Personal Identifiable Information (PII) Detection Utilizing Named Entity Recognition (NER) with Naïve Bayes and Logistic Regression Algorithms

Muhammad Sohel - msohel@students.kennesaw.edu

Kennesaw State University. CS7347. Fall 2023.

Abstract:

Personal Identifiable Information, also known as PII is the information consumers usually provide to businesses or agencies when getting a service. This information is usually stored by companies to identify what service the individual had received. This can relate to almost any industry or sector. Whether it be a patient's medical lab reports, a defendants' legal paperwork, or a customers' e-commerce order, all of them contain some sort of PII that links back to the individual. Many laws and regulations worldwide require proper handling of PII, due to the major security concern that can arise from misuse or mishandling of this sensitive information. This data must be identified and properly handled to ensure PII is safe and secure. Natural Language Processing (NLP) plays a big role in identifying PII within different styles of documents. Being able to identify PII quickly and efficiently through NLP relives the strain of many compliance departments in their handling process. Using NLP techniques such as NER, TF-IDF, or ML algorithms such as Naïve Bayes or Logistic Regression for developing a model that can lead to a solution which can accurately and efficiently detect PII within different documents provides a beneficial tool to many, and ultimately protects an individuals' privacy.

Keywords: Personal identifiable information, PII, Natural language processing, NLP, documents, detection. naïve bayes, logistic regression

I. INTRODUCTION

The importance of detecting PII information comes from both legal and ethical grounds. Ensuring that this information is properly, accurately, and efficiently detected for safe handling would come at the utmost importance for many legal compliance departments across almost every sector. This data is usually collected and held in the masses for large companies, having an efficient method to collect PII would almost become necessary. PII itself includes types of basic information such as name, address, date of birth (DOB), social security number (SSN), credit card information, IP address, and MAC address. Some laws even further classify PII into multiple categories, for example the U.S. Department of Homeland Security divides this type of information into PII and Sensitive Personally Identifiable Information (SPII) in its "Handbook for Safeguarding Sensitive PII" (DHS, 2017). Overall, detecting any category of PII is crucial to say the least. This brings the motivation to efficiently protect this type of data, regardless of what sector the data is collected in.

Using natural language processing techniques can be beneficial in identifying the different specific types of PII within different style documents by creating a system that can competently recognize and detect such information. The use of NLP techniques and tools such as Names Entity Recognition (NER) are useful in tasks such as PII detection. NER is conditions on identifying and classifying entities within text. Entities are classified into different categories such as ('PERSON') for an individual, ('ORG') for organizations, and ('LOCATION') for a physical location. Term Frequency-Inverse Document Frequency (TF-IDF) is a well-known extraction technique that uses numerical statistics of a term within a document and compares to its prevalence across an entire corpus of text. It assigns weights to terms based on their frequency within the corpus, creating a representation of textual data for machine learning programs. Well known and mostly used in document categorization and information retrieval, it provides a solid way of determining the importance of words within a document in a numerical manner. Machine Learning (ML) algorithms are very popular in modern artificial intelligence, showing how systems learn from data and produce predictions and or decisions based on the given data. These algorithms recognize different patterns and improve themselves iteratively as more data and patterns are fed to it program. Popular algorithms such as probabilistic or generative types are used to address specific challenges based upon their strengths and provide valuable, reinforced data driven results. Having an efficient ML algorithm that caters to the needs of the task at hand is key to get streamlined and productive ML model to solve a given task.

The overall goal of this project is to aid in the detection and extraction of PII within different documents to get a solid baseline and understanding within NLP and ML topics. The

specific goals of this project include development of an NLP system using NLP techniques and following proper methodology such as data collection, data processing, feature extractions, model selection, testing, and training.

II. Related work

The need for PII detection is widely known due to its high demand, gaining attention from almost every field of study. While just a few studies look into the idea of creating a system to detect PII using NLP techniques, there are many studies that use NLP to help in the detection of other types of entities. In the study by (Kulkarni, 2021) PII was detected from large unstructured text corpus. Python programming was used to design and implement a system to achieve its goal of creating a model. The model was broken into 5 parts, which include dataset, data visualization, preprocessing, modeling using NLP, and implementation. In (Hathurusinghe et al., 2021) they were able to extract PII from an unstructured text. Their unstructured data set text was biographies that were extracted from Wikipedia pages. Once extracted, the dataset was then tagged using BIO scheme and put through additional NLP techniques to eventually get an averaged F-score to evaluate the results. Another similar study by (Silva et al., 2020) focused more on detecting privacy violations within contracts, but successfully extracted PII data using different model. These models included NLTK, Stanford CoreNLP, and spaCy. These models were all evaluated using F1 scores, precision, recalls, true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN). Overall, successfully "evaluated the effectiveness of three different NLP tools and their NER sub-tasks in discovering PII". (Silva et al., 2020, pg. 3) Giving us great insight on how to evaluate this proposed project. A Russian study by (Gultiaev & Domashova, 2022) goal was to develop 15 different NER models using machine learning (ML) methods to detect private data and compare each of these models to each other. The models underwent pre-processing, datasets were trained mostly by Python libraries, some were split into groups such as classical ML methods like naïve bayes and logistics regression, while others were neural networks. The results were evaluated using F1, precision, and recall metrics. In conclusion, the bidirectional LSTM neural network method as the most successful, the authors also state that this model took the longest to train as well, making it perform fairly well. In a medical study by (Ghiasvand & Kate, 2018) show the importance of using machine learning models that are trained on a corpus. They trained the model to determine negative and positive examples of medical terms and compared their results to an existing medical NER model. Their model performed better when comparing results, such as precision, F1, and recall. The study of (Hutchinson, 2020) goes into dept different literatures of NLP and ML archival processes, as well as the functional requirements needed in such a study. They determine that principles such as usability, interoperability, flexibility, and configurability is essential in creating a more commercialized tool. This style of implementation allows for models and tools to have a greater potential and "broader use by the archival community". (Baron and Borden, 2016, pg. 168). In (Wei et al., 2021), machine learning is used to detect PII. Their methodology involved first preprocessing of dataset, then segmentation into words or phrases and then converted into vectors in a vector space model. Techniques such as TF-IDF was used to evaluate importance of words, k-NN algorithm was used for text classification, and cross validation to find the best function. F1- measure was calculated for evaluation and conclusions. Similarly in the study of (Subramani et al., 2019) they distinguished different categories of PII, and assignsed different risk levels to them. Their goal was to test two models of NLP, both presdio and regex. Their data set included a compilation of different smaller datasets that include science, law, research papers, mathematics, books, subtitles, patents, and philosophy. They concluded that both models perform fairly well, with presido performing better with some categories of PII and regex performing better with others. A manual audit was done for validation of results, while accuracy and precision were checked for comparison. The study of (Carroll et al., 2012) is another medical study, specifically dealing with Electronic Health Records (EHR). The dataset comes from physician reviewed charts from three different universities. Their goal is to identify patients with rheumatoid arthritis (RA) using NLP processing systems. The use of an algorithm to help identify RA patients with these EHR. Different NLP techniques such as finding medications from outpatient was done using regular expressions. The algorithm was tested using a logistic regression model on the datasets to help determine RA with the patients' records. Cross validation was used as a measure of the algorithm's performance within the different university records. Overall results were analyzed by sensitivity and PPV.

In the study of (Kashti & Prasad, 2019), their goal was to detect fake reviews and usernames from online shopping platforms. Although this does not have to do with PII detection, their method of using NLP and text algorithms to detect information within text can also be brought over to PII detection. Their workflow consisted of data acquisition, data pre-processing, and classifier/algorithms to reach their result.

These different studies provided great input and analysis on the importance of PII and how to apply different NLP techniques to help aid in its detection. Popular techniques such as spaCy NER and NLTK were used, as well as different algorithms to create ML models. There prior studies placed the foundation and model to follow for this project to be successful in detecting PII using different NLP techniques.

III. Methodology

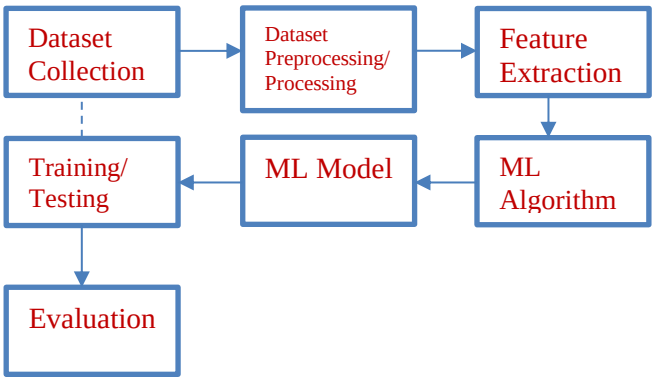

Figure: Methodology Workflow Diagram

Data Collection:

Just as the ultimate goal of this project is to protect PII information of individuals, obtaining a real PII dataset is not in the best legal or ethical interest. Protecting any real information of individuals is at the utmost of importance. Many previous researchers pull or create their own datasets from public information or obtain synthetic made "fake" datasets to fulfill a PII dataset for model testing. For this project, two datasets were obtained. The first is a dataset that contains fictious Electronic Medical Record (EMR) information with 10,000 records, it includes columns such as name, age, gender, insurance provider, test results, etc.…The second dataset is another synthetic EMR, with columns for names, ethnicities, marital status, addresses, etc…. Containing a total of 16,164 patient records. The first step in data processing will be to take the datasets and load them into the python code. This will be done using pandas library with its read csv file feature.

Data Processing:

Once the data set is now loaded in the code, it must be processed. Before the official processing occurs, a preprocessing step of removing columns with "white noise" such as age, room number, gender, or any single letter or single integer columns which don't necessarily apply to PII. For the first EMR dataset, the columns of Name, Medical Condition, Insurance Provider, Hospital, and Test Results. The second dataset used columns FIRST, LAST, RACE, BIRTHPLACE, and ADDRESS. These columns were selected respectively for both of the datasets because they contain a good mixture of both PII and Non-PII information to train and test the model.

Now for processing these dataset columns, the spaCy Named Entity Recognition (NER) feature will be used to detect and extract the 'PERSON' identity within the columns. In the case of EMR's, associating names with other confidential information such as test results or insurance provider would be considered PII, so detecting and extracting the names will be essential. Another point to consider is how to take note and specify which columns are PII and which are not, as well as labeling them in a manner so it can be identified later on. Labeling such as integers 1's and 0's for confirmation of PII can be a useful technique for later parts of the project such as vectorization.

Feature Extraction:

The goal of feature extraction is to turn data into numerical vectors though vectorization. To accomplish this, The NLP technique of TF-IDF (Term Frequency-Inverse Document Frequency) will be used. This allows for a better understanding of how important each word is and how it fits inside a dataset relative to frequency. The TF-IDF value is calculated by multiplying the term's frequency (TF) in the document by the logarithm of the inverse document frequency (IDF) of the term across the documents (Pradeep, 2023).

$$\mathbf{T\,F - IDF = F\,F * \log(\frac{N}{DF})}$$

TF-IDF equation (Welderufael, 2019)

The breakdown of the TF-IDF include N which indicates the number of documents, and DF is the number of documents that contain the feature, in the case of this project it would be PII.

The conversion of the text into numerical vectors and identifying the importance of words is important for the detection of PII because it helps create features and understandings that can be used for the ML model. This assigned numeral is also imperative for ML algorithms to process and the next step of this project.

ML Algorithm and Model:

Two of the most popular NLP algorithms that can efficiently run on datasets are Naïve Bayes and Logistic Regression. Naïve Bayes is a generative probabilistic classifier that uses linear time, meaning that it deals with probabilities of the data it is given. Naïve Bayes also assumes that features of the text are conditionally independent. Logistic regression is known to be a discriminative classifier that does not use probabilities to make a decision, but instead it uses and assigns weights to form vectors to eventually get to its decision. Logistic regression also does not make the assumption that the features of the text are conditionally independent.

Naïve Bayes was one of the chosen algorithms of this project because of its well-known efficiency and simplicity. Known as a lightweight and easy to understand algorithm. The specific type of Naïve Bayes that will be used is the Multinomial version because "This type of Naïve Bayes classifier assumes that the features are from multinomial distributions. This variant is useful when using discrete data, such as frequency counts," (IBM, 2023). The use of Multinomial Naïve Bayes is exactly what is needed to run the frequency counts of the text that were obtained through TF-IDF vectorization. Making it the most appropriate version and choice of Naïve Bayes for this project.

$$\mathbf{P(y|xi) = \frac{1}{1+e-y(wTxi+b}} \quad \textbf{for y} \in \{+1,-1\}$$

Naïve Bayes Equation (Weinberger, 2018)

Logistic Regression is the other classifier algorithm chosen for this project. It is also known as a very popular algorithm but applies a completely different approach to reach its target. It's usage of weights to predict can be very useful in the task of identifying whether a text is PII or not. It is known to be well suited and widely used for binary classification projects. This makes it a very solid and functional algorithm to choose for this project on determining whether an entity is PII or not (binary). The Logistic Regression equation is expressed as a sigmoid function, that transforms linear combinations of any feature into a probability value between 0 and 1, in the case of this project it would be PII or Non-PII.

$$\mathbf{P(y|xi) = \frac{1}{1+e-y(wTxi+b)}}$$

Logistic Regression Equation (Weinberger, 2018)

IV. Experimental Setup

Implementation details:

This project, ML algorithm, and model will be implemented on a Macbook pro 1.4 GHz Quad-Core Intel Core i5, with 8 GB memory. Python Version 3.9.7 will be the programming language used. Pycharm IDE will be the developing environment. Three different libraries will be imported and used for this project. The pandas library will be used for its ability to import and read csv files into the code. The spaCy library v1.0.1 will also be used for its NER entity to help detect our PII information within the data of the different datasets. The scikit-learn library will be the most used because it provides lots of resources that make this project possible. The ability to vectorize and transform our text using TF-IDF is given, along with the Naïve Bayes and Logistic Regression algorithms, which are both essential in creating our ML model. Scikit also provides the ability to split our text into training and testing section. Lastly, scikit provides the calculations needed for all of the evaluations metrics, such as accuracy, precision, recall, and F-1 scores.

Dataset description:

Two datasets were obtained from the Kaggle AI and ML community datasets. Terms such as “Names”, “Records”, “Medical” and “PII” were searched, and the following datasets were chosen as a diverse and efficient dataset. Both datasets are synthetic electronic medical records (EMRs).

The first dataset is a CSV file titled “Healthcare Dataset” and contains valuable information for this project. “Each column provides specific information about the patient, their admission, and the healthcare services provided, making this dataset suitable for various data analysis and modeling tasks in the healthcare domain” according to source website Kaggle. It contains a total of 15 columns and 10,000 total records.

Name	Medical Condition	Hospital	Insurance Provider	Medication	Test Results
Tiffany Ramirez	Diabetes	Wallace-Hamilton	Medicare	Aspirin	Inconclusive
Ruben Burns	Asthma	Burke, Griffin and Cooper	UnitedHealthcare	Lipitor	Normal
Chad Byrd	Obesity	Walton LLC	Medicare	Lipitor	Normal
Antonio Frederick	Asthma	Garcia Ltd	Medicare	Penicillin	Abnormal
Mrs. Brandy Flowers	Arthritis	Jones, Brown and Murray	UnitedHealthcare	Paracetamol	Normal
Patrick Parker	Arthritis	Boyd PLC	Aetna	Aspirin	Abnormal

Table: Healthcare Dataset Columns Sample (6)

The second dataset is titled “Payload-Normal” and contains 16,164 patient records (name, address, birthdate, gender, race, ethnicity etc.)

FIRST	LAST	RACE	BIRTHPLACE	ADDRESS
Dixie181	Welch248	white	Gloucester MA US	37829 Coty Pine Haverhill MA 01830 US
Chin491	Bergstrom451	white	Newburyport MA US	26507 Kshlerin Knoll Worcester MA 01607 US
Elbert922	Walsh907	hispanic	Boston MA US	22168 Schroeder Manor Everett MA 02149 US
Gale768	Sauer905	asian	Brookline MA US	67962 Cummerata Ramp Walpole MA 02081 US
Star826	Halvorson947	white	Worcester MA US	941 Erdman Falls Apt. 607 Worcester MA 01607 US
Kyoko951	D'Amore595	black	Boston MA US	92461 Adrain Ranch Boston MA 02119 US

Table: Payload-Normal Dataset Columns Sample (6):

Both datasets were processed and cleaned to get rid of white noise or irrelevant information. Columns with string values with “normal words” were selected, while columns with just integers or single letters were excluded. The two samples above and the data processing section both show which specific columns were selected for this project. These were selected because we can also confirm which columns have PII in them and which don’t. The goal was to train both naïve bayes and logistic regression to see how accurately it can determine that the name columns are PII, and the rest of the columns are non-PII. Based on this, we can confirm there are 10,000 PII names in the healthcare dataset and 16,164 PII names in the second dataset, giving those as the total positive values for each dataset respectively.

Training/Testing:

The datasets will be split using the 70-30 set, with 70% being trained, and the other 30% being tested. The model be tested and trained on the two datasets respectively. The TF-IDF vectors will then be fed to the algorithms to create the ML model.

Evaluation:

To determine the efficiency and performance of the ML model, evaluation metrics such as accuracy, precision, recall, and F-1 scores will be all calculated. According to according to (Tigerschiold, 2022), precision is calculated by determining the number of true positives divided by the total amount of positives. Recall is calculated by the number of true positives divided over the number of true positives plus false negatives.

F1-scores uses the equation 2 * ((precision * recall) / (precision + recall)). Using the scikit-learn library for Python, all of these evaluations are built in and able to be used to evaluate the scores of these ML algorithms and models.

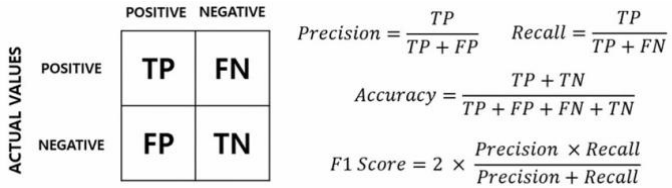

Figure: Precision, Recall, Accuracy, and F1 score (Buckland, 1994)

V. Results

Both datasets were trained and tested with both algorithms to create the ML model. Their Accuracy, Precision, Recall, and F1 Score were all obtained using NLTK and represented by the table and graphs provided below.

Healthcare Dataset:

Naïve Bayes		Logistic Regression	
Accuracy	0.8314	Accuracy	0.96726667
Precision	1	Precision	1
Recall	0.40591966	Recall	0.88466056
F1 Score	0.57744361	F1 Score	0.93880095

Payload Dataset:

Naïve Bayes		Logistic Regression	
Accuracy	0.92959663	Accuracy	0.97026314
Precision	1	Precision	1
Recall	0.18442427	Recall	0.65551839
F1 Score	0.31141589	F1 Score	0.79191919

Tables: Dataset Results (6)

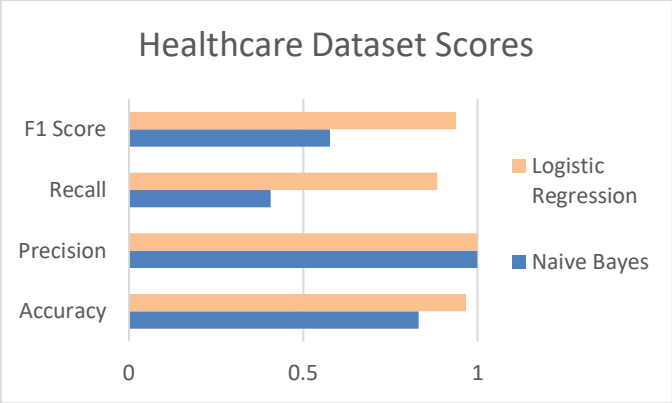

Figure: Healthcare Dataset with Evaluation Scores

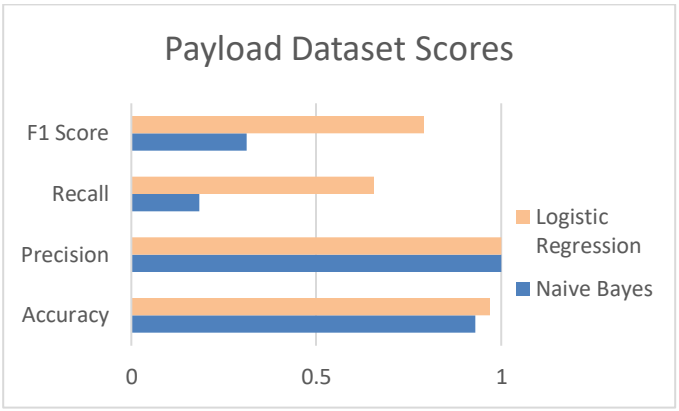

Figure: Payload Dataset with Evaluation Scores

The evaluation and results of both algorithms and how they performed on both datasets brought great insights into their performance. Starting with the precision of 100% indicated in all algorithms and datasets, meaning that when a model predicted an instance as positive, it was indeed positive because we knew from beforehand what is considered PII and what is non-PII. Starting with the Healthcare dataset, the Naïve Bayes classifier received an accuracy score of 83.14%, showing its overall correctness of predictions. A recall value of 40.59% suggested that the model struggled in identifying more than half of actual positive instances, thus giving a lower F1 score of 57.74%. The Logistic Regression model outperformed Naïve Bayes with an accuracy of 96.73% and recall value of 88.47%. This indicated a highly accurate model that effectively identified positive instances, leading to a very solid F1 score of 93.88%.

In the Payload dataset, the Naïve Bayes classifier had an accuracy of 92.96%. A very low recall of 18.44% indicated a challenge in identifying actual positive instances, ultimately giving a poor F1 score of 31.14%. The Logistic Regression model excelled and outperformed the naïve bayes again in this dataset with a solid accuracy of 97.03%, recall of 65.55% and better F1 score of 79.19% when compared to Naïve Bayes.

VI. Discussion & Conclusion

The performance of both Naïve Bayes and Logistic Regression across both datasets show how these two algorithms differ in their strengths and that the choice of which algorithm to choose definitely influences the effectiveness of the ML model. Overall, logistic regression showed far superior in performance to Naïve bayes with its ability of higher recall, leading to higher F1 scores. Having results from two different datasets reaffirmed this dominance for logistic regression over naïve bayes. These findings from the experiment not only underline the importance of algorithm selection in ML models, but also the importance of determining the strengths of specific algorithms and how they relate to the task at hand.

In conclusion, this project's goal was to find PII data within medical records, utilizing NLP techniques such as NER, TF-IDF, Naïve Bayes, and Logistic Regression. Through python code, this experiment was able to be computed and tested, giving us reasonable results and evaluations. We covered the

importance and need of PII detection and how it is needed in almost every field. NLP and ML can play an important role in PII detection with the use of techniques and tools such as NER, TF-IDF, and ML algorithms. This project was able to use these tools and test two different algorithms, where it saw logistic regression outperform naïve bayes. Future studies can look into other ML algorithms that can possibly surpass logistic regression, achieving higher recall and F1 scores. Finding other NLP techniques or ways to improve NER can also be looked into to help furthering the advancement of PII detection. The importance of PII detection only seems to be growing, creating a solid high performing and efficient system will become paramount for most companies.

REFERENCES

[1] Buckland,M.; Gey,F. The relationship between recall and precision. Journal of the American society for information science 1994, 45, 12–19. https://doi.org/10.1002/(SICI)10974571(199401)45:1%3C12::AID-ASI2%3E3.0.CO;2-L.

[2] DHS Privacy Office . (2017, December 4). DHS handbook for safeguarding sensitive pii -homeland security. https://www.dhs.gov/sites/default/files/publications/dhs%20policy%20directive%20047-01-007%20handbook%20for%20safeguarding%20sensitive%20PII%2012-4-2017.pdf

[3] Ghiasvand, O., & Kate, R. (2018). *Learning for clinical named entity recognition without manual annotations*. Science Direct. https://www.elsevier.com/locate/imu

[4] Gultiaev, A. A., & Domashova, J. V. (2022). Developing a named entity recognition model for text documents in Russian to detect personal data using machine learning methods. Science Direct.

[5] Hathurusinghe, R., Nejadgholi, I., & Bolic, M. (2021). A Privacy-Preserving Approach to Extraction of Personal Information through Automatic Annotation and Federated Learning.

[6] Hutchinson, T. (2020). Natural language processing and machine learning as practical toolsets for archival processing. Records Management Journal, 30(2), 155–174. https://doi.org/10.1108/RMJ-09-2019-0055

[7] IBM. (n.d.). What are naive Bayes classifiers?. IBM. https://www.ibm.com/topics/naive-bayes

[8] Kashti, R., Prasad, P. (2019) Enhancing NLP Techniques for Fake Review Detection.

[9] Kulkarni, P. (2021). Personally Identifiable Information (PII) Detection in the Unstructured Large Text Corpus using Natural Language Processing and Unsupervised Learning Technique.

[10] Pradeep. (2023, March 21). Understanding TF-IDF in NLP: A comprehensive guide. Medium. https://medium.com/@er.iit.pradeep09/understanding-tf-idf-in-nlp-a-comprehensive-guide-26707db0cec5

[11] Robert J Carroll, Will K Thompson, Anne E Eyler, Arthur M Mandelin, Tianxi Cai, Raquel M Zink, Jennifer A Pacheco, Chad S Boomershine, Thomas A Lasko, Hua Xu, Elizabeth W Karlson, Raul G Perez, Vivian S Gainer, Shawn N Murphy, Eric M Ruderman, Richard M Pope, Robert M Plenge, Abel Ngo Kho, Katherine P Liao, Joshua C Denny, Portability of an algorithm to identify rheumatoid arthritis in electronic health records, Journal of the American Medical Informatics Association, Volume 19, Issue e1, June 2012, Pages e162–e169, https://doi.org/10.1136/amiajnl-2011-000583

[12] Subramani, N., Luccioni, A., Dodge, J., (2023). Detecting Personal Information in Training Corpora: an Analysis.

[13] Silva, P., Gonçalves, C., & Godinho, C. (2020). Using Natural Language Processing to Detect Privacy Violations in Online Contracts.

[14] Tigerschiold, T. (2022, November 17). What is accuracy, precision, recall and F1 score?. What is Accuracy, Precision, Recall and F1 Score? https://www.labelf.ai/blog/what-is-accuracy-precision-recall-and-f1-score

[15] Weinberger, K. (2018). Lecture 6: Logistic regression. https://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote06.html#:~:text=In%20Naive%20Bayes%2C%20we%20first,(y%7Cx)%20directly.

[16] Welderufael, T. (2019). PrivacyBot: Detecting Privacy Sensitive Information in Unstructured Texts. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8931855

[17] Wei, Y.-C., Liao, T.-Y., & Wu, W.-C. (2022). Using machine learning to detect PII from attributes and supporting activities of information assets. Journal of Supercomputing, 78(7), 9392–9413. https://doi.org/10.1007/s11227-021-04239-9

Dataset links:

https://www.kaggle.com/datasets/prasad22/healthcare-dataset

https://www.kaggle.com/datasets/saurabhshahane/mlbased-cyber-incident-detection-for-emr/data