Using NLP and Machine Learning to Detect Data Privacy Violations

Paulo Silva, Carolina Gonçalves, Carolina Godinho, Nuno Antunes, Marilia Curado
CISUC, Department of Informatics Engineering
University of Coimbra
Coimbra, Portugal
pmgsilva@dei.uc.pt, [mariapg,godinho]@student.dei.uc.pt, [nmsa,marilia]@dei.uc.pt

***Abstract*—Privacy concerns are constantly increasing in different sectors. Regulations such as the EU's General Data Protection Regulation (GDPR) are pressuring organizations to handle the individual's data with reinforced caution. As information systems deal with increasingly large amounts of personal data in essential services, there is a lack of mechanisms to help organizations in protecting the involved data subjects.**

In this paper, we propose and evaluate the use of Named Entity Recognition as a way to identify, monitor and validate Personally Identifiable Information. In our experiments, we used three of the most well-known Natural Language Processing tools (NLTK, Stanford CoreNLP, and spaCy). First, we assess the effectiveness of the tools with a generic dataset. Then, machine learning models are trained and evaluated with datasets built on data that contain personally identifiable information.

The results show that models' performance was highly positive in accurately classifying both generic and more context-specific data. We observe the relationship between the datasets' training size and respective performance and estimate the appropriate size for model training within this context. Furthermore, we discuss how our proposal can effectively act as a Privacy Enhancing Technology as well as the potential risks and associated impacts.

***Index Terms*—Privacy Violations, Machine Learning, Natural Language Processing, Personally Identifiable Information**

I. Introduction

Privacy concerns are becoming more evident in most various sectors. Recent data breaches and privacy scandals have likely triggered discussion, more specific policy-making and further research within the area. To comply with regulations and efficiently increase privacy assurances, it is necessary to develop mechanisms that can not only provide such privacy assurances but also with increased automation and reliability. In the scope of data privacy, the automated monitoring of *Personally Identifiable Information* (PII) can effectively be measured in terms of reliability while being properly automated at the same time.

Certain public or private organizations are legally bound to release contractual information, to publish anonymized data or to store sensitive data. Such data may contain sensitive information (e.g., persons' names, addresses, identification details, financial or employment information) about not only organizations but individuals as well. The data should be autonomously and effectively monitored not only due to its nature but also its size when considering big data. As the data can be released in an unstructured format (i.e., text), the usage of *Machine Learning* (ML) naturally qualifies for this task. Within the ML domain, *Natural Language Processing* (NLP) and *Named Entity Recognition* (NER) allow for a transparent monitoring and detection of PII, thus uncovering potential privacy violations.

In this work, we start by analyzing three NLP tools regarding their characteristics and capabilities. Further on, we used a generic publicly available dataset to assess the performance of the NLP tools. It was possible to observe that the $F_1 score$ of our models was approximately 90% in the best cases. Then, we searched for data-sets that contained any type of publicly available PII like names, addresses, contract numbers or other related types (e.g., publicly released contracts). Moreover, we manually tagged the entities to train the models. We discovered that in the best cases, the $F_1 scores$ were equally high, with approximately 90% score. Finally, we assessed the generalization capabilities of the models. Following, a list of our main contributions:

1) Evaluation of NLP tools' performance with general-purpose and multi-dimension data-sets;
2) Manual labeling (gold-standard) of publicly available datasets with entities such as names, address, employment, organizations, and others;
3) Analysis of NLP tools' performance on correctly retrieving entities classified as PII on publicly available data;
4) Presentation of proof of concept NER models for PII monitoring and respective discussion of its applicability as a Privacy Enhancing Technology.

The rest of this paper is organized as follows. Section II provides the necessary background regarding NLP and the Machine Learning techniques used in NLP. Section III presents related work in the field. Section V explains the methodology followed, provides a comparison of the features and characteristics of the three NLP tools used, the datasets used and metrics. Section VI presents the experimental results obtained. Section VII discusses the lessons learned and the applicability of our proposal. Section VIII concludes the paper and highlights the main findings.

II. Background

NLP is a branch of *Artificial Intelligence* (AI) and ML that helps computers understand, interpret and manipulate human language. NLP pipelines usually start by performing text or speech recognition and speech-to-text, depending on the application. Then, it continues by dividing the text into tokens. These tokens can be words, punctuation or numbers. The various NLP tools use different techniques for associating meaning to each token or combination of tokens. Nonetheless, the underlying process is similar for all.

NER, one of NLP's sub-tasks, seeks to find and classify named entities present in a text into specific and pre-defined categories [1]. Those categories can be people's names, addresses, states, countries, money, organizations, laws or any other kind of PII. With NER it is possible to automatically scan text documents, data structures (or any other text file container) and understand the importance of those entities in the context of the text. Performing NER with different NLP tools may lead to different NER performances due to its internal mechanisms. Also, a NER system designed within a tool for one project may execute differently in another project or not do the task at all [2].

Several applications and usage of machine learning are based on supervised learning [3], as is NLP. In the experimental part of this paper, different NLP tools are used and supervised learning is used to train machine learning models in all the tools. In the case of *Natural Language Toolkit* (NLTK), a Naive Bayes classifier [4] and *Hidden Markov Models* (HMM) [5] are applied. For Stanford CoreNLP, *Conditional Random Fields* (CRFs) are applied. CRFs are probabilistic models that perform segmentation and labeling of sequential data [5], which is the case of text used in NLP tasks. Finally, spaCy uses *Convolutional Neural Networks* (CNNs) [6] with pre-trained word vectors [7] to train its models.

III. Related Work

It is possible to find in the literature extensive work and publications regarding NLP, its characteristics, and its performance. For instance, Omran and Treude [8] perform a systematic literature review on how to choose an NLP library. The most commonly mentioned NLP tools are the NLTK, Stanford CoreNLP and spaCy.

The NLP's sub-task that is more suitable for the type of analysis we refer to is NER as this method uses models to classify the entities (e.g., Persons or Locations) it finds in the input text. Jiang et al. [9] reviewed tools to assess which ones are more accurate in NER. Of course, this can be applied in a wide variety of fields. For instance, Ritter et al. [10] used NER to recognize Named Entities in tweets. On a different application, Vlachos [11] evaluated NER systems for biomedical data.

Despite the availability of comprehensive NLP research in the literature, there is insufficient work relating to NER, PII, its implications, and possible use cases. There are links with clinical or biomedical data but not in the broad spectrum of PII, which encompasses many different kinds of personal information. We argue that using NLP and NER models can be a very adequate Privacy Enhancing Technology when applied in privacy-preserving data analysis (e.g., active or passive monitoring of text for compliance verification) as this avoids the usage of dictionary approaches and the involvement of human operators. Supporting our claims are the results of the experimental work we conducted with different NLP tools.

IV. NLP Tools

NLTK [12] is one of the most well-known NLP tools. It is community-driven and open-source Python software, which allows the manipulation of different corpora, categorizing text or analyzing linguistic structure (e.g., tokenization, *Part of Speech* (POS) tagging or NER). Its development and open source contributions give it a large market adoption for NLP starters and many other simple other activities. Although it could have the potential to perform better in production environments (with further improvements and development), it usually serves as a base tool for several academic courses and NLP teaching activities.

The Stanford CoreNLP [13] tool stands out as a reference tool in the field of NLP. The tool is open-source, developed in Java and, among other features, it is capable of performing sentiment analysis, dependency parsing, or NER, for instance. When compared with NLTK, the Stanford CoreNLP offers additional standard features out of the box (e.g., dependency parsing). Similar to NLTK, continuous development allows it to perform better at each new software release - surpassing NLTK. It is not only used for academic purposes or NLP introduction, but it is also been more referenced in more production environments.

In 2016, ExplosionAI introduced spaCy [14] as the fastest NLP library in the world. The fact is that it is not only fast, but also performs well against similar tools and supports similar features. Furthermore, as an advantage, spaCy has both *Neural Networks* (NNs) models and Integrated Word Vectors. Also, with their new tokenization algorithm a better balance between performance, ease of definition, and ease of alignment into the original string are ensured.

V. Experimental Approach and Data

It was necessary to devise a methodology to determine to which extent NLP and NER can effectively be used to reliably detect and identify PII, and ultimately used as *Privacy Enhancing Technologies* (PETs). For that purpose, we used different NLP tools and trained (and tested) NER models with different data. This allows us to evaluate not only the models we train but also the NLP tools used: NLTK, Stanford CoreNLP, and spaCy.

The approach followed is divided into three parts: the first with generic data, the second with publicly available contracts containing PII, and the third with mixed datasets. The three parts involve the training and validation of machine learning models. The following sections provide the necessary information for each part.

A. Generic Data

We started by gathering generic data that was already tagged with named entities. For this purpose, we used Kaggle [15] - an online community for sharing and improving datasets.

The datasets used in the experiments were based on the Groningen Meaning Bank data [16], which are composed or public domain English text. The datasets contain, for instance, news, reports and other public domain publications.

To better evaluate the performance of the tools and respective models we partitioned the dataset in smaller chunks. The objective was to assess how the performance of the models was affected by the datasets' size (e.g., number of tokens or sentences). Also, this provided an estimate of the amount of data (i.e., sentences or tokens) that are necessary to train models and achieve desirable results.

The dataset had 1.354.149 tokens. Therefore, to evaluate performance of the NLP tools, as well as the models' classification capabilities using data with different sizes, the dataset was sliced in smaller portions (5%, 10%, 20%, 30%, 40% , 50%, 60%, 70%, 80%, 90% and 100%). For each portion was then applied the 70% and 30% proportion rule for training and validation, respectively. In the second stage, the dataset contained 19.838 tokens (equivalent to 1150 sentences). Thus, no size reduction was performed. The training and validation proportions of 70%-30% were kept unchanged.

B. Publicly available data with PII

At this point, the procedure was identical to the first. The only difference was the data, which focused on publicly available data containing PII. The dataset created was a fusion between contracts available in online sites [17], [18] and other contracts from the U.S *Department of Defense* (DoD) [19].

After retrieving publicly available contracts in PDF format, it was necessary to extract the information and convert it to text files using the PyPDF2 Python library. Only then, it was possible to perform the necessary tokenization and proceed with the manual tagging of entities.

The focus was on different kinds of entities, namely the ones described next (which are at the same time PII). For that, it was necessary to search legal and publicly available datasets containing these kinds of entities as much as possible. Finally, it was possible to find datasets containing such information. One kind of source was the U.S. DoD, where they publish the daily expenses of the military branches [19]; the other sources are publicly available contracts found online [17], [18]. It is possible to observe that from the total list of entities we have defined, 68% of them were labeled during the annotation process.

C. Mixed datasets

To address the generalization capabilities of our models we merged different datasets for training. Additionally, we used *United States* (US) voters' registration data [20] to increase the diversity of the dataset.

We generated a mixed dataset that was composed of a combination of 120K lines of the Voters dataset and 50K lines of the Kaggle dataset. Moreover, we used different validations files. For instance, training models with generic datasets (e.g., Kaggle) validating with context-specific datasets (e.g., U.S DoD contracts or US voters' registration data).

D. Model Training and Evaluation

For all datasets, it was necessary to divide the original dataset into two. The first part for training (70%) and the second part for validation (30%). After performing the preprocessing of the datasets, it was possible to proceed to the model training and evaluation of each tool. Training a model requires quality data in sufficient amounts. With the manual-labeling (also known as gold-labeling) complete, it is possible to process with the model training. For each tool, the default settings were kept unchanged.

After training the model, it is necessary to evaluate the models' performance. This is, assess how well it predicts entities using datasets that it has never processed before. For that, we provide each model with validation datasets which are equally labeled, as the training datasets. Each model then classifies the entities in the validation dataset and then compares the results to the actual tag that corresponds to each entity.

E. Metrics

Following the previous procedures, after the model evaluation is completed, it is possible to determine the precision, recall, accuracy, and $F_1 scores$ of each NER model. This provides the necessary information to analyze and discuss not only the models' performance but the respective capability of identifying PII.

To analyze and evaluate the performance of the NLP tools and respective NER system, it is necessary to identify the number of True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN) derived from the model classification. Such values allow the computation of the Precision and Recall of the models and ultimately the $F_1 score$ (a harmonic mean of the two).

The time spent on training each model, as well as the number of iterations (also known as epochs) were also registered. With these values, it is also possible to assess the models' training performance in terms of speed and optimization capabilities.

VI. Experimental Results

The following section describes the results obtained in the different parts of the experimental work.

A. Generic Data

Figure 1 shows the results of the three NLP tools with generic data partitioned in different sizes. NLTK obtained a $F_1 score$ of 0.47 using the smallest portion of the dataset (2.5%). Stanford CoreNLP and spaCy reached approximately 0.65. The entire dataset (100%) achieved the best results: NLTK achieved approximately 0.67, while Stanford CoreNLP and spaCy obtained 0.84 and 0.86, respectively. There was no significant difference between the 20%-sized dataset and the larger ones that follow. The $F_1 score$ difference between the 20%-sized dataset and the full dataset is between 0.03 and 0.05, between Stanford CoreNLP and spaCy.

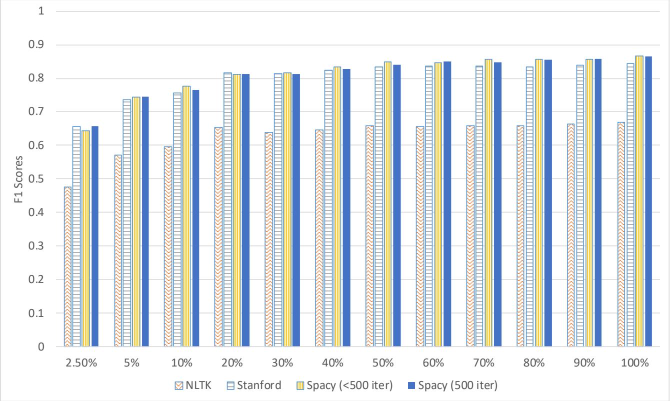

Fig. 1. F_1 Scores (NLTK, Stanford CoreNLP and spaCy)

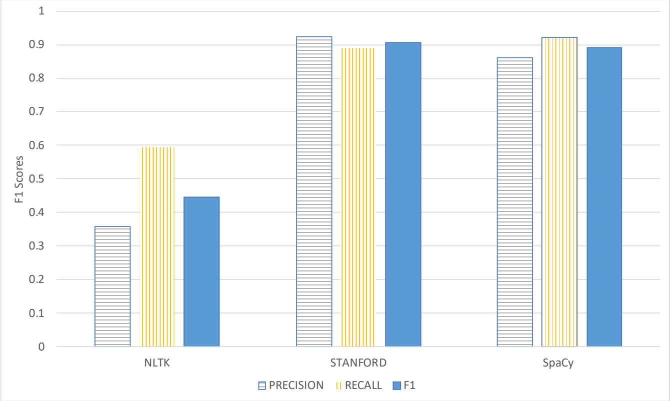

Fig. 2. F_1 Scores, Precision and Recall values (NLTK, Stanford CoreNLP and spaCy)

In our experiments, the worst performance was NLTK. On the other hand, although Stanford CoreNLP and spaCy achieved similar results, the best performance was achieved with spaCy, although with a small margin. The results indicate that without any comprehensive tuning of the model training settings, spaCy provides the best results for F1-Score. Additionally, we observed than training the models for less than 500 iterations provides similar results and requires much less training time.

Regarding training time, NLTK was the fastest. The elapsed time was 2 seconds for the smallest dataset and 75 seconds for the largest dataset. Stanford CoreNLP takes approximately 10 minutes to train the smallest dataset. On the other hand, it takes approximately 120 minutes to train the largest dataset. The training time with spaCy differs according to the number of training iterations. Setting a maximum of 500 iterations, the training time for the largest dataset was 6000 minutes. However, when the number of iterations is reduced to approximately half, the same dataset size takes only 2000 minutes and the $F_1 scores$ are very similar. Therefore, it is possible to achieve good results while spending less time training. According to spaCy's documentation, developers should experiment with different parameters and fine-tune the model in order to provide the best results.

B. *Publicly Available Data with PII*

Figure 2 shows the precision, recall, and $F_1 scores$ obtained while evaluating the models that were created from manually-labeled contracts. It was possible to observe that NLTK's best results were approximately 0.45. On the other hand, Stanford CoreNLP and spaCy reached very similar values (approximately 0.90). Moreover, the difference between these two is 0.01, being Stanford CoreNLP the one with better results in this case.

Regarding the model-training time, the results have shown that it is not necessary to spend a great amount of time (or training iterations) to devise a system that is able to correctly retrieve PII-related entities. The longest session takes approximately 6500 seconds in spaCy, while Stanford CoreNLP takes approximately 1125 seconds and performs similarly in terms of $F_1 score$.

The less positive aspect is the time necessary for manually hand-labeling the data fed to the models. Each document took an average of 4.75 hours of work. The measure indicates the time spent by one person labeling the entities in the referred datasets. Afterward, each document was reviewed by at least one other person for consistency purposes.

It becomes evident that the models perform similarly regardless of having generic or PII-specific content for training purposes. By knowing the behavior of the machine learning algorithms behind such systems, one could say that this should be the expected outcome. However, the size of the sample may not fully define the model's behavior with other kinds of data. Nevertheless, we see that with approximately 20 hours of manual labeling, it is possible to create a model able to identify entities such as person, city, title, employment details, and others.

C. *Mixed Datasets*

Since NLTK and Stanford CoreNLP do not support re-training, spaCy was the only tool used in this scenario. Figure 3 shows the results obtained while evaluating models using datasets from different domains, but also re-training existent models.

In the first case, the model was evaluated with the validation section of the 20% Kaggle dataset and the results were not good. On the other hand (second and third case), using a validation dataset that resembles more to the re-training data shows better results. Therefore, the results of the re-trained the models (first three cases) suggest that the models tend to forget previous information and retain more recent data.

The last three cases depicted in Figure 3 indicate the generalization capabilities of spaCy models are low.

VII. Discussion

Considering all the collected results, it is now possible to further discuss the lessons learned throughout the experimental work. Therefore, the following section highlights the

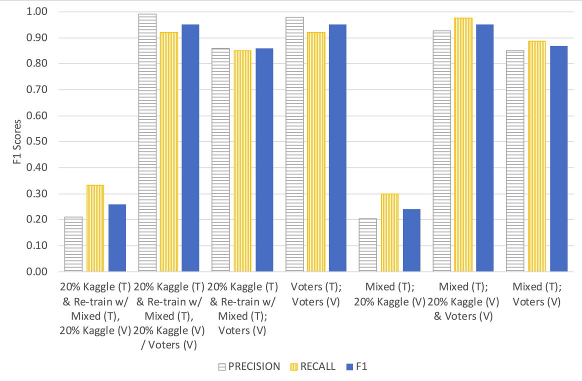

Fig. 3. F_1 Scores, Precision and Recall values of the training and re-training sessions with spaCy

main conclusions of this work, the applicability and utility of our proposal as a *Privacy Enhancing Technology* (PET), and finally, the potential risks and impacts associated with performing data analysis with such mechanisms.

A. Lessons Learned

Performing each step of our approach allowed us to derive several conclusions about our work. Not only regarding NLP, NER or ML but mainly how the combination of such technologies can be used for the sake of privacy protection. As such, we highlight the main lessons and findings of this work:

1) Dataset Size and Classification Accuracy: During the model training with generic data, there was a much larger sample available. There was a total of 47959 sentences (the same as 1.354.149 tokens) readily available and labeled. Dividing the dataset in smaller chunks allowed us to also verify when (i.e., dataset size) would the classification performance metric ($F_1 score$) stabilize.

We concluded that 20% of the total dataset size was already sufficient to provide results that are very similar to the ones obtained using the full-sized dataset. The proportion of 20% is equivalent to 9590 total sentences, and the $F_1 score$ variation is 0.01, 0.03, and 0.05 for NLTK, Stanford CoreNLP, and spaCy, respectively.

2) Manual-labeling Effort: Manually labeling data is time-consuming. Nevertheless, it is essential unless there are other sources of labeled data. However, since PII is a very sensitive type of information, it is not as readily available as other types of data. Therefore, it was possible to realize that it is not feasible (mainly for smaller teams) to manually label large amounts of data.

To counter this issue, we consider two possibilities. The first is to assess the feasibility and reliability of using a synthetic data generator [21]. The second possibility is to recur to online annotation services. However, the latter still depends on finding appropriate and sufficient datasets, which is hard. Nevertheless, future work is likely to pursue one of these two approaches.

3) Data Diversification: As previously stated, it is hard to retrieve quality data that matches our requirements (e.g., containing publicly available PII). It is natural since this is the type of information that should be kept out of the public domain or out the reach of any individual. Moreover, we noticed that the generalization capabilities of the models are not optimal.

At the same time, based on the entities we defined as PII, we used 68% of them during the labeling process. This means we are not fully reaching all of our system's potential in terms of types of personal data that are identified. To overcome this issue, we expect to employ one of the two possibilities mentioned before (i.e., synthetic data generator or online annotation services).

B. Applicability as a Privacy Enhancing Technology

The main objective of the work described in this document was to devise an approach that combines the aforementioned technologies and provides an effective Privacy Enhancing Technology. There are several use cases where our approach could be effectively enforced. Such use cases are described next.

1) Data Validation: General data validation is enforced in a variety of services and fields. From validation of text boxes in web pages to more complex processes that ensure the delivery of clean and validated data. With our approach, systems could be able to not only validate data types and formats but also the contents. Systems managing text data inputs (e.g., forms) would be able to distinguish if the inputs match the actual description. For instance, our system would be capable of generating a warning if the "Comments" fields (or any other insensitive field) would be filled with sensitive data (i.e., PII). This way avoiding the submission of sensitive information in unnecessary circumstances.

2) PII Discovery: The discovery of PII is closely linked to the previous point. This allows not only to perform data validation but also to discover previously unidentified PII. This kind of monitoring can be applied in several scenarios. For instance, transactions or information exchanges between systems and/or users, documents or databases. In fact, any other kind of big data processing task, always depending on the context and the privacy implications. This would allow the system to warn users (i.e., data owners) or systems administrators so they could take appropriate actions.

3) Permission Checking: Another advantage of our system is for permission checking purposes. In this case, permission-based systems would be able to map and verify if the actual data matches the textual description of the respective permissions granted. However, in this particular situation, it would be necessary to devise and implement an *Natural Language Understanding* (NLU) module for the extraction of the meaning of such permissions. In systems where there no such textual descriptions of the permissions, it is possible to directly map the permission type, to the PII type, thus allowing permission verification on a higher level.

4) Compliance and Transparency: On top of all the possible application scenarios, there is the need for compliance with privacy regulations (e.g., *General Data Privacy Regulation* (GDPR) or other privacy-related regulations). Therefore, the enforcement of our system is intended to be open and transparent. With open-source code, the system's design becomes available for peer review. Additionally, it shows how all the data is processed and then discarded every time it runs. This is because our system relies on ephemeral storage and it does not communicate with any other service for PII data exchange.

C. Potential Risks and Impacts

Even though there are many benefits and advantages associated with this concept, there still risks and less positive impacts. For instance, if these mechanisms are misused, they can lead to unlawfully searches in data repositories to look for PII. This disadvantage goes against the privacy rights and assurances that we are striving to protect.

Another problem faced in the experimental work was the difficulty in obtaining reliable data for the model training phase. It was quite hard due to the lack of publicly available PII. Consequently, we did not possess the desirable amount of data for the third part of our work. Additionally, directly related to this issue is the fact that manually labeling data is very expensive. It demands several resources and a lot of time.

VIII. Conclusion

In this work, we discussed how our proposed usage of NLP and ML can be used to detect data privacy violations. In the process, we evaluated the effectiveness of three NLP tools and their NER sub-tasks in discovering PII. Ultimately, we discussed the applicability as a PET.

We developed an experimental setup where different ML models were trained and evaluated with generic as well as context-specific datasets. We also show that related work usually focuses on specific areas such as clinical or biomedical data and not PII in its broader definition. Whereas our method can equally include such data. The positive results of our proposal are verified by two main NLP tools (Stanford CoreNLP and spaCy).

Acknowledgment

The work presented in this paper was partially carried out in the scope of the PoSeID-on project - Protection and control of Secured Information by means of a privacy enhanced Dashboard, Grant Agreement Number: 786713. H2020-DS-2016-2017/ DS-08-2017.

References

[1] L. J., S. A., H. J., , and L. C., "A survey on recent advances in named entity recognition from deep learning models," in *Proceedings of the 27th International Conference on Computational Linguistics*. Santa Fe, New Mexico, USA: Association for Computational Linguistics, Aug. 2018, pp. 2145–2158.

[2] R. L. and R. D., "Design challenges and misconceptions in named entity recognition," in *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, ser. CoNLL '09. Stroudsburg, PA, USA: Association for Computational Linguistics, 2009, pp. 147–155. [Online]. Available: http://dl.acm.org/citation.cfm?id=1596374.1596399

[3] A. T., *Types of Machine Learning Algorithms*. Rijeka, Croatia: InTech, 02 2010, ch. 3, p. 366.

[4] C. J., H. H., T. S., and Q. Y., "Feature selection for text classification with naïve bayes," *Expert Systems with Applications*, vol. 36, no. 3, pp. 5432–5435, 2009.

[5] L. J., M. A., and P. F., "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of the Eighteenth International Conference on Machine Learning*, ser. ICML '01. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001, pp. 282–289. [Online]. Available: http://dl.acm.org/citation.cfm?id=645530.655813

[6] Z. L. and S. P., "A survey of randomized algorithms for training neural networks," *Information Sciences*, vol. 364, pp. 146–155, 2016.

[7] P. J., S. R., and M. C., "Glove: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1532–1543.

[8] O. F. and T. C., "Choosing an nlp library for analyzing software documentation: A systematic literature review and a series of experiments," in *Proceedings of the 14th International Conference on Mining Software Repositories*, ser. MSR '17. Piscataway, NJ, USA: IEEE Press, 2017, pp. 187–197. [Online]. Available: https://doi.org/10.1109/MSR.2017.42

[9] J. R., B. R., and L. H., "Evaluating and combining name entity recognition systems," in *Proceedings of the Sixth Named Entity Workshop*. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 21–27.

[10] R. A., C. S, Mausam, and E. O, "Named entity recognition in tweets: An experimental study," in *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*. Edinburgh, Scotland, UK.: Association for Computational Linguistics, Jul. 2011, pp. 1524–1534.

[11] V. A., "Evaluating and combining and biomedical named entity recognition systems," in *Biological, translational, and clinical language processing*. Prague, Czech Republic: Association for Computational Linguistics, Jun. 2007, pp. 199–200.

[12] B. S., K. E., and L. E., *Natural Language Processing with Python*. Boston, USA: O'Reilly Media, 01 2009.

[13] M. C., S. M., B. J., F. J., B. S., and M. D., "The Stanford CoreNLP natural language processing toolkit," in *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. Baltimore, Maryland: Association for Computational Linguistics, Jun. 2014, pp. 55–60.

[14] ExplosionAI. (2019) spacy - industrial-strength natural language processing. [Online]. Available: https://spacy.io

[15] Kaggle. (2019) An online community of data scientists and machine learners. [Online]. Available: https://www.kaggle.com

[16] U. of Groningen. (2019) Groningen meaning bank. [Online]. Available: https://gmb.let.rug.nl

[17] T. D. of Information Resources. (2019) Our mission is to provide technology leadership, technology solutions. [Online]. Available: https://dir.texas.gov/View-Search/Contracts-Detail.aspx?contractnumber=DIR-TSO-4101

[18] Metrolink. (2019) Metrolink is southern california's premier regional passenger rail system serving over 55 stations across the region. [Online]. Available: https://www.metrolinktrains.com/globalassets/about/contracts/may-26-2019/contract-no.-sp452-16-conformed-contract-fully-executed.pdf

[19] U. D. O. Defense. (2019) Official website for u.s. department of defense. [Online]. Available: https://www.defense.gov/Newsroom/Contracts

[20] N. C. S. B. of Elections. (2020) Election results data. [Online]. Available: https://www.ncsbe.gov/Public-Records-Data-Info/Election-Results-Data

[21] MostlyAI. (2019) Creating ai-generated synthetic data. [Online]. Available: https://mostly.ai