DATA SCIENCE PROJECT

CONCEPT SELECTION

"Development of an AI system for Classifying, Evaluating and Interpreting Privacy Policies and Terms of Conditions"

Table of Contents

1. Executive Summary

This report outlines the conceptual design of our AI system intended to classify, highlight, and summarize concerning sections of Privacy Policies and Terms and Conditions (T&Cs). This project aims to assist users in comprehending the digital legal documents they encounter, thereby enhancing their awareness and informed consent. Our system's primary goal is to develop a machine learning model capable of detecting these concerning sections with at least 85%-90% accuracy and transforming it into understandable descriptions. Additionally, we aspire to build a simple, user-friendly interface to receive input and deliver output for our solution.

We have collected, labelled, and trained the model with the Privacy Policies and T&Cs from both public dataset and hand collecting. Our approach involves data preprocessing, feature engineering, and the development of multiple different classification models. We will evaluate and compare model performance using accuracy, recall, precision, and F1 score as the main metrics. Regarding the summarization of problematic content, we have adopted two distinct approaches: Extractive and Abstractive Summarization. To determine which approach is more effective to be included in the final solution, we will conduct evaluations using the ROUGE (Recall-Oriented Understudy for Gisting Evaluation) [11] metric in addition to human assessments. The solution will be implemented with a simple user interface running on local host using Python script, providing as a demo version for testing.

When interpreting the results and recommendations of this solution or making decisions based on its findings, these limitations should be considered: the Imbalance between acceptable and unacceptable datapoints, the requirement of quality input data, model interpretability, scalability, and specifications. The overall of project planning and management is also included. The project lasts 9 weeks, breaking into 3 milestones: (1) Research and Data Collection; (2) Data labelling and Model development; and (3) Additional functions, Evaluation and Integration. Tasks are broken down and allocated to team members based on their capabilities and strengths, ensuring a reasonable workload allocation for each team member in accordance with the project timeline.

2. Introduction

Privacy policies or Terms and Conditions (T&Cs) are the foundation of digital privacy. These documents frequently set forth rules for the gathering, using, and disseminating of personal data. However, because these contracts are commonly lengthy, intricate, and loaded with legalese, it can be difficult for people to thoroughly understand the meaning of the policy or the implications of their consent [1]. The project, focusing on building an Artificial Intelligence (AI) system, aims to classify, highlight privacy-sensitive terms, and summarize the privacy policy as outputs, hence empowering individuals to make clearly informed decisions about their privacy usage. This report will attempt to provide a blueprint of our system design proposals from the input of the end-user to the result, providing details into our methodologies system development process. Additionally, a project management plan

is crafted encompassing timelines and task breakdowns, all aimed at ensuring the successful and high-quality completion of the project.

3. Objective & Desired Outcome

Our objective for this project is to architect and finetune an AI model that demonstrates adeptness in evaluating privacy policies and terms and conditions documents. This necessitates that the AI system strictly adhere to ethical standards, legal precepts, and user-centric standards. The design vision for the model encompasses its ability to delve deep into these documents. These pinpointing sections are either too complex or bear potential risks. The aspiration is to uplift user comprehension levels concerning how they are often presented in legal documents and improve users' understanding of the subject.

4. System Overview

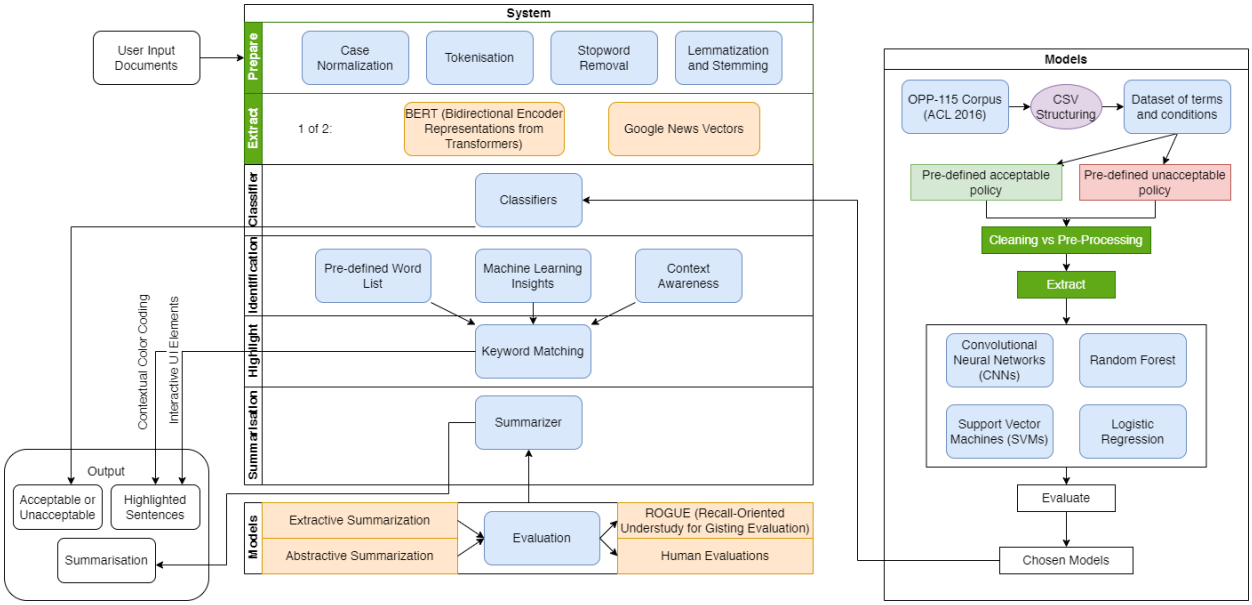

Figure 1.1. System overview

Our system, based on Humphrey O. Obie paper [22], initiates by taking a user document as input, followed by a series of data preprocessing steps. From here, the input data is subjected to data extraction using one of two proposed models: BERT [3] or Google News Vectors [4], with the selected model managing the preprocessing and feeding the results into a Classifier. The Classifier will be trained and refined with Corpus data and the same data preprocessing and extracting process as System structure (marked as green colour) using an array of models. The input will also be used to identify keywords and highlight potentially troubled phrases using various methods. A dual-model approach is employed for text summarisation tasks. Ultimately, the end-user will receive three kinds of output: a result of the input being acceptable or unacceptable, the section of the input highlighted as troublesome if the given input is deemed unacceptable, and a succinct summary that effectively communicates the core content and implications of the privacy policies to end-users.

5. Design Concept

A comprehensive approach encompassed various stages of machine learning to construct an AI system proficient in classifying, understanding, and clarifying legal documents for users. This transforms complex policy texts into understandable, classified, and summarized content, thus promoting greater data privacy comprehension. Here is our detailed methodology brief:

5.1. Input

The project was built based on several studies highlighting most users' neglect of privacy policies due to its complex and lengthy nature. Our system will be built as a web-based application where the end user can access the web page and type down the policy they want to examine.

5.2. System

5.2.1 Text Cleaning and Preprocessing

The given input will undergo a comprehensive cleaning phase to remove possible noise before it will be used to classify. We applied tokenization, stop word removal, case normalization, lemmatization, and stemming techniques.

Case normalization: This method involves converting original review texts to lowercase, ensuring we do not count the same word in different letter cases (like "Privacy" and "privacy") as separate features [13].

Tokenization: At the outset, content undergoes tokenization, which dissects vast blocks of text into individual, manageable tokens. We create a foundation for complex textual analysis by converting the text into these atomic units. [14]

Stopword Removal: Common English terms important to people can obstruct machine learning. Words such as "and", "the", and "is" barely change the meaning at all. To direct our models towards important content, we eliminate certain stopwords. [15]

Lemmatization and Stemming: While lemmatization convert words to their base or dictionary form, stemming truncates words to their root base [16]. For instance, "running" becomes "run". This reduction minimizes data redundancy and fosters better model comprehension. [17]

5.2.2 Feature Extraction

The transformation of raw text for machine learning is a crucial difficulty in natural language processing. Text is converted into numerical vectors for ML models using feature extraction. The best match for the intended results is determined after testing two pre-trained models:

BERT (Bidirectional Encoder Representations from Transformers): This cutting-edge model has significantly advanced the NLP domain due to its bidirectional understanding of text context. This language model is trained on the BooksCorpus, containing 800 million words, and the English Wikipedia, which has 2.5 billion words [2]. Instead of considering words or sentences in isolation, BERT examines

both the preceding and following text, capturing a more comprehensive semantic meaning [3]. By leveraging pre-trained BERT embeddings, we transform our textual data into high-dimensional vectors that encapsulate intricate linguistic patterns and contexts.

Google News Vectors: This model is based on Word2Vec, a two-layered neural net that processes text by vectorizing words [4]. The Google News model has been trained on vast amounts of news data, making its vectors rich in contextual information. By incorporating this model, we ensure our feature extraction is rooted in broad linguistic patterns observed in real-world text.

5.2.3. Classification

To classify the given input from end user, the model that is used in the system must be trained to know what it is dealing with. Thus, we need to collect a dataset about privacy policies and use the same techniques such as text cleaning and data preprocessing, feature extraction for the dataset before fitting it into the model. In this phase, we need to train four supervised algorithms combining with two methods of feature extraction mentioned above then evaluate to choose the best model for the system.

a. Data Collection

To classify the given input from end user, the model that is used in the system must be trained to know what it is dealing with. The primary training data source is the collection of policies named OPP-115 Corp from UsablePrivacy.[18] We will be working with the sanitized policies, which encompasses a plethora of privacy policy documents from various companies, formatted as HTML files. Critical data points were collected pertaining to privacy policies of various enterprises, ensuring the accuracy and relevance of our dataset. The team has been able to collect and process a dataset consists of more than 4500 data points with one data point being one paragraph from various terms and conditions, privacy policies, … The team has pre-defined the labels using the codebook as the rule.

b. Data Preparation

After manual extraction, these data points will be organized into structured CSV files using Python. This format provides ease for subsequent data processing. A new column, "Acceptable," are added to our dataset. Based on human values and ethical considerations, referring to our labelling codebook, each data point will be labelled as "acceptable" (1) or "unacceptable" (0) with the ratio of 50:50. This ratio is determined to ensure that the model can be exposed to many data points for an unbiased result.

c. Data Preprocessing and Model Training

The same techniques will be used as mentioned in section 5.2.1 and 5.2.2 to preprocess and extract the feature in the dataset. Based on the requirements and the nature of the project, the classification of privacy policies will be handled by supervised learning models, built as a backbone for a robust and comprehensive classification system for privacy policy documents. A suite of machine learning models is utilized to help us select a classification model that yields the optimal result, includes:

Convolutional Neural Networks (CNNs): While traditional machine learning techniques like Naive Bayes and Decision Trees are quick and easy, CNNs are adept at recognizing patterns within vast tracks of text. For the text data, CNNs identify and capture patterns or motifs within the textual sequences, allowing them to recognize specific structures or phrasings indicative of policy terms. With the setback being in the computational power, their hierarchical architecture enables them to detect simple and complex patterns, which can be influential in classifying and understanding intricate legal documents.[5]

Support Vector Machines (SVMs): SVMs excel in high-dimensional data, and their core functionality revolves around finding the optimal hyperplane that best separates data into classes. Given the high dimensionality of the given textual data due to the policies' complex nature especially after feature extraction, SVMs can be exceptionally effective. Their ability to handle complex decision boundaries makes them ideal for classifying our privacy policy segments as "acceptable" or "unacceptable." [6]

Logistic Regression Models: A probabilistic, linear classifier, logistic regression estimates the probability of an instance belonging to a particular class. Within our framework, logistic regression aids in quantifying the likelihood of a particular segment of a privacy policy being labelled as "acceptable" or "unacceptable." The probabilistic outputs can be pivotal in scenarios requiring a nuanced understanding beyond binary classification. [7]

Random Forest: An ensemble learning method, Random Forest uses multiple decision trees during training and outputs the mode of the classes (classification) or mean prediction (regression) of individual trees. Due to its ensemble nature, Random Forest offers resilience against overfitting and can efficiently handle large datasets with higher dimensionality. This model provides a broader perspective by considering multiple decision trees in our project, ensuring no singular pattern or anomaly disproportionately affects our classification.[8]

Cross-validation [19] and four classification metrics [21] will be used for the evaluation effort. Cross-validation is used to ensure our models' robustness and generalizability.

5.2.4. Identification & Highlighting

To help further users' comprehension of why specific policies are claimed to be unacceptable by the model, the system will take one step further and highlight which part of the input document is judged by the model to be problematic. To tackle this problem, multiple methods were considered:

Pre-defined Word List: Our team has curated a list of words and phrases often red flags in privacy policies. These could include terms that imply data sharing with third parties, ambiguous terminologies, or phrases hinting at excessive data collection. This list gives our system a benchmark against which to compare.

Machine Learning Insights: Leveraging the feature importance outputs from our machine learning models, we identify which sections or terms in the privacy policies significantly influence the

classification of a document as "acceptable" or "unacceptable." This automated identification supplements our pre-defined word list.

Context Awareness: A crucial aspect of our identification algorithm is context sensitivity. Words in isolation can be benign, but within a particular context, they can be problematic. For instance, "share" in "share with friends" is different from "share with third parties". The algorithm can learn and attune to these nuances to avoid false alarms.

To present problematic sections identified through our methods to the user in an intuitive manner, our system utilizes several user-friendly techniques. We implement contextual colour coding, which offers a spectrum of colours instead of a binary highlighting approach. In addition to colour coding, our platform incorporates interactive UI elements. When a user hovers over a highlighted section, they receive a brief explanation or insight into why that section is considered problematic, enhancing their understanding of the issues within the policy documents. This approach ensures a more user-friendly and informative experience.

5.2.5. Summarization

Our design concept emphasizes a dual-model approach for text summarization to address the challenge of sifting through lengthy and often cryptic privacy policies. This strategy ensures that users understand these documents concisely, enhancing transparency and trust. The overarching goal is to produce succinct and intelligible summaries that effectively communicate the core content and implications of the privacy policies to end-users. Two techniques will be employed:

Extractive Summarization: This method selects vital sentences or fragments from the original text to form a summary. Key advantages include retaining the original wording and context ensuring accuracy in representation. Utilizing libraries such as NLTK and employing TF-IDF scores to gauge sentence importance within the privacy policy content. [9]

Abstractive Summarization: This more sophisticated approach aims to understand the content and produce a 'human-like' summary, often reformulating phrases for brevity and clarity. It's beneficial when the original text contains redundant or verbose explanations. Leveraging pre-trained models like BART or T5 from the Hugging Face Transformers library, known for their proficiency in generating coherent and contextually relevant summaries. [10]

To ensure the quality and accuracy of our generated summaries, we employ a two-fold evaluation approach. Firstly, ROUGE (Recall-Oriented Understudy for Gisting Evaluation) metric will be used to address the generated summaries' quality and accuracy. ROUGE facilitates a comparison between the generated summaries and a reference summary (usually human-generated), assessing various facets like an overlap of n-grams, and longest common subsequence. Such a metric ensures our summaries are concise but also relevant and accurate in capturing the essence of the original document [10]. Secondly, we incorporate human evaluation to provide an assessment of summary quality. Though ROUGE can be

a great evaluation metric, it only accounts for syntactical matches rather than the actual meaning of the words. That is why human judgment should also be considered.

5.2.6. Integration & Deployment

The team has designed a pathway for deploying Python scripts on a local host for this project. We utilize Flask, a lightweight web framework, to simplify web application development with easy-to-use features. Some steps involve creating a Python script that defines the application's routes and running the Flask app via a local development server. Real production deployment typically involves web servers like Apache or Nginx with production-ready servers. However, for simplicity and clarity, the current setup is ideal for development and experimentation. [11]

5.3. Output

For the result, three types of output will be introduced to the end-user: A result of the user input document being classified as "Acceptable" or "Unacceptable". This will serve as the initial assessment for the user, visual encoding of the section deemed potentially problematic for the user in the said document (if the result being "unacceptable"). This will serve to grab the user's attention to the area of concern and help the user in the review process, and a concise summary of the document. The summary from the original document will contain the key information.

The actual output of the system can depend on the request and the needs of the end-user as it can save both the user's time and the computational resources from the system. Nevertheless, the result of the summarization module will interface with the classification and keyword-matching components for the actual model on the website, creating a cohesive website, promoting user awareness and informed decision-making for digital users.

6. Challenges

Various limitations and restrictions that may affect the process and outcome of a project are also considered. Addressing them early in the design process ensures that the proposed solution is feasible and realistic. Here is an elaboration on the design constraints for our project:

Data Balance: Given that our dataset comprises labels of 'acceptable' and 'non-acceptable', ensuring a fair representation of both classes is critical. A skewed or imbalanced dataset can lead to biased model predictions.

Data Quality and Accuracy: The reliability of our model is dependent on the quality and accuracy of the data we use. The data must be free from errors, inconsistencies, and bias. Manual labelling can also introduce human error, so checks and balances must be in place to mitigate the potential risks.

Model Interpretability: Our project's aim is not just classification but also making the results understandable for users. Therefore, even if a model offers high accuracy, it may not be suitable if it is a black box, and its decisions cannot be interpreted or explained to users.

Scalability: The solution should be scalable to accommodate growing datasets in the future. This implies that the infrastructure and the models should be designed keeping future growth in mind.

Specifications: In terms of hardware requirements, the machine learning models necessitate a setup comprising a minimum of 16 GB RAM and a CPU supporting four threads, with GPUs playing a crucial role in handling intensive computational tasks, especially in deep learning. High-speed configurations of SSDs can be a solution for efficient data access. On the software front, our project is centred around Python, utilizing libraries like Pandas, Numpy, Scikit-Learn, and Genshim. We leverage Google's Colaboratory platform to harness the advantages of cloud computing, facilitating collaborative Python execution and overcoming local resource limitations, mainly when testing with large models.

7. Conclusion

In conclusion, this project serves as an initiative to address the challenge of comprehending and navigating digital legal documents. We have employed machine learning models to accurately classify, highlight, and summarize concerning sections within these documents through our model's careful collection, labelling, and training with a dataset comprising over 4,500 data points. Our multifaceted approach, encompassing diverse classification models and summarization techniques, reflects our commitment to providing a comprehensive solution. This AI system can be developed further to be a valuable solution for enhancing user awareness and informed consent, which is one direction to the potential of artificial intelligence in solving real-world challenges.

8. References

[1] A. Hanlon and K. Jones, "Ethical concerns about social media privacy policies: do users have the ability to comprehend their consent actions?," *Journal of Strategic Marketing*, pp. 1–18, Jul. 2023, doi: 10.1080/0965254X.2023.2232817.

[2] B. Bhasuran, "BioBERT and Similar Approaches for Relation Extraction," in Biomedical Text Mining, K. Raja, Ed. New York, NY: Humana, 2022, vol. 2496, Methods in Molecular Biology. [Online]. Available: https://doi.org/10.1007/978-1-0716-2305-3_12.

[3] S. Baranwal, "Understanding BERT," Towards AI, 18 February, 2020. [Online]. Available: https://pub.towardsai.net/understanding-bert-b69ce7ad03c1

[4] "word2vec," Google Code Archive, 30 July, 2013. [Online]. Available: https://code.google.com/archive/p/word2vec/

[5] "What are convolutional neural networks?" IBM Topics. [Online]. Available: https://www.ibm.com/topics/convolutional-neural-networks

[6] R. Gandhi, "Support Vector Machine: Introduction to Machine Learning Algorithms," Towards Data Science, 8 June, 2018. [Online]. Available: https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47

[7] J. M. Hilbe, Logistic Regression Models. 2009. [Online]. Available: https://books.google.com.au/books?id=tmHMBQAAQBAJ

[8] S. J. Rigatti, "Random Forest," J. Insur. Med., vol. 47, no. 1, pp. 31–39, 2017. [Online]. Available: https://doi.org/10.17849/insm-47-01-31-39.1

[9] S. Rahul, S. Adhikari, and Monika, "NLP based Machine Learning Approaches for Text Summarization," in 2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2020, pp. 535-538. doi: 10.1109/ICCMC48092.2020.ICCMC-00099.

[10] Y. Chen and Q. Song, "News Text Summarization Method based on BART-TextRank Model," in 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing, China, 2021, pp. 2005-2010. doi: 10.1109/IAEAC50856.2021.9390683.

[11] C.-Y. Lin, "ROUGE: A Package for Automatic Evaluation of Summaries," Information Sciences Institute, University of Southern California, Marina del Rey, CA, 90292. [Online]. Available: https://aclanthology.org/W04-1013.pdf

[12] "How a Flask app works". PythonHow. [Online]. Available: https://pythonhow.com/python-tutorial/flask/How-a-Flask-app-works/

[13] R. Prabhakar, M. Christopher, and S. Hinrich, *Introduction to Information Retrieval*, 3rd ed. Cambridge University Press, 2008.

[14] J. Dan and M. James, *Speech and Language Processing*, 2nd ed. Pearson, 2008.

[15] E. D. Liddy, “Natural Language Processing,” in *Encyclopedia of Library and Information Science*, 2nd ed.New York: Marcel Dekker, 2001.

[16] B. Steven, K. Ewan, and L. Edward, *Natural Language Processing with Python*. O’Reilly Media, 2009.

[17] M. F. Porter, “An algorithm for suffix stripping,” *Program*, vol. 14, no. 3, pp. 130–137, Mar. 1980, doi: 10.1108/eb046814.

[18] “OPP-115 Corpus (ACL 2016),” *Usable Privacy Policy Project*. https://usableprivacy.org/data (accessed Sep. 15, 2023).

[19] R. Kohavi, “A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection,” in *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2*, in IJCAI’95. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1995, pp. 1137–1143.

[20] D. Richard, H. Peter, and S. David, *Pattern Classification,* 2nd ed. Wiley-Interscience, 2000.

[21] M. Sokolova and G. Lapalme, “A systematic analysis of performance measures for classification tasks,” *Inf Process Manag*, vol. 45, no. 4, pp. 427–437, Jul. 2009, doi: 10.1016/j.ipm.2009.03.002.

[22] H. O. Obie *et al.*, “Automated Detection, Categorisation and Developers’ Experience with the Violations of Honesty in Mobile Apps,” Nov. 2022.