 0.3028. In the case of EMAIL, the votes of the transformer and CRF have the greatest impact on the final vote. Table VII presents the per-class validation F1 weights used by the weighted voting aggregator to combine predictions from the five component models.

Table VII displays some of the important properties of the system: For EMAIL and URL, the Transformer and CRF are almost the same in terms of dominant weights. The weight of the Transformer is 0.9838 for PER, followed by CRF (0.9573) and Bi-LSTM (0.9035). The weights for the Regex model and the Dictionary model are less than 0.50 per entity class, implying that both models are auxiliary and not the main decision makers. Note that there is no ORG or PHONE in Table VII. These two

classes are not present in the HF data set, and thus, all models have zero (or near-zero) F1 for the class ORG and PHONE on the HF validation set, and meaningful per-class weights cannot be obtained from the HF validation set for these classes. As a result, for those classes, the ORG and PHONE votes cast by all models are all given the same weight (all models have uniform weight = 1.0), and the fallback scheme is to use majority voting. Table VIII reports the per-entity precision, recall, and F1-score achieved by EnsemblePII on the HF test set.

TABLE VII. PER-CLASS F1 WEIGHT MATRIX ON HUGGING FACE VALIDATION SET

Model	PER	LOC	EMAIL	ID	URL	Overall
Regex	0.3028	0.0000	0.0183	0.0736	0.1272	0.1911
Dictionary	0.2815	0.4577	0.0219	0.0864	0.1379	0.2762
CRF	0.9573	0.9210	0.9811	0.8014	1.0000	0.9501
Bi-LSTM	0.9035	0.8519	0.6325	0.8522	0.6044	0.8491
Transformer	0.9838	0.9635	1.0000	0.9314	1.0000	0.9808

TABLE VIII. ENSEMBLEPII PER-ENTITY RESULTS ON HF TEST SET

Entity	Precision	Recall	F1-Score
EMAIL	1.0000	1.0000	1.0000
ID	0.9357	0.9850	0.9597
LOC	0.9571	0.9470	0.9520
PER	0.9837	0.9703	0.9769
URL	1.0000	1.0000	1.0000
F1-Score	—	—	0.9749

Table IX reports the per-entity precision, recall, and F1-score achieved by EnsemblePII on the Mendeley financial test set.

TABLE IX. ENSEMBLEPII PER-ENTITY RESULTS ON MENDELEY TEST SET

Entity	Precision	Recall	F1-Score
EMAIL	0.9962	0.9925	0.9943
ID	0.9908	1.0000	0.9954
LOC	0.9804	0.9998	0.9900
ORG	0.9368	0.0823	0.1513
PER	0.8214	0.9925	0.8989
PHONE	0.9703	0.9190	0.9439
URL	0.9842	1.0000	0.9920
F1-Score	—	—	0.8433

While EnsemblePII achieves the highest weighted F1 on the HF test set (0.9749), its average F1 across both test sets (0.9091) is lower than that of the DistilBERT Transformer (0.9724), CRF (0.9554), and Bi-LSTM (0.9196). This finding demonstrates that ensemble-based integration improves in-distribution performance but is subject to calibration-induced degradation in cross-domain scenarios, specifically when entity classes present in the target domain (ORG, PHONE) are absent from the calibration corpus.

The ensemble achieves a perfect F1 on EMAIL and URL, matching the Transformer. For the ID class, the ensemble achieved an F1-score of 0.9597. The ensemble's lower ID F1 score (0.9597 vs. the CRF's 0.8014 individual score) reflects the diluting effect of non-zero voting weights assigned to the Regex and Dictionary models for this class (0.0736 and 0.0864, respectively; Table VII), which introduce systematic noise from lower-performing components.

However, on the Mendeley Test Set, the EnsemblePII records a weighted F1 of only 0.8433. This is well below the Bi-LSTM's 0.9959 and the Transformer's 0.9703. The drop is mainly due to the ORG class, which falls to an F1 of 0.1513 on Mendeley. The low ORG F1 on Mendeley (0.1513) is not due to the lack of precision, but rather extremely low recall (0.0823). The ensemble precision for ORG is 0.9368, meaning that approximately 93.68% of predicted ORG tokens are correct. However, since ORG calibration weights are also calculated on the HF validation set, which does not include ORG, all five component models contribute equally (and non-zero) in the ORG predictions. The models that naturally make ORG predictions, via their architecture (probably the Transformer model and the Dictionary model), sometimes predict ORG for ORG tokens, while the remaining models suppress ORG predictions, so the ensemble recall is very low. The number of rare predictions that do pass through is very accurate, which is why the precision is high, and the recall is low, and hence the low F1. ORG and PHONE are completely missing from the HF dataset (validation and test splits), and so no validation weights could be computed for these classes with HF. The ensemble consequently resorts to voting for ORG and PHONE with equal weights when tested on Mendeley.

E. Aggregate Performance Comparison

Table X provides an aggregate comparison of all evaluated models on both test sets, including their rankings and average F1-scores.

TABLE X. MASTER PERFORMANCE COMPARISON: ALL MODELS ON BOTH TEST SETS

Rank (HF)	Model	HF Test F1	Rank (Mendeley)	Mendeley Test F1	Avg. F1
1	EnsemblePII	0.9749	4	0.8433	0.9091
2	Transformer (DistilBERT)	0.9744	2	0.9703	0.9724
3	CRF	0.9469	3	0.9639	0.9554
4	Bi-LSTM	0.8433	1	0.9959	0.9196
5	Dictionary+Regex	0.2635	6	0.4432	0.3534
6	Regex	0.1837	5	0.4801	0.3319

As shown in the aggregate performance matrix, no single model performs best on both test sets. EnsemblePII achieves the highest score on the HF Test Set, with an F1-score of 0.9749, but its performance drops to 0.8433 on the Mendeley Test Set. In contrast, the Bi-LSTM ranks fourth on the HF Test Set with an F1-score of 0.8433, but achieves the best performance on the Mendeley Test Set with an F1-score of 0.9959. The DistilBERT Transformer is the most stable model across both datasets, achieving an average F1-score of 0.9724 and ranking second on both test sets.

When comparing the models by average F1-score across the two test sets, the Transformer is the most consistently reliable model, followed by the CRF, Bi-LSTM, and EnsemblePII. The lower average score of EnsemblePII, 0.9091, compared with 0.9724 for the Transformer, is mainly caused by its weak ORG detection on the Mendeley Test Set. This suggests that the ensemble's reduced cross-domain performance is primarily due to domain-specific weight calibration rather than the ensemble architecture itself. Fig. 8 compares the weighted F1-scores of all evaluated models on the HF and Mendeley test sets.

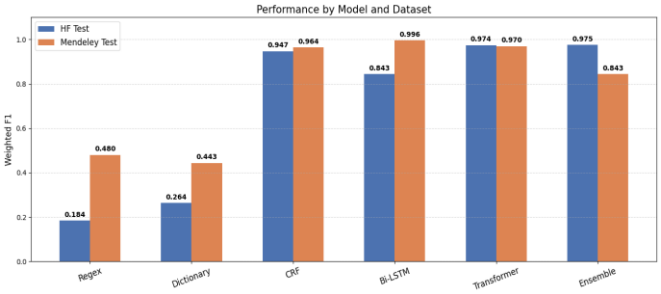

Fig. 8. Weighted F1 bar chart: All models vs both test sets.

Fig. 9 presents the per-entity F1-score heatmap for all models on the HF test set, showing how performance varies across different PII classes.

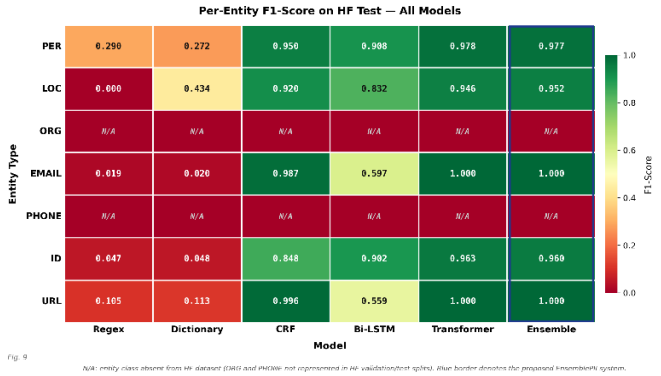

Fig. 9. Per-entity F1 heatmap on HF test set.

Fig. 10 presents the per-entity F1-score heatmap for all models on the Mendeley test set, highlighting model-specific strengths and weaknesses across financial PII classes.

The heatmap places the seven entity classes in rows and the six systems in columns; each cell encodes the F1-score for the corresponding entity-system pair. This visualization offers a quick summary of how each model performed for each class. Transformers and Ensembles show the darkest cells for the EMAIL, URL, PER and ID. However, the ORG and PHONE rows for all models on the HF set are uniformly marked as absent (N/A) in red, reflecting their complete exclusion from the HF dataset rather than a model failure. This CRF column is flatter than the Bi-LSTM column—the CRF shows more uniformity of performance across the entities on the HF data. Contrasting this behavior to the Mendeley heatmap, it is seen that the CRF, Bi-LSTM and Transformer models obtain consistently strong scores. In contrast, EnsemblePII performs well for most classes but shows reduced ORG and PER performance because weaker component models influence the aggregated output for these classes.

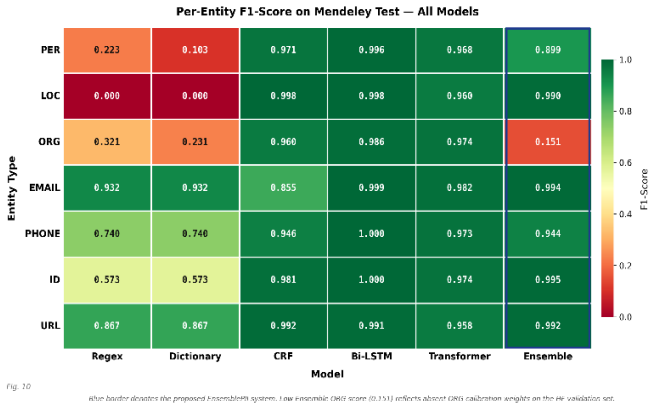

Fig. 10. Per-entity F1 heatmap on the Mendeley test set.

F. Precision–Recall Trade-off Analysis

An important part of evaluating PII detection is the precision-recall trade-off. In privacy protection tasks, recall is usually the more important measure. A missed PII entity represents a false negative and may create a privacy gap, whereas over-prediction produces additional false positives and unnecessary masking. Table XI shows the precision and recall for the three best models on each test set compared to the EnsemblePII framework [30].

TABLE XI. OVERALL PRECISION AND RECALL FOR TOP-3 MODELS

Model	HF Test Precision	HF Test Recall	Mendeley Precision	Mendeley Recall
CRF	0.9543	0.9398	0.9660	0.9624
Bi-LSTM	0.8370	0.8499	0.9948	0.9970
Transformer	0.9776	0.9714	0.9994	0.9424
EnsemblePII	0.9796	0.9703	0.9515	0.8513

The DistilBERT Transformer shows the highest precision on the Mendeley dataset and the highest precision among individual models on HF, while EnsamblePII has the highest HF precision overall, which means that its positive predictions are reliable. On Mendeley, almost every entity that the Transformer identifies is correct (macro precision ≈ 0.9994), even though it misses around 5.8% of true PII entities, resulting in a recall score of 0.9424. The CRF model achieves a balanced precision-recall profile on HF Test, with a difference of only 0.0145 between precision (0.9543) and recall (0.9398). The Bi-LSTM achieves a balanced performance on Mendeley (precision 0.9948, recall 0.9970), showing a good balanced for that domain.

The EnsemblePII shows high precision on HF (0.9796) but a lower score on Mendeley (0.9515). it shows a degraded recall on Mendeley (0.8513), again driven by some baseline collapse for complex entity classes. For applications where recall is critical, the Bi-LSTM on Mendeley-type data or the Transformer on HF-type data would be preferred over the full ensemble.

G. Discussion

The progressive performance improvements from symbolic baselines to the full ensemble system can be attributed to three cumulative factors: 1) feature richness, 2) contextual depth, and 3) ensemble complementarity, each of which is examined in detail below.

Feature richness is most clear in the improvement from the Dictionary model (F1 = 0.2635 on HF) to the CRF (F1 = 0.9469). The CRF's feature engineering, which includes character n-grams, token shapes, and all seven entity gazetteers, gives the statistical model enough information for the final classification signal to reach near state-of-the-art performance without using deep learning. This suggests that, for sequence labeling on well-structured PII data, carefully designed features can remain competitive with neural methods.

Contextual depth explains the Transformer's improvement over the CRF in most individual entity classes. The DistilBERT's self-attention mechanism processes all tokens together, allowing it to determine entity boundaries that depend on sentence-level context. For example, it can tell the difference between a person's name and an organization's name in "John Smith from Smith and Associates." The Bi-LSTM provides intermediate contextual depth. It is stronger than the CRF's fixed context window, but not as expressive globally as self-attention.

The independence of errors by different models (ensemble complementarity) explains the slight improvement of the ensemble over the Transformer on the HF Test. The performance of the CRF in terms of LOC recall (0.9111) is close to that of the Transformer (0.9462), and the ensemble's LOC recall (0.9470) exceeds that of either model.

The effects of the combination of both datasets for training were clear. The CRF model is evaluated on Mendeley with a score of 0.9639, which is far above what is expected from a model trained in this way, given the masking problems with HF data. The Bi-LSTM obtains an outstanding score of 0.9959 on Mendeley, demonstrating its capability of comprehending the structure of synthetic financial text. We hypothesise that DistilBERT's cross-domain stability arises from simultaneous exposure to both HF and Mendeley training distributions, enabling the model to learn domain-invariant contextual representations — a hypothesis that can be tested by training DistilBERT exclusively on HF data and evaluating on Mendeley. The DistilBERT is quite good on both datasets (HF: 0.9744, Mendeley: 0.9703), implying that it learned both distribution styles simultaneously.

An implicit component-contribution analysis is available through the per-class weight matrix in Table VII. The Regex and Dictionary models receive overall validation weights of 0.1911 and 0.2762, respectively, substantially lower than the CRF (0.9501), Bi-LSTM (0.8491), and Transformer (0.9808). These near-zero weights indicate that the two symbolic models exert minimal influence on token-level decisions for most entity classes; the ensemble effectively reduces to a three-model weighted vote among CRF, Bi-LSTM, and Transformer for the majority of predictions. The main contribution of the symbolic components is a precision-oriented signal for highly structured PII patterns such as EMAIL and URL, where deterministic regex and gazetteer matching provide reliable but lower-recall coverage that complements the learning-based models.

However, combined training introduces limitations during the ensemble weight-calibration step. Since ORG and PHONE are completely absent from the HF dataset, there are no validation F1 scores for these classes in the HF validation dataset, and meaningful per-class calibration weights can not be derived for them. This means in this ensemble, there is no calibrated guidance to detect ORG and PHONE on Mendeley, which is a good representation of both classes. This leaves the ensemble unaware of these entity classes, even in the case of evaluating on Mendeley, where they are well defined. This is a problem that could be solved with a domain-aware weighting strategy [31].

VI. Comparison with State-of-the-Art Hybrid Systems

In order to put the performance of EnsemblePII in perspective, the performance of two existing hybrid PII detection systems from the literature is compared to that of EnsemblePII.

Study [9] reported strong results for a BERT-based hybrid model, with 99.558% accuracy and a 99.559% F1-score. Their system, however, is assessed as a binary document-level classification system, in which a classification is performed based on the presence or absence of PII. EnsemblePII tackles a more specific task, the multi-class token-level sequence labelling with BIO tags.

Hence, the results are not directly comparable, but it demonstrates the effectiveness of transformer-based approaches for PII detection.

EnsemblePII attains an F1 of 97.49% on the HF test set, compared to the 91.1% F1 reported by the study [10]; however, direct numerical comparison must be treated with caution because study [10] operates at the entity-span detection level, whereas EnsemblePII addresses the more granular multi-class token-level BIO tagging task, a fundamentally different evaluation granularity. Within this constraint, EnsemblePII also demonstrates a more balanced precision–recall profile (Precision: 97.95%, Recall: 97.03%) than [10] (Precision: 94.7%, Recall: 89.4%). It does not, however, perform as well on a Mendeley financial dataset, where most problems stem from a domain mismatch in ensemble weight calibration, impacting the most important entity classes ORG and PHONE. Table XII compares EnsemblePII with selected hybrid PII detection systems from the literature, while noting differences in architecture, evaluation dataset, and task level.

TABLE XII. COMPARATIVE ANALYSIS: ENSEMBLEPII VS SOTA HYBRID SYSTEMS

System	Architecture Type	Precision	Recall	F1-Score	Evaluation Dataset	Task Level
[9]	NLP preprocessing + fine-tuned BERT	99.564%	99.558%	99.559%	pii-masking-200k (binary)	Document-level binary
[10]	Rule-based NLP + spaCy NER	94.7%	89.4%	91.1%	Synthetic financial (synthetic)	Entity-span level
EnsemblePII (HF Test)	**Weighted voting: Regex + Dict + CRF + Bi-LSTM + DistilBERT**	**97.95%**	**97.03%**	**97.49%**	**HF PII43k (multi-class token)**	**Token-level multi-class**
EnsemblePII (Mendeley)	**Weighted voting ensemble**	**95.15%**	**85.13%**	**84.33%**	**Mendeley financial (multi-class token)**	**Token-level multi-class**

Fig. 11 compares EnsemblePII with selected state-of-the-art hybrid systems using precision, recall, and F1-score values reported in the manuscript.

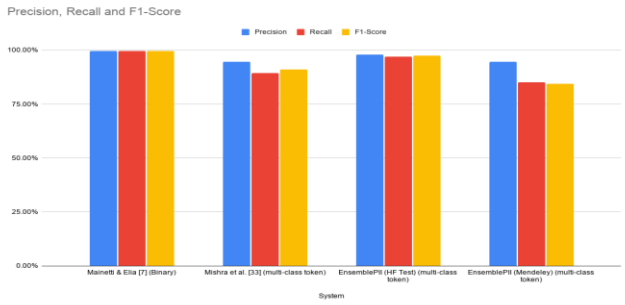

Fig. 11. EnsemblePII vs SOTA: Precision/Recall/F1 Bar.

EnsemblePII outperforms the system in [10] (91.1%) on a more challenging multi-class token-level task on the HF test set with an F1 score of 97.49%. On the Mendeley financial test set, EnsemblePII achieves an F1-score of 84.33%; this lower result is primarily attributable to domain-specific weight-calibration failures for ORG and PHONE entities. Concerning the Mainetti and Elia system: since it is a binary document level task (PII present/not present), it is not appropriate to make a direct numerical comparison. Finally, it should be emphasized that EnsemblePII tackles a much more difficult problem: multi-class token-level sequence labelling with 15 BIO tags, and that it yields strong performance on this more difficult task. First, concerning the complexity of the tasks: EnsemblePII is a multi-class task, token-level classification with 15 BIO tags and 7 entity types, whereas binary detection or single-class entity extraction is simpler. The score of 97.49% F1 on the test set (HF) shows the robustness of the ensemble on general-purpose PII text compared to the 91.1% F1 obtained by the study [10] on a similar entity span task. Second, when it comes to robustness, the two comparison systems do not report on cross-domain generalization. The only system included in the comparison that has been tested on two different types of test sets is EnsemblePII, which used both domain-specific financial documents and general-purpose PII text. This provides more evidence of its real-world applicability. Third, it comes to Precision-Recall Balance, EnsemblePII on HF Test gives better balance (Precision: 97.95%, Recall: 97.03%) than the study [10] (Precision: 94.7%, Recall: 89.4%). This results in EnsemblePII capturing the sensitive entities with fewer false alarms. This twofold benefit is particularly important for applications that have privacy requirements.

VII. CONCLUSION

In this study, we introduced EnsemblePII, a weighted voting ensemble framework for multi-class PII detection in unstructured text. The framework combines five complementary detection approaches: Regex pattern matching, Dictionary+Regex matching, CRF sequence tagging, Bi-LSTM sequence modelling, and DistilBERT-based token classification. The models were evaluated on two test sets representing different domains: the HF PII43k dataset and the Mendeley Synthetic Financial Documents dataset. The results show that symbolic baselines alone are insufficient for reliable PII detection, especially in noisy or masked text. Regex and Dictionary+Regex achieved low weighted F1-scores on the HF test set, while the CRF, Bi-LSTM, and Transformer models provided much stronger performance through contextual and sequence-based learning.

The experimental results indicate that EnsemblePII achieved the best performance on the HF test set, with a weighted F1-score of 0.9749, slightly outperforming the DistilBERT Transformer model, which achieved 0.9744. However, the ensemble did not achieve the best cross-domain performance. On the Mendeley financial test set, its weighted F1-score dropped to 0.8433, mainly because the ORG and PHONE classes were not represented in the HF validation set used for class-specific weight calibration. In contrast, the DistilBERT Transformer was the most stable individual model across both datasets, achieving 0.9744 on HF and 0.9703 on Mendeley, while the Bi-LSTM achieved the highest single-dataset score with 0.9959 on the Mendeley test set. These findings show that ensemble-based PII detection can improve in-distribution performance, but cross-domain effectiveness is contingent on the quality and domain coverage of the calibration data. Across both test sets, the DistilBERT Transformer achieves the highest average F1-score (0.9724) among all evaluated models, making it the most consistently reliable choice when cross-domain generalization is a deployment priority.

VIII. FUTURE WORK

Several directions for future work emerge from this study. First, domain-adaptive weight calibration, for instance, through meta-learning over multiple validation domains or multi-domain validation sets, should be investigated to address the ORG/PHONE calibration failure identified on the Mendeley test set. Second, the framework should be extended to real-world, non-synthetic corpora, such as medical records and legal documents, to assess performance under natural language

variability and annotation noise. Third, multilingual PII detection represents an important extension, as PII formats are language- and culture-specific, and the current framework is English-only. Fourth, the effect of differential weight-thresholding, excluding models below a class-specific F1 threshold from the ensemble vote, warrants systematic ablation. Fifth, privacy-preserving ensemble training, such as federated learning with differential privacy, is an important direction given the sensitive nature of the training data.

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to the Deanship of Graduate Studies and Scientific Research, Taif University, for its support and for funding this work. The authors also thank their colleagues and reviewers for their valuable guidance and constructive feedback throughout this research.

REFERENCES

[1] I. Makhdoom, M. Abolhasan, J. Lipman, N. Shariati, D. Franklin, and M. Piccardi, "Securing personally identifiable information: a survey of SOTA techniques, and a way forward," IEEE Access, vol. 12, pp. 116740-116770, 2024.

[2] V. Garg, "Safeguarding sensitive information: a comprehensive approach to PII anonymization and data masking," International Journal for Multidisciplinary Research, vol. 4, no. 6, art. 21490, 2022.

[3] M. Abreu de Magalhães, "Data protection regulation: A comparative law approach," International Journal of Digital Law, vol. 2, no. 2, pp. 33-53, 2021.

[4] A. Makhambet and A. Moldagulova, "A comparative analysis of machine learning methods for personal information recognition (PII) in unstructured texts," Computing & Engineering, vol. 3, no. 1, pp. 41-52, 2025.

[5] T. Burr and A. Skurikhin, "Conditional random fields for pattern recognition applied to structured data," Algorithms, vol. 8, no. 3, pp. 466-483, 2015.

[6] F. Mortezapour Shiri, T. Perumal, N. Mustapha, and R. Mohamed, "A comprehensive overview and comparative analysis on deep learning models," Journal on Artificial Intelligence, vol. 6, pp. 301-360, 2024.

[7] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in Proc. Conf. North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 2019, pp. 4171-4186.

[8] A. Rahali and M. A. Akhloufi, "End-to-end transformer-based models in textual-based NLP," AI, vol. 4, no. 1, pp. 54-110, 2023.

[9] L. Mainetti and A. Elia, "Detecting personally identifiable information through natural language processing: a step forward," Applied System Innovation, vol. 8, no. 2, p. 55, 2025.

[10] K. Mishra, H. Pagare, and K. Sharma, "A hybrid rule-based NLP and machine learning approach for PII detection and anonymization in financial documents," Scientific Reports, vol. 15, art. no. 22729, 2025.

[11] S. Fugkeaw and P. Sanchol, "Enabling efficient personally identifiable information detection with automatic consent discovery," ECTI Transactions on Computer and Information Technology, vol. 17, no. 2, pp. 245-254, 2023.

[12] H. Rajgarhia, S. Gupta, A. Shaik, G. Praveen Kumar, Y. Santhoshraj, S. N. T. Nishitha, and A. Mukherji, "An evaluation study of hybrid methods for multilingual PII detection," arXiv:2510.07551 [cs.CL], Oct. 2025.

[13] P. Kulkarni, C. N. K., and H. R., "An advanced semantic feature-based cross-domain PII detection, de-identification, and re-identification model using ensemble learning," International Journal of Advanced Computer Science and Applications, vol. 15, no. 12, pp. 763-779, 2024.

[14] C. C. Chiu, C. S. Yang, and C. K. Shieh, "A method to improve the accuracy of personal information detection," Journal of Computer and Communications, vol. 11, no. 6, pp. 131-141, 2023.

[15] J. L. Leevy, T. M. Khoshgoftaar, and F. Villanustre, "Survey on RNN and CRF models for de-identification of medical free text," Journal of Big Data, vol. 7, art. no. 73, 2020.

[16] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural architectures for named entity recognition," in Proc. 2016 Conf. North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego, CA, USA, 2016, pp. 260-270.

[17] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," arXiv:1910.01108 [cs.CL], Oct. 2019.

[18] ai4Privacy, "pii-masking-43k," Hugging Face, 2023. [Online]. Available: https://huggingface.co/datasets/ai4privacy/pii-masking-43k

[19] K. Mishra, H. Pagare, R. Bidwe, and S. Mishra, "Synthetic dataset for PII detection and anonymization in financial documents," Mendeley Data, vol. V1, 2024.

[20] E. F. Tjong Kim Sang and F. De Meulder, "Introduction to the CoNLL-2003 shared task: language-independent named entity recognition," in Proc. Seventh Conf. Natural Language Learning at HLT-NAACL 2003, 2003, pp. 142-147.

[21] Q. Lhoest et al., "Datasets: a community library for natural language processing," in Proc. Conf. Empirical Methods in Natural Language Processing: System Demonstrations, Punta Cana, Dominican Republic, 2021, pp. 175-184.

[22] F. Pedregosa et al., "Scikit-learn: machine learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825-2830, 2011.

[23] M. Abadi et al., "TensorFlow: a system for large-scale machine learning," in Proc. 12th USENIX Conf. Operating Systems Design and Implementation, USA, 2016, pp. 265-283.

[24] D. Asimopoulos et al., "Benchmarking advanced text anonymisation methods: a comparative study on novel and traditional approaches," in Proc. 13th International Conf. Modern Circuits and Systems Technologies, Sofia, Bulgaria, 2024.

[25] D. Singh and S. Narayanan, "Unmasking the reality of PII masking models: performance gaps and the call for accountability," arXiv:2504.12308 [cs.CL], 2025.

[26] X. Ma and E. Hovy, "End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF," in Proc. 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 2016, pp. 1064-1074.

[27] D. P. Kingma and J. Ba, "Adam: a method for stochastic optimization," in Proc. 3rd International Conf. Learning Representations (ICLR), San Diego, CA, USA, May 2015.

[28] M. Waqas and U. W. Humphries, "A critical review of RNN and LSTM variants in hydrological time series predictions," MethodsX, vol. 13, 2024.

[29] T. Wolf et al., "Transformers: state-of-the-art natural language processing," in Proc. 2020 Conf. Empirical Methods in Natural Language Processing: System Demonstrations, Online, 2020, pp. 38-45.

[30] C. Mansfield, A. Paullada, and K. Howell, "Behind the mask: demographic bias in name detection for PII masking," in Proc. 2nd Workshop on Language Technology for Equality, Diversity and Inclusion, Dublin, Ireland, 2022, pp. 76-89.

[31] J. Zhang, "Evaluating machine learning approaches for sensitive data identification: a comparative study of NLP and rule-based methods," Journal of Advanced Computing Systems, vol. 4, no. 7, pp. 26-38, 2024.