Named Entity Recognition for protecting sensitive data using Hybrid CNN

Sheela Gowr. P
Department of Computer science and Engineering
Vels Institute of Science Technology and Advanced Studies
Chennai, India
sheela.se@velsuniv.ac.in

Kumar. N
Department of Computer science and Engineering
Vels Institute of Science Technology and Advanced Studies
Chennai, India
kumar.se@velsuniv.ac.in

***Abstract* - Named entity recognition is a natural language processing technique that effectively recognizes and categorizes named entities in a document. The named entity recognition helps to bring out dynamic information about a document or gather critical data to store in a database. Deep learning helps to develop over time, while NLP examines the structure and standards of language and produces an automated system that can discern meaning from text. Mining the essential entities in a text helps identify related data, which is vital when functioning with enormous datasets. The proposed system has a feature that can retrieve and identify sensitive data such as PAN numbers, bank account numbers, and Aadhar numbers from unstructured text data.. The proposed model is designed using Hybrid CNN and it attains 95% F1-Score, 97.5% precision, and 98.3% recall.**

Keywords: Cloud Computing, Named Entity Recognition, spaCy, Convolutional Neural Network, Random Forest, Conditional Random Field.

I. INTRODUCTION

A named entity has a specific meaning, such as an individual's name, a location name, an association name, a period, and so on. The named entity recognition (NER) system includes new entity types based on the requirements from unprocessed input sentences. The concept of an entity might be fairly broad; it can be anything that is a specific piece of text that the business requires. The NER is based on the NLP task and has been utilized for a variety of purposes [1]. The cloud is a collection of hardware, networks, memory, operations, and protocols that allow the delivery of computing services.

In a traditional computer configuration, there must be proximity to the data storage device. There is no longer a need to be in the same physical area as the hardware that saves our data. The fundamental advantage of cloud computing in terms of cost is that customers only pay for what they use. Resources can be logged on from the cloud at any period and any place with an internet connection [2]. Security system management experts use a wide range of security data sources, in the natural language form, in both study and experimentation. This is problematic since the process takes a long time, and keeping up with constantly changing security threats, vulnerabilities, attacks, countermeasures, and hazards is challenging. Figure 1 shows the structural design of cloud data storage.

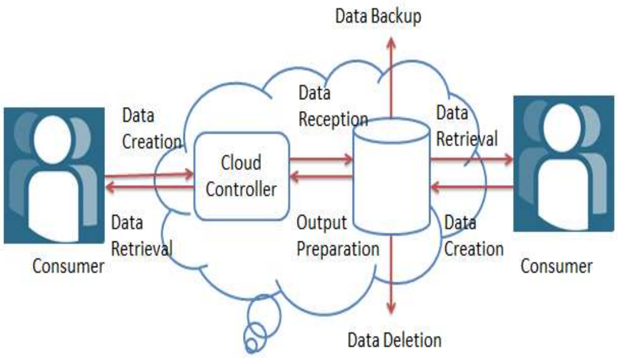

Figure 1. Structural design of cloud data storage

Cloud providers, from the attacker's perspective, combine access to numerous victims' data into a single point of entry. As cloud environments become more mainstream, they will become more vulnerable to cyber-attacks. Cloud data security is frequently used in authentication, data confidentiality and integrity, and other security mechanisms; security is today's most significant barrier in the cloud computing market [3]. When security vulnerabilities are uncovered, the service's reliability suffers severely. There are no universal standards or guidelines for installing cloud applications [4]. Many unique strategies are developed and correlated in the cloud; these methods fail to guarantee total security due to the challenging cloud environment.

II. LITERATURE SURVEY

Cloud computing has been a hot topic of research since its initial launch in 2000. Cloud computing makes virtual

978-1-6654-7467-2/23/$31.00 ©2023 IEEE

storage capacity, shared servers, networks, applications, and resources more accessible. When the entire network was endangered or hacked, CCAF multi-layered security provided protection. This technique protects all data in real-time, blocks threats, and quarantines the Data Center's petabyte systems. Our method selects the appropriate algorithms to increase data security performance in terms of execution speed and real-time virus/Trojan detection [5].

To determine the number of individuals login into the cloud data centre, the Map-Reduce framework is employed. Using the MetaCloudDataStorage interface, the framework safeguards the mapping of multiple data items to every source [6]. The strategy necessitates a significant amount of effort to execute, but it gives essential data for the cloud computing environment, which will substantially affect upcoming systems. The system focuses on ensuring data security by employing encryption techniques. The system considers the scenario with the arbitrator, who refused to give access to the user data.

The adoption of the AES encryption technique for transferring data eliminates the potential of the system becoming unavailable during the influx of massive data. The risk of intruders mimicking the guarantor and accessing the network is prevented by restricting access to the service provider. For cloud user data, the approach provides a cost-effective AES-based encryption strategy [7]. For data security and data integrity, a hierarchical identity-based cryptographic approach is utilized to ensure that there has been no tampering by a vicious attacker or CSP for its gain. The data integrity, signature creation, and authentication mechanism are secured and devised to assist in determining if the data attained is complete or has been modified by an invader [8].

The CSP's encrypted data storage decreases the threat of data effusion and loss, and the auditing obligation relieves the client and administrator of the data verification scheme. Heroku is a managed container system for deploying and hosting modern apps, with integrated data services and a robust ecosystem. Data security, which is handled through encryption methods, is an important concern in cloud computing. Advanced Encryption Standard is one way of encrypting data [9]. An algorithm based on genetics (GA) CryptoGA is a program that deals with data integrity and secrecy concerns. It creates encryption and decryption keys combined with a cryptographic technique to secure cloud data confidentiality. The approach protects the user's data from unauthorized parties by ensuring its integrity and privacy [10].

A hybrid method for safeguarding cloud data that employs three separate security measures for various sensitive data to give data administrators the most control over the data's storage, processing, and access. The data is first categorized based on its sensitivity and relevance, and then cryptographic techniques like AES, SHA-1, and ECC. For each encryption and decryption, two different keys are used. A cloud user must first register with the cloud service provider and the cloud owner. To access the encrypted cloud data, the individual must first register and receive a user login ID, password, and a One Time Password delivered to the individual's mobile number [11].

The system allows for finer data access control and protects against a variety of assaults in the cloud. The decryption key is encrypted using the IDTRE technique. The proposed scheme's performance is assessed using the PBC library [12]. Users are unable to extract content on their initiative and must rely on data owners to do so. The spaCy NER methodology makes the hiring process simple by automatically extracting the required entities from resumes. Recruiters can choose the required applicants based on the scores rather than sifting through reams of resumes from unqualified prospects [13]. A gold standard annotated with individual names was created using a rule- and lexicon-based method. It was also used to generate training data for various annotation levels, and to train the Stanford and spaCy Named Entity Recognition (NER) systems. All of the models, as well as a rule- and lexicon-based system, were tested on example texts: a portion of the gold standard and a similar-sized autonomous newspaper text [14]. The models are integrated into the NER&Beyond Web platform, which provides a variety of NE-related functions.

III. PROPOSED SYSTEM

The method of automatically classifying entities into pre-defined groups such as 'person,' 'organization,' 'place,' and so on is known as named-entity recognition (NER). The spaCy package allows training NER models by modifying a current spaCy model to fit the environment of text documents or creating a new NER model. In this technique, the text is entered into the model as a token of a corpus, or a class of the NLP job in the form of distinct integer values for each input. The model has a lexicon containing hashed values and associated vectors at this point. Values flow through the Convolutional Neural Network and Long Short Term Memory and are integrated with their context to exploit adjacent vectors in the encoding state. The matrix must first be processed via a Hybrid CNN Attention Layer, which uses a query vector to summarize the input before an ID can be predicted. During prediction, a softmax function is utilized to anticipate a super tag using part of speech and morphology information. Figure 2 shows the proposed cloud data security model.

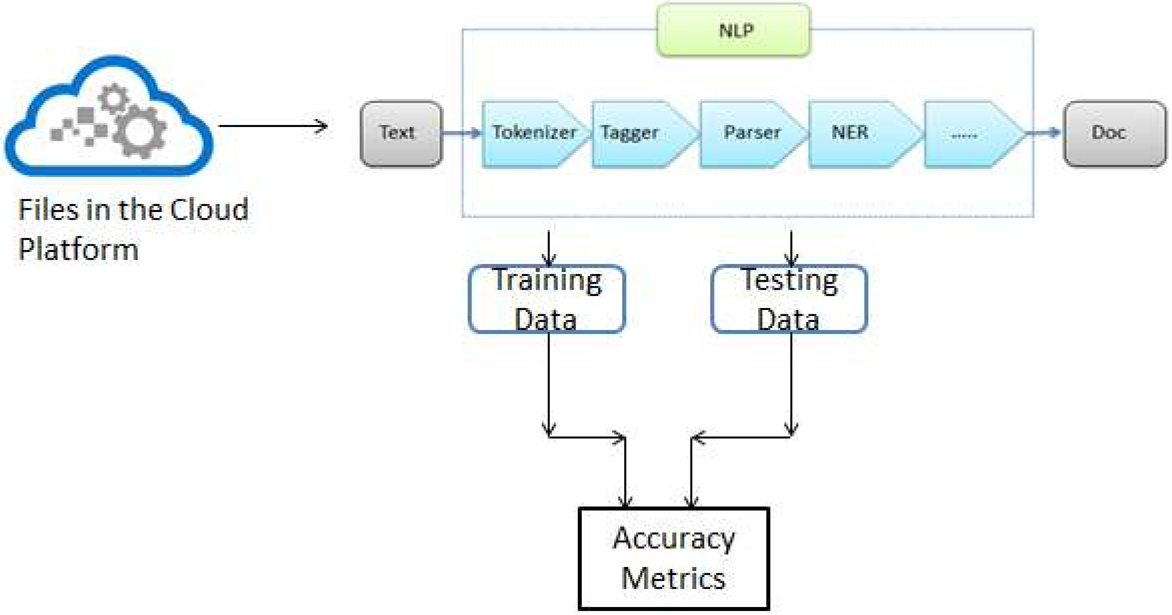

Figure 2 Cloud Data Security Architecture

A. Preprocessing Module

The primary part of the automated system is the preprocessing Module. It prepares and treats the input data, and executes preprocessing activities, to use it as input to the following module. spaCy is a Python library built-in for progressive Natural Language Processing (NLP). It is open-source. It makes life easier when there is a lot of text to work with [15]. spaCy is a tool used to generate apps and manage a large quantity of text. It is used to extract data, natural language perception systems, and text preprocessing. Numerous linguistic annotations are available in spaCy to assist in understanding a text's sentence rules. It includes many different types of words, such as parts of speech, and the relationships between them. A Doc keeps the source text information, such as whitespace characters, even after it is processed. A tagged corpus is a set of documents that have several entity-type annotations. Table 1 shows a few often-used datasets.

Table 1. Collection of Dataset

Corpus	Year	Text Source	#Tags	URL
MUC-6	1995	Wall Street Journal	7	https://catalog.ldc.upenn.edu/LDC2003T13
CoNLL03	2003	Reuters news	4	https://www.clips.uantwerpen.be/conll2003/ner/
OntoNotes	2007 – 2012	Magazine, news, web, etc.	18	https://catalog.ldc.upenn.edu/LDC2013T19
W-NUT	2015 – 2018	User-generated text	10-Jun	http://noisy-text.github.io
WiNER	2012	Wikipedia	4	http://rali.iro.umontreal.ca/rali/en/winer-wikipedia-for-ner
N3	2014	News	3	http://aksw.org/Projects/N3NERNEDNIF.html
Gillick	2016	Magazine, news, web, etc.	89	https://arxiv.org/e-print/1412.1820v2

FG-NER	2018	Various	200	https://fgner.alt.ai/
NNE	2019	Newswire	114	https://github.com/nickyringland/nested_named_entities

The component's input is made up of text and tables that can be found in a document. The main purpose is to identify and classify sensitive data based on its class. A NER system pipeline comprises data preprocessing such as tokenization, sentence splitting, feature extraction, applying models to the data for tagging, and then post-processing to eliminate any tagging discrepancies. This pipeline is depicted in Figure 3.

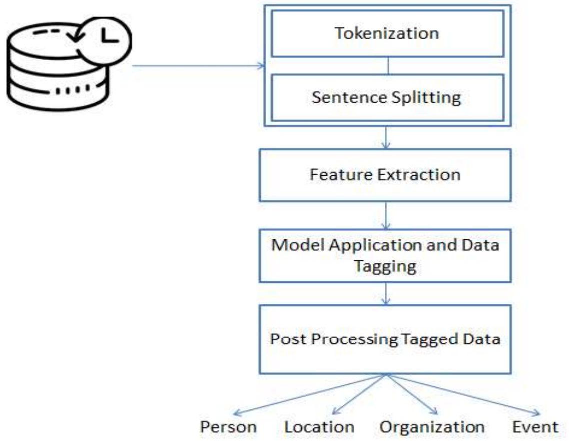

Figure 3. NER Pipeline

Tokenization: The text is tokenized using spaCy, which divides it into words, accents, and other aspects. It is achieved through the use of language-specific instructions [16]. Whitespace characters are used to divide the raw text. From left to right, the tokenizer processes the text. It conducts two checks on each substring:

1. Is there a tokenizer exception rule that matches the substring?

2. Is it possible to separate a prefix, suffix, or infix?

The operation is executed when a match is discovered, and the tokenizer loop continues with the newly divided substrings. SpaCy separates complex, encapsulated tokens such as contractions and abundant punctuation marks in this manner.

Parse and Tag: After a Doc has been tokenized, SpaCy will parse and tag it. SpaCy's trained pipeline and statistical models invade, enabling it to guess whichever tag or label is used in a given scenario. A trained component comprises binary data formed by providing a system with sufficient samples to make generally valuable predictions. [17].Token properties are manageable for linguistic annotations. To save memory and increase efficiency, spaCy converts all strings to hash values.SpaCy is capable of comparing two things and predicting their similarity. It is helpful to predict similarity when creating recommendation systems or reporting duplication.

When spaCy calls Natural Language Processing on words, it first tokenizes it to create a document object. It is then processed through a set of phases known as the processing pipeline. The training pipelines commonly include a tagger, a lemmatizer, a parser, and an entity recognizer. Every component in the pipeline returns the processed Doc, which is then forwarded to another component [18]. The Preprocessing Module output is the input of this module. The words are classed with their corresponding classifications as a result of this module's output. In this module, the entity types to identify were described by sensitive data. The set of entity classes evaluated in this research is shown in Table 2.

Table 2. Set of Entity Classes

Type	Description
PERSON	Individuals, as well as unreal.
ORG	Enterprises, assistances, institutions, etc.
FAC	Constructions, landing field, roads, bridges, etc.
EVENT	Named hurricanes, encounter, warfare, athletics, etc.
PRODUCT	Objects, Automobiles, nutriments, etc.
LANGUAGE	Any named language.
WORK_OF_ART	Book titles, tunes, etc.
QUANTITY	Measurements.
LAW	Named documents made into laws.
PERCENT	Percentage.
MONEY	Financial values, comprising unit.
TIME	Times lesser than a day.
DATE	Absolute or comparative dates or periods.
PAN	PAN card Number
AADHAR	Aadhar card Number
BANK	Bank Account Number

B. NER Model

a. Conditional Random Field

NER techniques are based on statistical models that often necessitate considerable training data. These haven't been employed as much as they may be to avoid the timeconsuming annotation process.. The undirected graphical model is used in NER employing a Conditional Random Field. It defines output nodes given values on other selected input nodes determined using CRF [19]. It incorporates context-dependent learning and dependent features. Figure 4 shows the working of CRF.

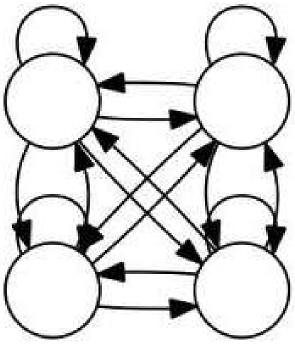

Figure 4. Conditional Random Field

The input sequence for the model is a = (a_1,…, a_m), where a represents the collection of ordered words that make up a phrase. As the output sequence, b = (b_1,…, b_m) states that correspond to named entity classes, that correspond to entity classes that match a.

$$P(b|a) = \frac{1}{z_x}\exp(\sum_{m=1}^{M}\sum_{k}\beta_k f_k\,(b_{m-1}, b_m, a, m))\text{…….(1)}$$

Here f_k(bm-1, bm, a, m) is a random feature function, where f_x is a normalization factor, k is a learning weight for each feature function, and Z_x is a normalization factor. F_k ( ) can be anywhere between –…+. The features, f_k, are determined by the collection of features that have been employed.

b. Random Forest

The Random Forest concept is based on decision trees. The model is trained with a random subset of inputs to build decision trees. To acquire a thorough understanding of the decision-making process, the instigated model is a simple tree-based classification model consisting of various deep trees, each of which is trained using a random data selection [20]. Figure 5 shows random forest classification.

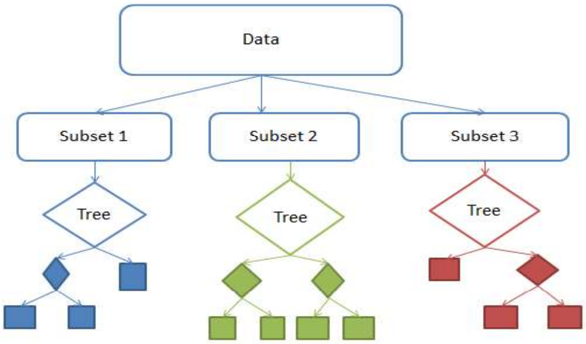

Figure 5. Random Forest Classifier

Starting from the root to the leaf, each tree follows a path that is made up of a succession of decisions, each of which adds to the ultimate projections. The model with x leaves distributes the feature space into X regions in this case, 1≤ n≤ N.

$$f(x) = \sum_{n=1}^{N} c_n\, X(a, R_n)\text{ ……(2)}$$

N is the total number of leaves in the tree, R is a feature space region denoting leaf n, c is a constant related to region n, and X is the indicator function. The value of c is decided during the tree's training phase, while R stands for the extracted features. The input must be transformed into a basic feature vector for every word before training the Random Forest model.

c. Hybrid CNN

In this work, a hybrid technique for automatically diagnosing sensitive information was developed. Figure 6 illustrates how CNN and LSTM networks were combined to form this technology's framework, with the CNN retrieving complex information from the dataset and the LSTM functioning as a classifier.

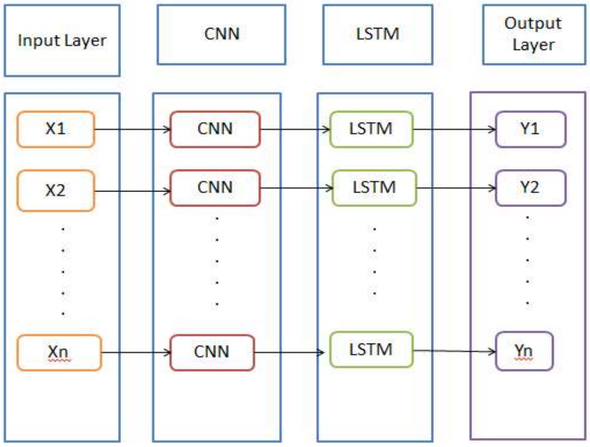

Figure 6 Architecture of CNN-LSTM network

An input layer, four CNN layers, an LSTM layer, a dense layer, a dropout layer, and an output layer make up the design of our system. Before building the model, the input has to be verified that all has been converted into a text format. The next vectorization stage used one-hot encoding to convert the padding sequences into a mxn matrix. The relu activation function is utilized to extract features. Before sending all the collected characteristics to the LSTM layer, they are combined using a flattened layer. Binary classification outputs were defined using the softmax activation function.

If the divergence differs, the training of the system can be finished more rapidly since the testing dataset takes account of how far along it is. Two or more one-hot encoded label classes resulted from categorical cross-entropy. Adam produces values for adaptive learning for every momentum-like parameter. It employs a softmax activation function to optimize multi-class classification models.

IV. EXPERIMENTAL RESULTS

Statistical models empower spaCy's tagger, parser, text categorizer, and many other features. Every decision these components make is a prediction depending on the current weight values, such as which part-of-speech tag to apply. The weight values are calculated using examples encountered by the model in training. The training data is needed to train a model that includes text samples and labels. It could be a named entity, a part-of-speech tag, etc. It is critical to analyze and compare new methodologies' findings in any research field. As a result, an objective measure is required to adequately cover the research's objectives. In contrast to several other NLP tasks, NER employs a well-accepted set of metrics. This set contains three metrics for describing the NER system's performance, each for a distinct part of the task. Precision, recall, and F-measure are the measurements' terms (F-score or F1 score). Precision, recall, and F-measure can now be defined as follows.

$$Precision = \frac{Positive}{Positive + False\ Positive} \quad (3)$$

$$Recall = \frac{Positive}{Positive + False\ Negative} \quad (4)$$

$$F - measure = \frac{2\ X\ Preision\ X\ Recall}{Precision + Recall} (5)$$

Precision is a level of trust in the things that have been labelled as positive. The recall is a measure of confidence in the fact that all of the positive objects have been marked. Precision and recall characterize different characteristics of outcomes. Furthermore, these measures are hostile. Figure 7 shows the detection of bank account numbers using Hybrid CNN. The Named Entity Recognition Model is used to extract sensitive data. The hybrid CNN is used for sensitive data classification in the proposed system.

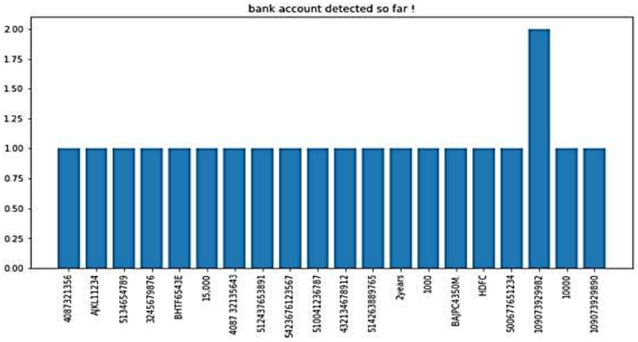

Figure 7 Bank Account Dataection

NER is usually handled as a sequence labelling issue, with models assessed using traditional classification metrics like precision, recall, and F-score. The NER task was approached using various CRF, RF, and hybrid methods. Entire tests and trials were run on a single machine and were written in Python. The evaluation measures are shown in Table 3.

Table 3. Model Evaluation using different Metrics

Evaluation Using Different Metrics				
S.No	**Methods**	**Precision (%)**	**Recall (%)**	**F1 score (%)**
1	Conditional Random Field	85.2	87.8	86.48
2	Random Forest	84	86.8	82.49
3	Hybrid CNN	97.5	98.3	95

A confusion matrix is a classification performance measurement technique. It's a kind of table that helps us to evaluate how well a classification model performs on several test data for which the true values are known. The confusion matrix visualizes a classifier's accuracy by assessing the current and predicted classifications. The squares that comprise the binary confusion matrix are shown in Figure 8.

	Class 1 Predicted	Class 2 Predicted
Class1 Actual	True Positive 4931	False Negative 69
Class2 Actual	False Positive 21	True Negative 4979

Figure 8. Confusion Matrix

Data handling is a difficult task as data volumes have increased. As businesses migrate to the cloud, a greater emphasis is placed on ensuring everything is safe and secure, with no chance of data hacking or breaches. There is often a risk of data misuse when multiple firms use the cloud to store their data. Data repositories must be secured immediately to avoid the risk. Figure 9 shows the model performance against different entities. We have checked various entities for the different classifiers. The Hybrid CNN shows better performance on accuracy metric It shows that the hybrid CNN can detect sensitive information better than other classifiers.

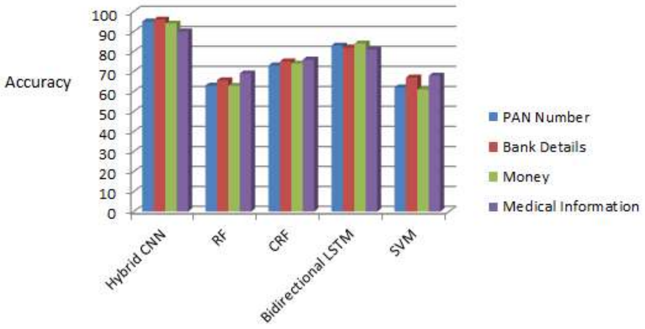

Figure 9. Performance on different Entities

V. CONCLUSION

In this research, a uniquely protected and efficient approach for data access control is proposed, which protects against a variety of assaults in the cloud. The major purpose of this project was to use spaCy and Hybrid CNN to construct a Named Entity Recognition system for cloud data security. The suggested model is compared to CRF and RF to model and predict the identified entities. According to the experimental data, the proposed technique is more productive than the current systems. The proposed NER recognition approach based on the Hybrid CNN is beneficial in enhancing performance and lowering training complexity through experimentation.The deep learning model surpassed the statistical-based approach in our experiment.

FUTURE SCOPE

This paper consideres data from an online platform that was in the form of English sentences for this study. As a result, documents in other languages do not fall under the sensitive classification; further research may use text documents that are offered in additional languages or a combination of several languages. In future work, we intend to develop a sophisticated framework that can more accurately filter sensitive documents in cloud platforms.

REFERENCES

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

[2] A. Hamdi *et al.*, "A multilingual dataset for named entity recognition, entity linking and stance detection in historical newspapers," in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021, pp. 2328–2334.

[3] A. C. Rouhou, M. Dhiaf, Y. Kessentini, and S. Ben Salem, "Transformer-based approach for joint handwriting and named entity recognition in historical document," *Pattern Recognit. Lett.*, vol. 155, pp. 128–134, 2022.

[4] B. Song, F. Li, Y. Liu, and X. Zeng, "Deep learning methods for biomedical named entity recognition: a survey and qualitative comparison," *Brief. Bioinform.*, vol. 22, no. 6, p. bbab282, 2021.

[5] V. Chang and M. Ramachandran, "Towards Achieving Data Security with the Cloud Computing Adoption Framework," *IEEE Trans. Serv. Comput.*, vol. 9, no. 1, pp. 138–151, 2016, doi: 10.1109/TSC.2015.2491281.

[6] G. Manogaran, C. Thota, and M. V. Kumar, "MetaCloudDataStorage Architecture for Big Data Security in Cloud Computing," *Procedia Comput. Sci.*, vol. 87, pp. 128–133, 2016, doi: 10.1016/j.procs.2016.05.138.

[7] K. M. Akhil, M. P. Kumar, and B. R. Pushpa, "Enhanced cloud data security using AES algorithm," in *Proceedings of 2017 International Conference on Intelligent Computing and Control, I2C2 2017*, 2018, pp. 1–5, doi: 10.1109/I2C2.2017.8321820.

[8] S. Kaushik and C. Gandhi, "Ensure Hierarchal Identity Based Data Security in Cloud Environment," *Int. J. Cloud Appl. Comput.*, vol.

9, no. 4, pp. 21–36, 2019, doi: 10.4018/ijcac.2019100102.

[9] T. H. Dang, H. Le, T. M. Nguyen, and S. T. Vu, "D3NER: biomedical named entity recognition using CRF-biLSTM improved with fine-tuned embeddings of various linguistic information," *Bioinformatics*, vol. 34, no. April, pp. 3539–3546, 2018, doi: 10.1093/bioinformatics/bty356.

[10] M. Tahir, M. Sardaraz, Z. Mehmood, and S. Muhammad, "CryptoGA: a cryptosystem based on genetic algorithm for cloud data security," *Cluster Comput.*, vol. 24, no. 2, pp. 739–752, 2021, doi: 10.1007/s10586-020-03157-4.

[11] V. Goyal and C. Kant, "An effective hybrid encryption algorithm for ensuring cloud data security," *Adv. Intell. Syst. Comput.*, vol. 654, no. 2, pp. 195–210, 2018, doi: 10.1007/978-981-10-6620-7_20.

[12] S. Namasudra, "An improved attribute-based encryption technique towards the data security in cloud computing," *Concurr. Comput. Pract. Exp.*, vol. 31, no. 3, pp. 1–15, 2019, doi: 10.1002/cpe.4364.

[13] K. H. K.Satheesh, A.Jahnavi, L.Iswarya, K.Ayesha, G.Bhanusekhar, "Resume Ranking based on Job Description using SpaCy NER model," *IRJET*, vol. 7, no. 5, pp. 74–77, 2020, [Online]. Available: https://www.irjet.net/volume7-issue5.

[14] B. Šandrih, C. Krstev, and R. Stanković, "Development and evaluation of three named entity recognition systems for Serbian - The case of personal names," in *International Conference Recent Advances in Natural Language Processing, RANLP*, 2019, pp. 1060–1068, doi: 10.26615/978-954-452-056-4_122.

[15] D. Farmakiotou, V. Karkaletsis, J. Koutsias, G. Sigletos, C. D. Spyropoulos, and P. Stamatopoulos, "Rule-based named entity recognition for Greek financial texts," in *Proc. of the Workshop on Computational lexicography and Multimedia Dictionaries (COMLEX 2000)*, 2000, pp. 75–78.

[16] M. Tanenblatt, A. Coden, and I. Sominsky, "The ConceptMapper approach to named entity recognition," *Proc. 7th Int. Conf. Lang. Resour. Eval. Lr. 2010*, pp. 546–551, 2010.

[17] J. R. Finkel and C. D. Manning, "Joint parsing and named entity recognition," in *Human Language Technologies: The 2009 Annual Conference ofthe North American Chapter ofthe ACL*, 2009, no. June, pp. 326–334, doi: 10.3115/1620754.1620802.

[18] L. Tanabe, N. Xie, L. H. Thom, W. Matten, and W. J. Wilbur, "GENETAG: A tagged corpus for gene/protein named entity recognition," *BMC Bioinformatics*, vol. 6, no. 1, pp. 1–7, 2005, doi: 10.1186/1471-2105-6-S1-S3.

[19] F. Souza, R. Nogueira, and R. Lotufo, "Portuguese Named Entity Recognition using BERT-CRF," *Comput. Lang.*, vol. 1, no. 1, pp. 1–8, 2019, [Online]. Available: https://doi.org/10.48550/arxiv.1909.10649.

[20] R. Priyadharshini, B. R. Chakravarthi, M. Vegupatti, and J. P. McCrae, "Named Entity Recognition for Code-Mixed Indian Corpus using Meta Embedding," in *2020 6th International Conference on Advanced Computing and Communication Systems, ICACCS 2020*, 2020, pp. 68–72, doi: 10.1109/ICACCS48705.2020.9074379.