[1]Preeti S. Joshi

[2]Dinesha H.A.

Multidimensional Forensic Investigation of Onion Sites Based on Fuzzy Encoded LSTM

Abstract: **-** The only way to access onion services is via the TOR browser providing anonymity and privacy to the client as well as the server. Information about these hidden services and the contents available on them cannot be gathered like websites on the surface web. So, they become a fertile ground for illegal content dissemination and hosting for cybercriminals. There is a persistent need to classify and block such content from onion sites. In this paper, we investigate data requested from onion services to help law enforcement agencies collect traces of cybercrime on these hidden services. We propose a system using fuzzy encoded LSTM to analyze contents retrieved from these sites and raise alerts if found illegal. The accuracy of fuzzy-encoded LSTM is found to be 81.04 % and it outperforms other classifiers.

Keywords: Onion sites, TOR network, fuzzy encoded LSTM

I. INTRODUCTION

The web provides access to multiple resources and services. Of these, those that are accessible only through The Onion Router (TOR) are called hidden services (HS) or onion services (OS). The domain name for these services is. onion and the task of connecting to onion sites is taken care of by directory service called Hidden Services Descriptors. HS cannot be accessed by IP addresses and their IP addresses cannot be traced due to onion routing protocol over the Tor overlay network, neither are these HS indexed by search engines to search them. The anonymity and privacy provided by these services make them a choice for illegal activities. With the fear of entrapment, these sites have limited lifetime and intermittent appearances on the web. Current and historical statistical information about the public TOR network can be referred to in [1]. Since October 2021, onion service Version 2(V2) is no longer supported by TOR and all current (approximately 6 to 8 lakh) onion sites fall under Version 3(V3). In V2 length of the address was 16 characters in V3 56 characters. Apart from this many modifications are made in V3 to preserve anonymity and privacy, like now in V3 onion services, mass collection of onion site information is avoided by generating a daily-rotated identification using key derivation called a blinded public key. [2] Mentions flaws in V2 and improvements in V3. If content from a website is illegal it should be identified at the client end device. Forensic tools will not be able to solve the purpose. We propose a deep learning algorithm to block such illegal content on client machines. The paper is organized as follows: Section II- A presents a study of review and research carried out to deanonymize the TOR network and accumulate information from onion sites and Section II-B introduces Natural Language processing and presents a review of deep learning algorithms for the same. Section III presents the proposed system for the identification of illegal content.

[1] Research Scholar, Dept. of CSE, VTU, Belgavi, Karnataka, India & Assistant Professor, Dept. of IT, Marathwada Mitramandal's College of Engg. Pune, India

preetijoshi@mmcoe.edu.in

[2]Professor (CSE) and Dean (R&D), Shridevi Institute of Engineering and Technology & Founder and Chief Executive Director, Cybersena (R&D) India Private Limited.

Tumakuru, Karnataka, India dineshameet@gmail.com

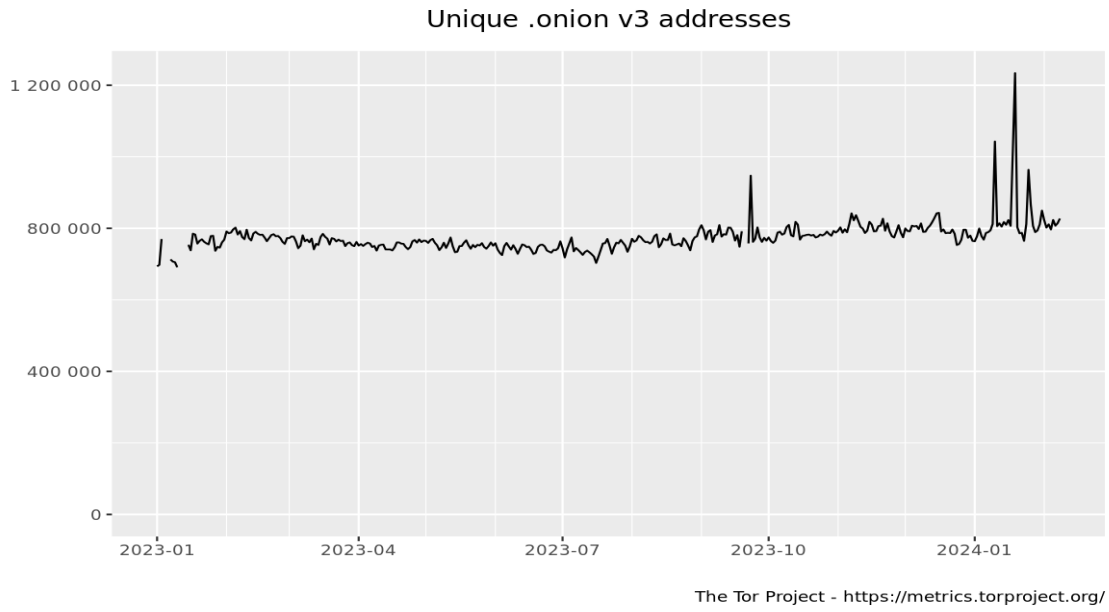

Figure 1.1 Number of unique. onion V3 addresses

II. RELATED WORK

A. Onion site investigation

TOR network has been an area of research due to the anonymity and privacy provided by it. Major research work on the TOR network is for deanonymization by traffic analysis [3][4][5]. In [6] the authors take a review of anti-forensic techniques provided by the TOR network and a detailed analysis of deanonymizing techniques implemented by researchers.

Onion services hosted on the TOR network are researched to find their popularity, and content on these sites. Bernaschi et.al in [7] have analyzed topological graphs of the TOR network and semantic analysis of TOR web pages. In [8] authors collected HSDir by finding flaws in protocol and implementation by shadowing technique, explored and analyzed the expanse of Tor hidden services. Scanned open ports and found that the most popular onion addresses are command and control centers of botnets and resources serving adult content.

A list of hidden services can be obtained by collecting data from hidden server descriptors HSDir. Hidden services are investigated by extracting onion addresses from HSDir in [8],[9], [10]. In [8] authors collected hidden services descriptors by exploiting flaws in the protocol and implementation of Tor and using the shadowing technique The expanse of Tor hidden services is explored and analyzed. It was found that the most popular onion addresses are command and control centers of botnets and resources serving adult content by scanning the ports

Website crawling is the software process of collecting web pages from websites to collect information.. It is seen that many researchers made use of crawlers to explore the onion sites, however crawling onion sites is more challenging than crawling surface web [11] [12] due to issues of scalability, content selection trade-off, social obligation, short life cycle of websites, accessibility through registration and login process, and denial to access to inactive members. We consider that intermittent appearance of sites and limited lifetime can also be a hurdle to crawl HS. Hidden services have been explored by many researchers by crawling the onion sites starting from seed sites. In [13] [14][15] to analyze the product prices and supplies on the sites, to rank HS based on the link-based approach, and to analyze the structure and privacy of Hidden services respectively. In [10] information on hidden services is extracted from descriptors onion addresses and crawled to find text types of services available on them, languages, popularity, up-time, and amount of service protected by descriptor cookie. In continuation to their previous work in [16] authors in [14] have contributed with a new dataset “Darknet Usage Text Addresses” DUTA10K[8] and ranking algorithm for HS and analyzed activities, content distribution, and languages on the web pages. The dataset is proposed since the lifespan of onion domains is very short. In[17] authors developed an efficient search engine based on a scrapy crawler for the TOR network to search illegal sites, the crawler ‘Black Widow’ achieved 240% improvement in the number of services indexed.

Table 2.1 Crawling Onion sites

Paper	Objective	Methodology	Tools	Outcome
[9]	Classify content of hidden services	Capture Data from DHT+ Custom crawler	Port scan to determine application running	●Darknet server configuration ●Popular content
[10]	Real time detection and analysis of onion services	Capture Data from DHT	20 volunteer relays	● Service Identification ● Service up-time, ● Language distribution, ● Top 10 onion services
[11]	To find illicit an extremist content	Crawling to find Keywords and Images on sites	Dark Crawler based on CENE	Content analysis of Onion sites
[12]	Crawling and investigating activities on darkweb	Crawling DOM,CSS Xpath data extraction	Darky crawler based on Scrapy	Challenges of Crawling onion sites
[13]	Framework to analyze product prices and Supplies in Darknet	Web Scraping	Selenium , Python library with Socks Support, RabbitMQ	Generic analysis Framework for TOR market
[14]	Ranking HS	Crawling + Analyze Hyperlinks to find Influential one	NetworkX library with Python Data Set used "DUTA-10K"	Rank top most influential onion domains
[15]	Structural and Privacy analysis of TOR HS	Crawled 99.46% of sites and not just home page	Developed Darkweb Crawler with PhantomJS	Size and coverage of onion sites
[17]	Recognize and index dark websites	Crawling	scrapy crawler	240% improvement Avoid crawler trap

The onion service has undergone a major change from version 2 to version 3. Earlier in V2 all onion addresses were in plaintext format and any relay with the HSDir flag set could collect the database of Hidden addresses which actually is a malicious behavior as per TOR, now in V3 addresses are stored in encrypted format and the V3 address is a public key by itself. Clients can always use the key stored in the .Onion address to decrypt that data. Clients must still request information from the directory regarding a specific onion address, which would once more enable mass collecting of onion addresses. With V3 onion services, this is avoided by generating a daily-rotated identification using key derivation called blinded public key [2]. With this change from version 2 to 3, earlier implementations of capturing onion site addresses and crawling will not work.

B. Review of Deep learning model implementation in NLP

The contents retrieved from the onion sites need to be classified as legal/permissible and illegal. This can be accomplished by intelligently processing it. A subfield of artificial intelligence called natural language processing (NLP) is responsible for processing, interpreting, and analyzing human language. Of the many techniques used in NLP are syntactic and semantic analysis, named entity recognition, summarization, keyword extraction, and text classification are used to classify text data of huge quantity. The different methods of text classification are: rule-based, machine learning algorithms, or hybrid methods. Manual searching or automated processes like rule-based classification is not sufficient. Multilayer perceptrons (MLPs) are the basic building blocks of neural networks that are utilized to automatically capture features for classification. Authors of [18] advocate deep learning for text classification to provide semantically meaningful representations for text mining. Deep neural networks can learn efficiently with feature extraction without having domain knowledge. Deep learning systems based on convolutional

neural networks (CNNs) and recurrent neural networks (RNNs) learn while they are working and aid to deduce meaning from raw and unstructured unlabeled text and voice data sets

CNN is used for character level classification as proposed in [19], [20] and sentence level classification in [21]. RNN for text classification is used in [22]. As proposed in [23] , models based on RNNs view text as a sequence of words and are intended to capture word dependencies and text structures for the purposes of text classification by capturing word dependencies. Variations in this model are MT-LSTM, and Bi-LSTM mentioned in [24][25][26] respectively. In[27] authors demonstrate that combining IndRNN with LSTM and attention model avoids suffering the gradient vanishing and exploding problem of training of the RNNs and better performance is achieved than the traditional RNN and LSTM models for text classification tasks[28-31]. CNN without activation function and LSTM improves performance as mentioned in [Research on text classification based on CNN and LSTM].

III. PROPOSED SYSTEM ARCHITECTURE

The anti-forensic techniques of the TOR network need more advanced tools and methods than the classical forensic tools to investigate illegal activities. We propose a deep learning-based architecture to identify illegal content on onion sites that will help law enforcement agencies in Figure 3.1.

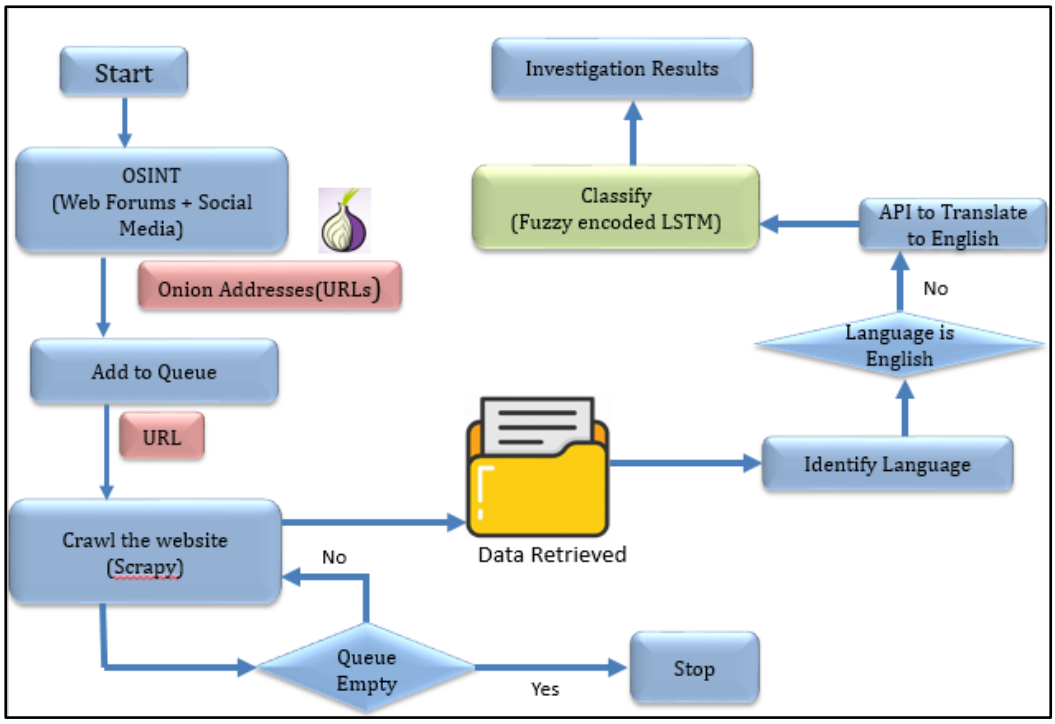

Figure 3.1 Proposed System Architecture

A. Feature Extraction and Normalization

Before processing, the raw data gathered from onion sites and online forums needs to be sanitized. Textual data is normalized by removing any unique characters and converting it to lowercase. The stop words were retained since understanding user attitudes depends on them. Once the text has been cleaned, it is tokenized in order to collect unigrams, or individual words, and determine where in the corpus they appear. The normalization process consists of tuning textual data to lowercase and removing the special characters. Because the stop words are crucial for understanding user attitudes, they were not eliminated. The cleaned text is next tokenized to collect unigrams (individual words) and determine their frequency across the corpus. This results in 55222 unique unigrams. This feature collection only includes unigrams with frequency greater than 250, yielding 325 unigram features. The same method yields 70 trigrams and 158 bigrams.

The task of categorizing data into multiple categories is not a simple binary classification exercise. Because a data developer may mention a range of subjects in his or her data, each data can be divided into several categories.

This strategy can be used with any binary classifier, including decision trees, LSTM, nearest neighbour, and so on, once the dataset has been divided into four distinct datasets. Even yet, this approach is simple and looks at each group independently. It ignores the relationship between categories as a result. This might not be the case, especially if there are similarities between the groups.

The amount of unstructured data that is currently available is enormous, and it continues to increase. Since almost all algorithms used in machine learning and deep learning rely on in-memory analytics, training data must be kept in memory. There is a finite quantity of training data that can be utilized to develop ML/DL models. Models should therefore be updated either in large batches or piecemeal. In the future, new data with relatively different contexts will be available. Therefore, it is important to investigate these traits in-depth without having to retrain the current model. Figure 3.2 represents the Fuzzy encoded LSTM. Input to it is sequential data and output it is in the form of reconstructed results.

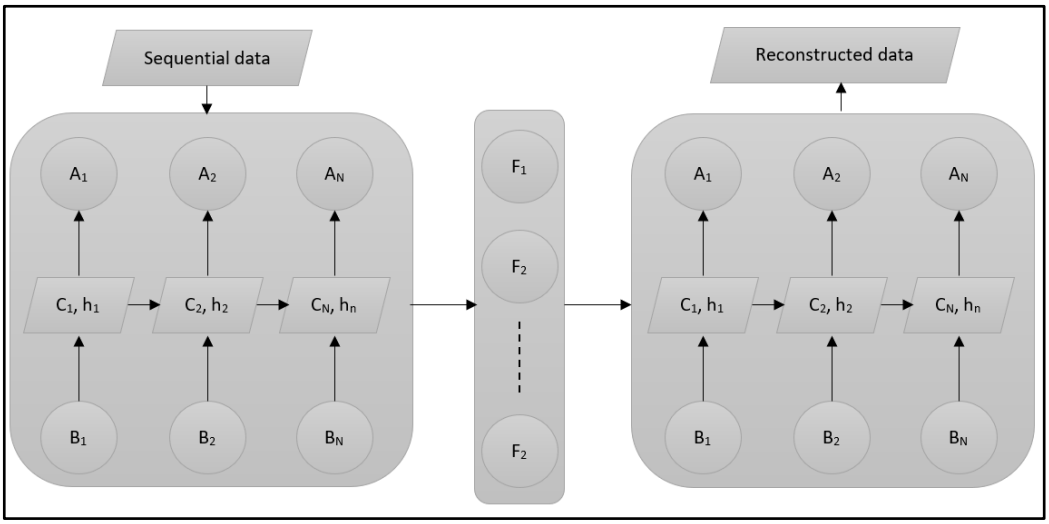

Figure 3.2 Fuzzy encoded LSTM

The proposed Fuzzy encoded LSTM indexFL_k, which is also the triangular membership function, is illustrated in following equation,

$$FL_k = (FL_k^L, FL_k^M, FL_k^U) = \left(\left(FL_{k+(T-W+1)\times m}, FL_{k+(T-W+2)\times m}, \dots, FL_{k+T\times m}\right), FL_k, \left(FL_{k+(T-W+1)\times m}, FL_{k+(T-W+2)\times m}, \dots, FL_{k+T\times m}\right)\right), k = 1, \dots, m \quad (1)$$

Where FL_k^L, FL_k^M, FL_k^U are the W-period lower bound, W-period smoothing-operators($1 \le W \le T$), and W-period upper bound, respectively.

It's critical to identify both similarities and differences inside the cluster in order to classify the text reviews first. To do this and obtain the most accurate results possible, the fuzzy encoded LSTM is employed in recursive mode. Consequently, a detailed analysis of the triangular fuzzy function—which is utilized for soteristics—should be conducted without retraining the current model.

$$FLSI_{k+(T+v)}^I \sim = \left(flsi_{k+(T+v)}^{LTr} \times FL_k^L \times \varepsilon, flsi_{k+(T+v)}^{MTr} \times FL_k^M \times \varepsilon, flsi_{k+(T+v)}^{UTr} \times FL_k^U \times \varepsilon\right) \quad (2)$$

Where$FLSI_{k+(T+v)}^I \sim$ gives the fuzzy similarity index value, $flsi_{k+(T+v)}^{LTr}$ represents the minimum allowable similarity index, $flsi_{k+(T+v)}^{MTr}$ mean value of similarity index and $flsi_{k+(T+v)}^{UTr}$ maximum allowable similarity index.

Therefore, the mathematical model of the fuzzy encoded LSTM is with minimum, mean and maximum value is represented as,

$$flf_{LTr}(x_i)\, o_{Li} \otimes tan\ h(c_{Li})\, flf_{MTr}(x_i)\, o_{Mi} \otimes tan\ h(c_{Mi})\, flf_{UTr}(x_i)\, o_{Ui} \otimes tan\ h(c_{Ui}) \quad (3)$$

Equation 3 clearly indicates that the flf_{LTr}, flf_{MTr} and flf_{UTr} is of the tan hyperbolic in nature. The fuzzy rules selected for encoding are tan hyperbolic in nature.

B. Result and Discussion

The initial step in determining the performance parameters is to compute a confusion matrix. Thus, performance parameters are calculated using equations 4 through 7. Figure 3.3 illustrates the confusion matrix followed by performance parameters equations.

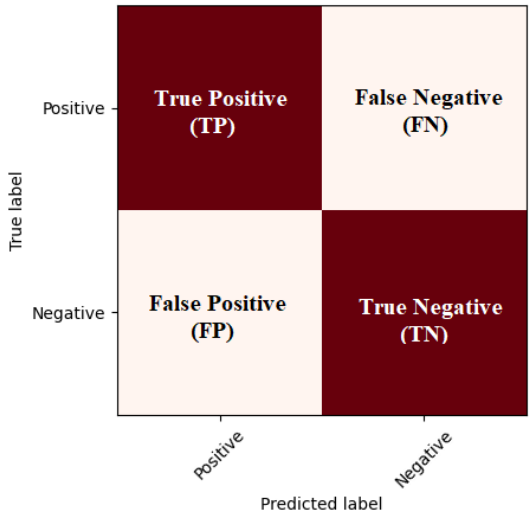

Figure 3.3 Standard confusion matrix

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad \text{…equation (4)}$$

$$Recall = \frac{TP}{TP+FN} \quad \text{…equation (5)}$$

$$Precision = \frac{TP}{TP+FP} \quad \text{…equation (6)}$$

$$F1score = \frac{TP}{TP+FP} \quad \text{…equation (7)}$$

The proposed system architecture (PSA) is tested on two classes' first, viz. "Normal Website" and "porn website". Then the third class is added to the system. The third added class is "Violence Website". The performance parameters of the system for 3 classes are tabulated in the following table. The performance parameters of the PSA are graphically presented in figure 5. The time [32] required to get the result of the classification is 0.053 seconds.

Table 3.1 Performance parameters of the PSA for the classification of the type of Website

parameter		Value (%)
Accuracy	All Classes	81.04
Precision	Normal Website	73.76
	Porn Website	71.75
	Violence Website	87.92
Recall	Normal Website	74.76
	Porn Website	68.45
	Violence Website	98.67
F1 score	Normal Website	72.96
	Porn Website	61.48
	Violence Website	91.89

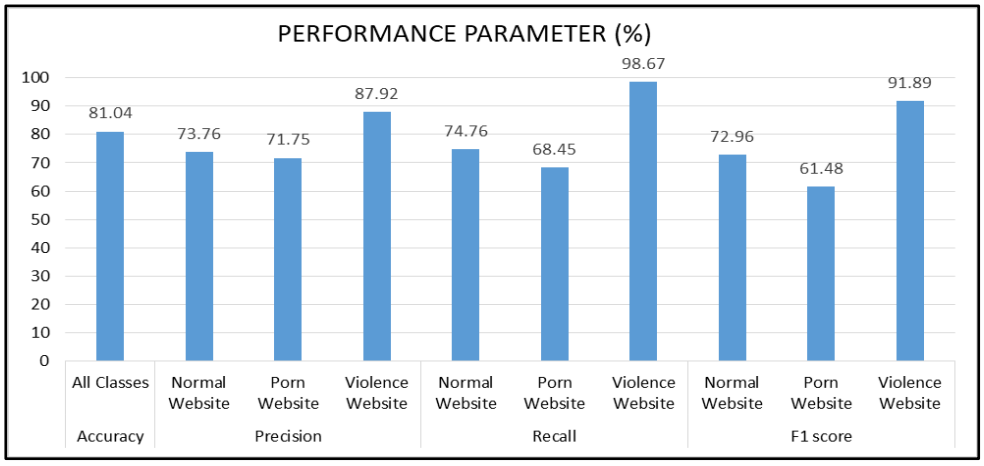

Figure 3.4 Performance parameters of the classification of the type of website using PSA

The results of the PSA are compared with other existing algorithms. The comparison of the PSA with other algorithms is tabulated in Table 3.2.

Table 3.2 Comparison of the PSA with other existing algorithms

	Accuracy	Precision			Recall			F1 Score		
	`All Classes	Normal Website	Porn Website	Violence Website	Normal Website	Porn Website	Violence Website	Normal Website	Porn Website	Violence Website
PSA	81.04	73.76	71.75	87.92	74.76	68.45	98.67	72.96	61.48	91.89
ContextAvg	73.48	56.48	51.79	80.49	55.61	29.59	90.11	56.04	37.66	85.03
AEContextAvg	75.27	62.09	55.47	81.36	57.65	36.22	90.52	59.79	43.83	85.7
LSTM	77.23	63.35	54.55	84.73	61.73	39.8	91.48	62.53	46.02	87.98
GRU	78.75	67.36	59.84	84.35	66.33	37.24	93.27	66.84	45.91	88.58
BiGRU	77.14	64.94	53.69	84.19	57.65	40.82	92.17	61.08	46.38	88
BiLSTM	78.3	65.13	56.64	85.55	64.8	41.33	91.9	64.96	47.79	88.61
TD-LSTM	78.66	72.88	54.55	85.09	65.82	45.92	90.93	69.17	49.86	87.92
TC-LSTM	77.41	67.78	55.7	83.69	62.24	42.35	90.93	64.89	48.12	87.16
AT-LSTM	78.04	70.06	61.25	81.23	67.27	25	96.29	66.49	35.51	88.12
AT-GRU	78.3	67.91	61.21	83.11	64.8	36.22	93.27	66.32	45.51	87.9
AT-BiGRU	77.77	65.13	59.84	83.56	64.8	37.24	92.17	64.96	45.91	87.66
AT-BiLSTM	78.84	68.45	67.82	82.27	65.31	30.1	95.6	66.84	41.7	88.44

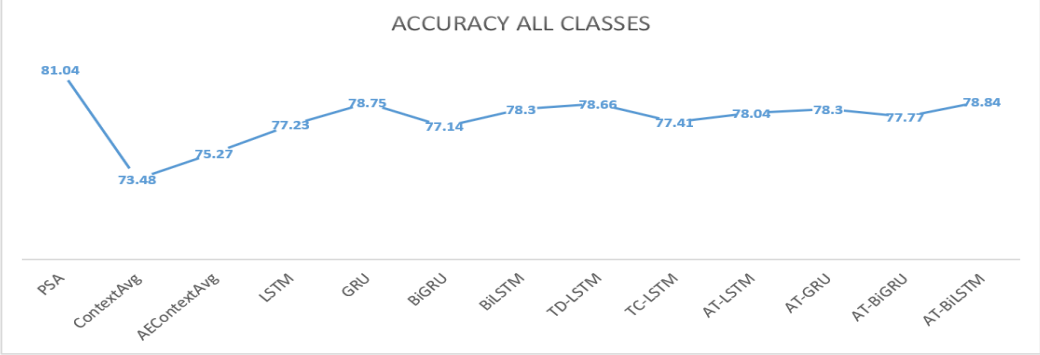

Figure 3.5. Accuracy of the Classification of the types of websites using different algorithms

IV. CONCLUSION

Onion sites over the dark web have been researched for identification of contents available on it using forensic techniques and recently machine learning. In several text categorization tasks, deep learning-based models have demonstrated superior performance compared to conventional machine learning-based techniques. The dataset of onion sites was built to support classification from web forums and social media sites. An algorithm for automatic labeling was developed in this study instead of carrying out the labeling task by hand, and the algorithm showed higher performance in terms of accuracy and effectiveness. These three classes—along with the others that are solely connected to illicit activity—violence, pornography, and bidding—were chosen, and the suggested model was trained using them. When it comes to text classification on onion sites, the suggested fuzzy encoded LSTM algorithm performs better than others.

REFERENCES

[1] https://metrics.torproject.org/

[2] Tobias Hoeller, Michael Roland, and René Mayrhofer. 2021. On the state of V3 onion services. In Proceedings of the ACM SIGCOMM 2021 Workshop on Free and Open Communications on the Internet. Association for Computing Machinery, New York, NY, USA, 50–56. https://doi.org/10.1145/3473604.3474565

[3] Montieri, D. Ciuonzo, G. Aceto and A. Pescapé; "Anonymity Services Tor, I2P, JonDonym: Classifying in the Dark (Web)"; in IEEE Transactions on Dependable and Secure Computing, vol. 17, no. 3, pp. 662-675, 1 May-June 2020, doi: 10.1109/TDSC.2018.2804394.

[4] Milad Nasr, Alireza Bahramali, and Amir Houmansadr. 2018. DeepCorr: Strong Flow Correlation Attacks on Tor Using Deep Learning. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security . Association for Computing Machinery, New York, NY, USA, 1962–1976. https://doi.org/10.1145/3243734.3243824

[5] Florian Platzer, Marcel Schäfer, and Martin Steinebach. 2020. Critical traffic analysis on the tor network. In Proceedings of the 15th International Conference on Availability, Reliability and Security (ARES). Association for Computing Machinery, New York, NY, USA, Article 77, 1–10. https://doi.org/10.1145/3407023.3409180

[6] Joshi, P.S., Dinesha, H.A. (2023). Study Report of Tor Antiforensic Techniques. In: Kumar, A.,Ghinea, G., Merugu, S. (eds) Proceedings of the 2nd International Conference on Cognitive and Intelligent Computing. ICCIC 2022. Cognitive Science and Technology. Springer, Singapore. https://doi.org/10.1007/978-981-99-2742-5_9

[7] Massimo Bernaschi, Alessandro Celestini, Stefano Guarino, and Flavio Lombardi. 2017. Exploring and Analyzing the Tor Hidden Services Graph. ACM Trans. Web 11, 4, Article 24 (November 2017), 26 pages. https://doi.org/10.1145/3008662

[8] Alex Biryukov, Ivan Pustogarov, Fabrice Thill, and Ralf-Philipp Weinmann. 2014. Content and Popularity Analysis of Tor Hidden Services. In Proceedings of the 2014 IEEE 34th International Conference on Distributed Computing Systems Workshops (ICDCSW). IEEE Computer Society, USA, 188–193. https://doi.org/10.1109/ICDCSW.2014.20

[9] Gareth Owen and Nick Savage. 2016. Empirical analysis of Tor Hidden Services. IET Information Security 10, 3 (May 2016), 113–118. https://doi.org/10.1049/iet-ifs.2015.0121

[10] Martin Steinebach, Marcel Schäfer, Alexander Karakuz, Katharina Brandl, and York Yannikos. 2019. Detection and Analysis of Tor Onion Services. In Proceedings of the 14th International Conference on Availability, Reliability and Security (ARES). Association for Computing Machinery, New York, NY, USA, Article 66, 1–10. https://doi.org/10.1145/3339252.3341486

[11] T. Zulkarnine, R. Frank, B. Monk, J. Mitchell and G. Davies, "Surfacing collaborated networks in dark web to find illicit and criminal content"; 2016 IEEE Conference on Intelligence and Security Informatics (ISI), Tucson, AZ, USA, 2016, pp. 109-114, doi:10.1109/ISI.2016.7745452.

[12] Alkhatib, Bassel &; Basheer, Randa. (2019). Crawling the Dark Web: A Conceptual Perspective, Challenges and Implementation. Journal of Digital Information Management. 17. 51. 10.6025/jdim/2019/17/2/51-60.

[13] York Yannikos, Julian Heeger, and Maria Brockmeyer. 2019. An Analysis Framework for Product Prices and Supplies in Darknet Marketplaces. In Proceedings of the 14th International Conference on Availability, Reliability and Security (ARES). Association for Computing Machinery, New York, NY, USA, Article 50, 1–7. https://doi.org/10.1145/3339252.3341485

[14] Mhd Wesam Al-Nabki, Eduardo Fidalgo, Enrique Alegre, Laura Fernández-Robles, ToRank: Identifying the most influential suspicious domains in the Tor network, Expert Systems with Applications, Volume 123, 2019, Pages 212-226, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2019.01.029.

[15] Iskander Sanchez-Rola, Davide Balzarotti, and Igor Santos. 2017. The Onions Have Eyes: A Comprehensive Structure and Privacy Analysis of Tor Hidden Services. In Proceedings of the 26th International Conference on the World Wide Web. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 1251–1260. https://doi.org/10.1145/3038912.3052657

[16] Mhd Wesam Al Nabki, Eduardo Fidalgo, Enrique Alegre, and Ivan de Paz. 2017. Classifying Illegal Activities on Tor Network Based on Web Textual Contents. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, pages 35–43, Valencia, Spain. Association for Computational Linguistics.

[17] S. M. M. Monterrubio, J. E. A. Naranjo, L. I. B. López and Á. L. V. Caraguay; “Black Widow Crawler for TOR network to search for criminal patterns”; 2021 Second International Conference on Information Systems and Software Technologies (ICI2ST), Quito, Ecuador, 2021, pp. 108-113, doi: 10.1109/ICI2ST51859.2021.00023.

[18] Qian Li, Hao Peng, Jianxin Li, Congying Xia, Renyu Yang, Lichao Sun, Philip S. Yu, and Lifang He. 2022. A Survey on Text Classification: From Traditional to Deep Learning. ACM Trans. Intell. Syst. Technol. 13, 2, Article 31 (April 2022), 41 pages. https://doi.org/10.1145/3495162

[19] Liu, B., Zhou, Y. & Sun, W. Character-level text classification via convolutional neural network and gated recurrent unit. Int. J. Mach. Learn. &; Cyber. 11, 1939–1949 (2020). https://doi.org/10.1007/s13042-020-01084-9

[20] Yoon Kim, Yacine Jernite, David Sontag, and Alexander M. Rush. 2016. Character-aware neural language models. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence. AAAI Press, 2741–2749.

[21] Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

[22] G. Arevian,; “Recurrent Neural Networks for Robust Real-World Text Classification”, IEEE/WIC/ACM International Conference on Web Intelligence, Fremont, CA, USA, 2007, pp. 326-329, doi: 10.1109/WI.2007.126.

[23] Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. 2021. Deep Learning--based Text Classification: A Comprehensive Review. ACM Comput. Surv. 54, 3, Article 62 (April 2022), 40 pages. https://doi.org/10.1145/3439726

[24] Pengfei Liu, Xipeng Qiu, Xinchi Chen, Shiyu Wu, and Xuanjing Huang. 2015. Multi-Timescale Long Short-Term Memory Neural Network for Modeling Sentences and Documents. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 2326–2335, Lisbon, Portugal. Association for Computational Linguistics.

[25] Peng Zhou, Zhenyu Qi, Suncong Zheng, Jiaming Xu, Hongyun Bao, and Bo Xu. 2016. Text Classification Improved by Integrating Bidirectional LSTM with Two-dimensional Max Pooling. In Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 3485–3495, Osaka, Japan. The COLING 2016 Organizing Committee.

[26] Graves and J. Schmidhuber, “Framewise phoneme classification with bidirectional LSTM networks”; Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005., Montreal, QC, Canada, 2005, pp. 2047-2052 vol. 4, doi: 10.1109/IJCNN.2005.1556215.

[27] H. Hu, M. Liao, C. Zhang and Y. Jing; “Text classification based recurrent neural network”, 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, China, 2020, pp. 652-655, doi: 10.1109/ITOEC49072.2020.9141747.

[28] M. Shobana, V. R. Balasraswathi, R. Radhika, Ahmed Kareem Oleiwi, Sushovan Chaudhury, Ajay S. Ladkat, Mohd Naved, Abdul Wahab Rahmani, "Classification and Detection of Mesothelioma Cancer Using Feature Selection-Enabled Machine Learning Technique", BioMed Research International, vol. 2022, Article ID 9900668, 6 pages, 2022. https://doi.org/10.1155/2022/9900668

[29] Ajay S. Ladkat, Sunil L. Bangare, Vishal Jagota, Sumaya Sanober, Shehab Mohamed Beram, Kantilal Rane, Bhupesh Kumar Singh, "Deep Neural Network-Based Novel Mathematical Model for 3D Brain Tumor Segmentation", Computational Intelligence and Neuroscience, vol. 2022, Article ID 4271711, 8 pages, 2022. https://doi.org/10.1155/2022/4271711

[30] Sunil L. Bangare, "Classification of optimal brain tissue using dynamic region growing and fuzzy min-max neural network in brain magnetic resonance images", Neuroscience Informatics,Volume 2, Issue 3,2022, 100019, ISSN 2772-5286, https://doi.org/10.1016/j.neuri.2021.100019.

[31] S.L. Bangare, G. Pradeepini, S.T. Patil, “Regenerative pixel mode and tumor locus algorithm development for brain tumor analysis: a new computational technique for precise medical imaging”, International Journal of Biomedical Engineering and Technology 27.1-2 (2018): 76-85. https://doi.org/10.1504/IJBET.2018.093087