JOMCOM

Journal of Millimeterwave Communication, Optimization and Modelling

editor in chief

Assoc. Prof. M. Tahir GUNESER

Volume:	3
Issue:	2
Year:	2023
ISSN:	2791-92-93

CONTENT

About the Journal

Journal of Millimeterwave Communication, Optimization and Modelling (JOMCOM) is an international on-line and refereed journal published 2 times a year (June and December) in English. Journal of Millimeterwave Communication, Optimization and Modelling (JOMCOM) published its first issue in 2021 and has been publishing since 2021. Manuscripts in JOMCOM Journal reviewed of at least 2 referees among the referees who have at least doctorate level in their field.

Journal of Millimeterwave Communication, Optimization and Modelling (JOMCOM) is an international online journal that is published 2 times in a year in English.

The purpose of JOMCOM is publishing the scientific research in various fields of communication.

All kinds of transactions and the application about the journal can be made from **https://jomcom.org**

The scientific responsibility of articles belongs to the authors.

ISSN: 2791-9293

Editor in Chief:

PUBLISHER

Assoc. Prof. Muhammet Tahir GÜNEŞER

Aims & Scope

Communication Technologies: Journal of Millimeter-wave Communication, Optimization and Modelling (JOMCOM) publishes original research and review articles in Communication Technologies, Innovative Technologies, and Systems in the broad field of Information-Communication Technology. Purpose of JOMCOM; To create value in the field by publishing original studies that will contribute to the literature in wireless communication sciences and be a resource for academia and industrial application whole over the world. Besides, JOMCOM aims to bring the valuable work of researchers working in Communication studies to a broader audience at home and abroad. Readership of JOMCOM; valuable representatives of the wireless communication area, especially those who do academic studies in it, and those who do academic studies about modelling and system design and other interested parties. Since JOMCOM will appeal to a broader audience in article submissions, it prioritizes studies prepared in English.

Optimization and Modelling: Journal of Millimeter-wave Communication, Optimization and Modelling (JOMCOM), within the scope of Wireless Communication Sciences, publishes articles on communication theory and techniques, systems and networks, applications, development and regulatory policies, standards, and management techniques. It also reports experiences and experiments, best practices and solutions, lessons learned, and case studies. Additional studies on System Design, Modelling and Optimization. Subject areas of interest covered in the journal include the following but are not limited to:

5G-6G Technologies

Circuits for Optical Communication Systems

Antenna Design

Communication Design Materials

Fiber Optic Communication

Innovative Designs for Communications

Integrated Circuits for Communications

Optimization Methods on Engineering

Realization of Antenna Systems

Realization of Microwave, Radar, and Sonar Systems

RF Circuits

System Design

Visible Light Communication

Wireless Communication

Bot Account Analysis on Social Media with Artificial Intelligence Support on Twitter Example

Received: 1 January 2023; Accepted: 28 February 2023 ***Research Article***

Refik Söylemez
Department of Computer Engineering
Istanbul Ticaret University
Istanbul, Türkiye
refiksoylemez@gmail.com
ORCID: 0000-0003-4580-067X

Ali Boyacı
Deparment of Computer Engineering
Istanbul Ticaret University
Istanbul, Türkiye
aboyaci@ticaret.edu.tr
ORCID: 0000-0002-2553-1911

***Abstract*— Launched in 2006 and available in 33 languages, Twitter is a social media platform that initially allowed users to share messages of up to 140 characters. It has since evolved into a platform used for various purposes, including communication, organization, sales and marketing, and microblogging. There have been numerous studies on data analysis (emotion, influence, education and training opportunities, political polarization, etc.) and data input (bot analysis, etc.) related to "tweets" - messages entered by users on Twitter since its inception. These studies have focused on analyzing bot tweets and accounts in order to prevent Twitter messages from informing people of false news. The accuracy performance of these analyses carried out with machine learning methods varies depending on the selection of training data used to create the model. In this study, the impact of randomly selected different training data on model performance was focused on and examined.**

Keywords— *Twitter, machine learning, bot tweet*

I. Introduction

As social media technology and popularity have grown, people today are organizing on Twitter and attempting to influence social peace and governments by starting various conflicts and wars. Many uprisings use bot accounts and try to change the agenda with bot tweets. To eliminate this problem that threatens society, it is necessary to discover bot accounts and take appropriate action. The detection of bot accounts and tweets is of great importance. If undetected, bot tweets can inform people of false news and guide them toward unrealistic trends. Today's widespread use of Twitter bots has led to academic research focusing on this subject. For example, one study has pointed out that between 9% and 15% of active Twitter accounts are bot accounts. The study discusses the interaction between simple bots and those that mimic human behavior. It also attempts to classify different types of bot accounts - those that send spam, promote themselves, or publish content from linked applications - through clustering analysis. The study highlights the various purposes for which bot accounts are used [1]. The development of bots' ability to respond and process information like humans are expected to have a wide-ranging impact and potentially lead to various sociological, psychological, political, and even economic effects. Therefore, in recent years, research in this area has increased, taking into account the impact of detecting bot tweets and accounts.

II. Conceptual And Theoretical Framework / Literature

A study from 2017 examined the use of Twitter in political communication, including the behavior of political users on the platform, the role of Twitter in political discussions, and the use of the platform in election campaigns, with a focus on the influence of bot accounts in the 2016 US elections [2].

Social media bots are prevalent today. As they are developed with the ability to respond like humans, they become increasingly influential. Therefore, detecting bot tweets and accounts using artificial intelligence algorithms is very important and has become a topic of much research in recent years. Bot accounts created by automated programs have targeted Twitter's increasing user numbers and overall structure. These bots have provided a platform for the spread of both good-faith content, such as news and blog updates, and spam or malicious content. Bots generally aim to follow many user accounts and be followed back randomly. Many efforts have been made to solve the problem of spam bots on social platforms. Different methods, such as extracting the text content of tweets, redirecting embedded URL addresses in other posts, and classifying the opening pages of URLs, have been tried to address this issue. A composite tool that can match tweets with commonly used basic templates has been proposed, going beyond the difficulty of labeling tweets without URLs as spam tweets [3]. A bio-inspired technique was introduced to model online social media user behaviors [4]. Instead of using more complex traditional feature engineering or natural language processing (NLP) tools, word embeddings were tried to encode tweets. This advantage allows the bot detection scheme to be faster and easier to implement and deploy. A Recurrent Neural Network (RNN) model using word embeddings, particularly a BiLSTM, was introduced to distinguish Twitter bots from human accounts. The study, which does not require prior knowledge or assumptions about user profiles, friendship networks, or past behaviors of the target account and is based only on tweets, and does not require heavy feature engineering, it is the first to develop an RNN model using word embeddings to detect bots. Their experiments on the publicly available Cresci-2017 dataset showed that models without hand-crafted feature engineering could achieve similar performance compared to existing studies [5]. The study evaluated the effectiveness of 30 classification algorithms for detecting bot tweets using supervised classification. Tree-based supervised classifiers performed the best, with the Random Forest classifier achieving the highest accuracy. The study also applied standard boosting and bagging techniques to further improve the accuracy of the Random Forest classifier [6]. Additionally, the study presents a system that utilizes supervised machine learning techniques to dynamically detect Twitter bot accounts. The classification results show a very high accuracy rate for this specific application [7]. An unsupervised method for detecting spam robots by comparing their behaviors to identify similarities among automatic accounts has been proposed. This bio-inspired method for modeling online user behaviors is called "Digital DNA" sequences. Extracting digital DNA from an account means associating it with a series of codes that encode behavioral information for that account. Although it achieves good detection performances, many handcrafted behavioral features are still required [4]. There are also methods for identifying Twitter bots that rely on the assumption that bots differ from humans in fundamental ways. These differences can be divided into two categories: technical differences and differences based on purpose. Bots are computer programs that can act instantly, while humans need time to think and may be busy with other tasks. Therefore, it can be assumed that the timing and direction of the content published differs from human behavior to bot behavior. Additionally, bots have clear goals, such as disseminating political messages and making references. Bots carefully bring specific content to the attention of users, hashtags, and URLs [8].

III. Method

In this study, the Social Honeypot dataset was used. The dataset was collected on Twitter from December 30, 2009, to August 2, 2010. It includes 22,223 spamming users, their following counts over time, 2,353,473 tweets, 19,276 legitimate users, their following counts over time, and 3,259,693 tweets [6]. After downloading the dataset made available as open source, data preprocessing steps were applied in the Python environment in the first step. These steps include removing punctuation marks, converting words to lowercase, removing repeating words, and lemmatization. Verbal expressions must be made meaningful for machine learning or deep learning algorithms. Therefore, words must be expressed numerically. Algorithms such as One Hot Encoding, TF-IDF, Word2Vec, FastText, and Count Vectorizer, known as word embedding techniques used to solve such problems, allow words to be expressed mathematically.

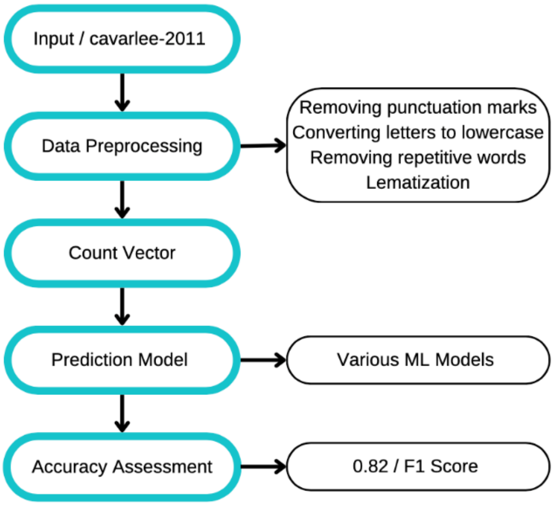


Fig. 1. Simple Workflow

In the second step, the data set, cleaned with preprocessing steps, is transformed into a vector form with Count Vectorizer and becomes input to the machine learning model in a format ready to be used.

IV. Results

A. Training with Default Hyperparameters

After splitting our dataset, which contained 1000 tweets, into a 25% test set and a 75% training set, the training phase was completed using various machine learning algorithms with their default parameters. The following accuracy results were obtained.

TABLE I. Accuracy Assessment 1

Algorithm	F1 Score Test	F1 Score Train
GaussianNB	0.83	0.99
MultinomialNB	0.81	0.97
ComplementNB	0.81	0.97
LinearSVC	0.80	0.99
SGDC	0.80	0.99
Logistic Regression	0.78	0.99
Random Forest	0.76	1.00
Decision Trees	0.74	1.00
Bagging	0.74	0.97
KNeighbors	0.69	0.68
AdaBoost	0.63	0.75

Based on the results obtained using the same test data, it can be seen that the best results were obtained using the Gaussian Naive Bayes algorithm when evaluating based on the F1 score. In the performance ranking, the Linear Support Vector Machines algorithm stands out immediately after the Naive Bayes algorithms, which are defined by probability principles. Machine learning models that can generalize well in problems with large feature spaces, such as SVM, produce better results than other models in text classification due to this power [9].

B. Training with Cross-Validation and Hyperparameter Optimization

The Logistic Regression, Decision Tree, Linear Support Vector Machines, Gaussian Naive Bayes, Multinomial Naive Bayes, Complement Naive Bayes, Random Forest, and K Nearest Neighbors algorithms were trained using hyperparameter optimization to improve the results obtained with default parameters. The Cross Validation method was also applied during training in addition to the previous study.

Cross-validation is a method used to evaluate and compare machine learning models by splitting the data into two sets: one for training the model and the other for validation. In simple cross-validation, the training and validation sets are rotated in such a way that each data point has an opportunity to be used for validation. This process involves using a diverse set of data for training and testing the model, which can provide a more accurate evaluation of the model's performance [10]. This ensures that each classifier is trained with a large number of parameter combinations and also prevents overfitting. The accuracy results obtained from the process are shown below.

TABLE II. ACCURACY ASSESSMENT

Algorithm	F1 Score Test	F1 Score Train
Random Forest	0.82	1.00
GaussianNB	0.82	0.99
Logistic Regression	0.82	1.00
MultinomailNB	0.82	0.99
ComplementNB	0.82	0.99
LinearSVC	0.71	0.99
SGDC	0.71	0.99
KNeighbors	0.69	1.00
Decision Trees	0.48	0.60

TABLE III. ACCURACY ASSESSMENT 3 - RANDOM STATE: 10

Algorithm	F1 Score Test	F1 Score Train
LinearSVC	0.81	0.99
MultinomailNB	0.81	0.97
ComplementNB	0.81	0.97
GussiamNB	0.81	0.99
Logistic Regression	0.80	0.99
SGDC	0.78	0.99
Random Forest	0.76	0.99
Decision Trees	0.74	1.00
KNeighbors	0.67	0.67

According to the performance graphs on the above, it can be seen that cross-validation and hyperparameter optimization had a positive effect on the performance of most algorithms. When examining the algorithms, it can be seen that Naive Bayes-based classifiers performed well in both experiments. This is because the Naive Bayes algorithm classifies based on probability principles. Studies have demonstrated that such classifiers give better results in language processing problems than linear classifiers.

C. Training with Different Random State Parameters

75% of training data is randomly split. However, this randomness can be controlled by a parameter. When this parameter changes, the data on which the machine learning models are also trained changes; therefore, the model results are also affected by this. The training data was created with four different randomness parameters to investigate the effect of training data split into different randomness on model results. A total of 36 results were obtained by training nine different algorithms with four randomly generated training data. The results obtained are shown in the tables below.

TABLE IV. ACCURACY ASSESSMENT 4: RANDOM STATE: 40

Algorithm	F1 Score Test	F1 Score Train
GussianNB	0.84	0.99
LinearSVC	0.83	0.99
MultinomailNB	0.82	0.98
ComplementNB	0.82	0.98
Logistic Regression	0.80	0.99
SGDC	0.79	0.99
Random Forest	0.75	1.00
Decision Trees	0.72	1.00
KNeighbors	0.65	0.69

V. DISCUSSION AND RESULTS

During the training part of a machine learning model, the data available is divided into two parts: training and testing. During training, the model does not see the test data. Therefore, the data that will be entered as training data for the model may vary depending on the given randomness parameter. A large number of training-test combinations are created depending on the size of the data. Therefore, it will be logical to take the accuracy averages of models trained with different randomness parameters to determine the most reliable accuracy. In addition, optimization can be done for the randomness parameter according to the desired performance in the study. According to the experiment results, the change in this parameter did not cause significant changes in performance. That is, there is no need to apply the optimization for hyperparameters for the randomness.

According to the results, the change in randomness only caused performance difference between 0-2%

Across experiments, Gaussian Naive Bayes and Linear Support Vector Machines consistently outperformed tree-based algorithms in text classification. Their probabilistic nature and linear approach make them superior choices.

TABLE V. ACCURACY ASSESSMENT 5: RANDOM STATE: 100

Algorithm	F1 Score Test	F1 Score Train
SGDC	0.81	0.99
GaussianNB	0.80	0.99
MultinomailNB	0.80	0.98
ComplementNB	0.79	0.99
LinearSVC	0.88	0.99
Logistic Regression	0.75	0.99
Random Forest	0.75	1.00
Decision Trees	0.75	1.00
KNeighbors	0.65	0.69

TABLE VI. ACCURACY ASSESSMENT 6: RANDOM STATE: 300

Algorithm	F1 Score Test	F1 Score Train
GussianNB	0.84	0.99
LinearSVC	0.82	0.99
MultinomailNB	0.81	0.98
Logistic Regression	0.82	0.98
ComplementNB	0.81	0.99
SGDC	0.80	0.99
Random Forest	0.77	0.99
Decision Trees	0.76	1.00
KNeighbors	0.69	0.68

CONTRUBITION OF THE AUTHORS

Asst. Prof. Dr Ali Boyacı played a pivotal role in guiding and supervising the research project. His expertise in relevant fields provided invaluable insights that shaped the direction of the study. Hi were actively involved in conceptualizing the research design, formulating research questions, and advising on methodology.

As the lead author of this article, Refik, the focus was on conducting extensive research regarding the impact of training data selection on the performance of machine learning models in analyzing Twitter data. This encompassed designing and executing experiments to systematically evaluate the influence of different training data sets on the accuracy of the models. Additionally, a comprehensive analysis of the results was undertaken, drawing meaningful conclusions and insights from the findings.

ACKNOWLEDGMENT

I would like to express my sincere gratitude to my supervisor, Asst. Prof. Ali Boyacı for their invaluable guidance and support throughout the research process. Their expertise and mentorship were instrumental in the successful completion of this work.

CONFLICT OF INTEREST

There is no any conflicts of interest between the authors.

STATEMENT OF RESEARCH AND PUBLICATION ETHICS

Research and publication ethics were observed. Ethics committee approval was obtained for research conducted in all branches of science that requires ethics.

TABLE VII. ACCURACY COMPARISON

Algorithm	Minimum F1 Score	Maximum F1 Score	Mean F1 Score
GussianNB	0.81	0.84	0.82
LinearSVC	0.79	0.83	0.81
MultinomailNB	0.80	0.82	0.81
Logistic Regression	0.78	0.82	0.80
ComplementNB	0.80	0.82	0.81
SGDC	0.79	0.80	0.81
Random Forest	0.75	0.77	0.76
Decision Trees	0.73	0.76	0.75
KNeighbors	0.65	0.69	0.67

REFERENCES

[1] Varol, O., Ferrara, E., Davis, C. A., Menczer, F., & Flammini, A. “Online Human-Bot Interactions: Detection, Estimation, and Characterization”, 2017.

[2] Campos Domínguez, E. M. (2017). Twitter y la comunicación política. El profesional de la información, 26(5),785-794.

[3] HongyuGao,YanChen,KathyLee,DianaPalsetia,andAlokNChoudhary. 2012. Towards online spam filtering in social networks.. In NDSS, Vol. 12. 1–16.

[4] S. Cresci, M. Petrocchi, A. Spognardi, and S. Tognazzi, "On the capability of evolved spambots to evade detection via genetic engineering," *Online Social Networks and Media*, vol. 9, pp. 1–16, 2019.

[5] Feng Wei and Uyen Trang Nguyen, "Twitter Bot Detection Using Bidirectional Long Short-term Memory Neural Networks and Word Embeddings" in IEEE TPS 2019.

[6] K. Lee, B. D. Eoff, and J. Caverlee, "Seven months with the devils: A long-term study of content polluters on twitter," in Proc. Fifth Int. AAAI Conf. Weblogs Social Media, 2011.

[7] M. Alsaleh, A. Alarifi, A. M. Al-Salman, M. Alfayez, and A. Al-muhaysin, "Tsd: Detecting sybil accounts in twitter," in *Proc. 13th Int. Conf. Mach. Learning and Appl.*, 2014.

[8] Jürgen Knauth. 2019. Language-Agnostic Twitter-Bot Detection. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 550–558, Varna, Bulgaria. INCOMA Ltd.

[9] Bellman, R.E. Adaptive Control Processes; Princeton University Press: Princeton, NJ, USA, 1961. [Google Scholar]

[10] Refaeilzadeh, P., Tang, L., Liu, H. (2009). Cross-Validation. In: LIU, L., ÖZSU, M.T. (eds) Encyclopedia of Database Systems. Springer, Boston, MA

Data Security Techniques and Comparison of Differential Privacy Techniques in Bioinformatics

Received: 29 January 2023; Accepted: 16 February 2023 *Research Article*

Nilgün İncereis
Department of Computer Engineering, Distance Education Application and Research Center
Istanbul Okan University, Bartın University
Istanbul, Bartın, Türkiye
niincereis@stu.okan.edu.tr
nincereis@bartin.edu.tr
ORCID: 0000-0001-5508-8159

Hilal Çakır
Deparment of Computer Engineering
Istanbul Okan University
Istanbul, Türkiye
hilcakir@stu.okan.edu.tr
ORCID: 0000-0002-5378-7930

Bekir Tevfik Akgün
Software Development
Yeditepe University
Istanbul, Türkiye
bekirtevfik.akgun@yeditepe.edu.tr
ORCID: 0000-0002-9726-1340

Abstract— Bioinformatics data is data containing information about biological systems and processes. This data can include genomic data, proteomic data, metabolic data, and similar data. The processing and analysis of bioinformatics data aims to achieve important goals such as conducting scientific research and improving healthcare systems. Data security of bioinformatics data ensures the security of data during processing and analysis as well as protecting individual privacy. In this study, five of the known techniques for data security in bioinformatics have been studied. These techniques include: data anonymization, data masking, data encryption, and role-based access control, and differential privacy. In this study, it is aimed to create functions for the above-mentioned data security techniques by using the dataset obtained from 1000 patients with lung cancer, and to anonymize the dataset by using Laplacian, Gaussian and Exponential mechanisms from differential privacy techniques. Looking at various comparison parameters from the differential privacy techniques, it is concluded that the Laplacian technique strikes the best balance between privacy and utility as it provides the highest privacy guarantee and accuracy, as well as the lowest noise and robustness.

Keywords— Bioinformatics, data security, data anonymization, data masking, data encryption, role-based access control, differential privacy.

I. Introduction

Today, bioinformatics is a field that integrates computer science and biology. Computer technology and software [1], [2], [3] are used in this field to collect, store, analyze and understand biological data. By analyzing genetic data, bioinformatics research helps to understand the composition and capabilities of human and animal genomes. This topic can also be used for modeling and simulation [4], [5] of many biological systems.

In other words, bioinformatics is an active research area that aims to develop intelligent systems for molecular biology analysis. Many methods based on formal linguistic theory, statistical theory, and learning theory have been developed to model and analyze biological sequences such as DNA, RNA, and proteins. In particular, grammatical inference methods are expected to find some grammatical structures hidden in biological sequences. A study [1] provides an overview of a number of grammatical approaches to biological sequence analysis and related research, focusing on learning stochastic grammars from biological sequences and predicting their functions based on learned stochastic grammars.

The science of bioinformatics focuses on using information technology to process, store, and analyze biological data. The methods utilized to store and safeguard this data are referred to as data security [6]. These methods are crucial since bioinformatics data frequently contains private and confidential information. Bioinformatics data, for instance, may comprise details such as genetic, clinical, or biological samples. In addition to ensuring that this data is utilized precisely and consistently, data security is crucial for safeguarding the privacy and security of personal information.

Because it combines computer science, molecular biology, and genetics, bioinformatics encompasses issues like genomics, proteomics, and metabolomics and enables the analysis of this data using techniques like artificial intelligence, machine learning, and data mining. However, the security of this data is crucial since it is frequently vital data. The protection of data from unlawful access, alteration, or destruction is ensured by data security. Methods including data anonymization [7], [8], data masking [9], data encryption [10] and role-based access control [11], and differential privacy [12], [13] are examples of data security strategies used in bioinformatics. These methods protect the accuracy and integrity of your data while ensuring its security.

In this study, an introduction to the field of bioinformatics was made in Chapter I. In Chapter II, the importance of bioinformatics data is explained. Chapter III describes 5 of the known data security techniques for bioinformatics. Chapter IV outlines the advantages of known data security techniques in bioinformatics. In Chapter V, applications of known data security techniques are explained.

II. The Importance Of Bioinformatics Data

In this section, bioinformatics data and its privacy, a brief history of bioinformatics, and present and future of bioinformatics will be discussed.

A. Bioinformatics Data and Its Privacy

Bioinformatics data is data that contains information about biological systems. This data may include data such as genetic information in an individual's DNA (genomic data) [14], structure of proteins (proteomic data) [15], and biochemical reactions (metabolic data). It targets important purposes such as processing and examining bioinformatics data, conducting scientific research and improving health systems.

During the processing and examination of bioinformatics data, it is important to protect the privacy of individuals and ensure the security of the data. To promote scientific advancement and cut costs, there have been greater efforts to share research data. It is now obvious that no scientist can promise complete anonymity, and it is also becoming more widely accepted that research will be more successful if scientists have more knowledge about the subjects, and that being recognizable has certain advantages [16]. However, Concern over the privacy of medical information has long been expressed by patients and study participants. In a study [17], 92% of the participants preferred to be requested permission before having their health information used for anything other than medical care, and 83% of them wanted to know the specifics of the research before consenting to have their health records used. According to the study, there are several topics that are particularly delicate, such as lists of previous procedures and current drugs, as well as family medical history, genetic abnormalities, mental illness, and drug or alcohol-related occurrences. There are ethical concerns regarding the ability of subjects with cognitive impairment to provide informed consent or people with addiction to take part in research that administer medicines of dependence [16].

B. *History of Bioinformatics*

More than 50 years ago, when desktop computers were merely a theory and DNA sequencing was not yet possible, bioinformatics was just beginning. Early in the 1960s, the use of computer techniques to protein sequence analysis lay the groundwork for bioinformatics (notably, de novo sequence assembly, biological sequence databases and substitution models). DNA analysis later developed as a result of developments in both computer science and molecular biology, which resulted in increasingly powerful and compact computers as well as new software that was better suited to handle bioinformatics tasks. These developments allowed for easier manipulation and sequencing of DNA. Significant advancements in sequencing technology and cost-cutting measures during the 1990s and 2000s led to an exponential growth of data [18].

C. *Present and Future of Bioinformatics*

Only a few of the current applications for bioinformatics include gene expression analysis [19], genome annotation [20], protein structure prediction [15], and drug development. It is also being used to identify genetic risk factors for diseases and to develop personalized medical methods [21].

It is anticipated that bioinformatics will continue to be essential to biology and medicine in the future. The analysis and interpretation of biological data using machine learning and artificial intelligence methods is one area of great interest. These methods could significantly increase our comprehension of biological systems and quicken the pace of scientific discovery [22].

The development of tools and methods for analyzing and interpreting data from large-scale, multi-omic studies [23], which involve the simultaneous analysis of multiple types of biological data, such as genomics [14], transcriptomics [24], and proteomics [15], is another area of focus for future bioinformatics research.

II. Known Data Security Techniques For Bioinformatics

There are many data security techniques for bioinformatics. In this study, data anonymization, data encryption, masking and role-based access control methods are given below.

A. *Data Anonymization*

Data anonymization [7] is the process of removing personal identifying information from data sets so that the individuals who are the subject of the data cannot be identified. This is often done to protect the privacy of individuals and to comply with laws and regulations that require personal data to be kept confidential. In the field of bioinformatics, data de-identification is important because it enables researchers to share data and collaborate without compromising the privacy of individuals [16].

By incorporating or integrating the data of study participants into bigger repositories, the practice of building massive databases has grown more widespread. As a result, sufficient sample numbers may be used for analysis of the kind seen in genome-wide association studies. The initial permission procedure for the smaller research frequently did not include the concept of data sharing, which is the joining of smaller datasets into bigger, independent datasets. Personal identifiers must be deleted from the data to ensure participant anonymity, and a coded link may be maintained between a participant's data and identity to enable potential therapeutic updates, longitudinal epidemiology studies, or the return of specific study findings [25].

There are several ways to anonymize bioinformatics data, including the following:

- *Remove personal identifiers:* Personal identifiers such as names, addresses, and social security numbers should be removed from the dataset.
- *Use anonymized identifiers:* Instead of using personal identifiers, anonymized identifiers can be used to identify records. These identifiers should not be linked to any personal information.
- *Remove or modify sensitive data:* Sensitive data such as medical diagnoses or genetic information should be removed or modified to prevent re-identification of individuals.
- *Use data aggregation and generalization:* Aggregating data and using general categories (e.g., age ranges) rather than specific values can further reduce the risk of re-identification.
- *Apply statistical techniques:* Statistical techniques such as noise injection and k-anonymization can be used to further de-identify the data.

To effectively protect the privacy of individuals, it's crucial to carefully consider the level of anonymization required for a given dataset and to use a combination of these techniques.

There are some libraries which can be used for data anonymization:

- *pynonymizer*: It is a Python library [26] which is used worldwide to convert private production database dumps into anonymous copies.
- *Faker*: It is a Python package [27] that can create fictitious data for the biological private data.

B. Data Masking

Data masking [9] is concealing or changing certain data in a data set. This procedure is frequently carried out to safeguard the data's confidentiality or to check its correctness. There are several ways to alter and recreate data sets using data masking techniques. For instance, it is possible to remove specific data from a data collection or disguise the names of the individuals who are included in it. In this manner, both the data set and its accuracy and privacy are preserved.

In the realm of bioinformatics, there are several libraries and tools that may be used for data masking. These may consist of:

- *py-anon*: This Python module [28] offers a straightforward and adaptable framework for anonymizing and masking sensitive data.
- *DataShield*: This R package [29] offers a selection of safe calculations and sensitive data masking functions, and it is built to operate in a distributed setting.
- *Anonymizer*: A selection of tools for data masking and anonymization, as well as supports for data modification, generalization, and suppression, are provided by this Java package [30].
- *Data Masker*: This [31] is a command line tool that provides data masking functionality.
- *DataRobot*: This [32] is a software platform for data science and also provides data masking functionality.

Using these frameworks and tools, data masking strategies may be applied to massive datasets in a variety of forms, including CSV, JSON, and SQL databases. It may also make use of various encryption keys and techniques. These libraries include pre-built masking routines and masking rules, as well as the ability to employ masking techniques programmatically.

It is crucial to consider that each library or tool has certain benefits, functions, and capabilities. The particular use case and requirements also determine the library or tools it should be used.

C. Data Encryption

Particularly in the field of bioinformatics, where sensitive and private information is frequently kept and transmitted, data encryption [10] is a significant component of data security. Data is protected from potential breaches and misuse via encryption, which ensures that it cannot be viewed or understood by unauthorized parties.

Various techniques, such as hash functions, symmetric key encryption, and asymmetric key encryption, can be used to encrypt bioinformatics data. Data is encrypted and decrypted using the same shared key in symmetric key encryption. This method is quick and efficient, but it necessitates a secure key exchange between the parties, which could be challenging. Asymmetric key encryption, also known as public key encryption, encrypts and decrypts data using a set of two keys: a public key and a private key. The public key is used to encrypt the data, and the private key is used to decrypt it. This method is more secure than symmetric key encryption, despite being slower and requiring more computing power. In contrast, hash functions do not encrypt data using keys. Instead, a mathematical procedure is used to transform the data into a hash, which is a fixed-size value. Hash functions are often used to check the accuracy of data because a hash value will change if the data is changed [33].

Along with these encryption techniques, there are other protocols and standards that can be used to protect bioinformatics data. Using the Secure Sockets Layer (SSL) and Transport Layer Security (TLS) protocols, for example, data sent over the internet is frequently encrypted. The Health Insurance Portability and Accountability Act (HIPAA), a US law, establishes standards for the security of medical data, including bioinformatics data [34]. Protecting sensitive data requires HIPAA compliance, which calls for the implementation of suitable security measures like encryption.

In the research [33], a cryptographic and bioinformatics-based encryption and decryption technique is offered. In the suggested algorithm, a novel technique using RNA and deoxyribonucleic acid as keys is produced for safe data encryption and decryption procedures over a communication to conceal message from intruder or third party.

There are many different libraries that can be used for data encryption in bioinformatics. These libraries are designed for different programming languages and support different encryption algorithms. Examples could be:

- *pycrypto*: It is a library [35] designed for the Python language and includes various encryption algorithms. For example, it supports algorithms like AES, RSA, DES.
- *PyNaCl*: It is a library [36] designed for the Python language and includes modern encryption algorithms. For example, it supports algorithms like Curve25519, Salsa20, Poly1305.
- *Cryptography*: It is a library [37] designed for the Python language and includes various encryption algorithms. For example, it supports algorithms like AES, RSA, DES.
- *Bouncy Castle*: It is a library [38] designed for the Java language and includes various encryption algorithms. For example, it supports algorithms like AES, RSA, DES.
- OpenSSL: It is a library [39] designed for the C language and includes various encryption algorithms. For example, it supports algorithms like AES, RSA, DES.
- Crypto++: It is a library [40] designed for the C++ language and includes various encryption algorithms. For example, it supports algorithms like AES, RSA, DES.

These examples are just a few libraries you can use for data encryption, there are actually many more libraries available and each optimized for different programming languages and different encryption algorithms. Which library to use depends on factors such as the programming language you use and the encryption algorithm you need.

D. Role-based Access Control (RBAC)

RBAC is a technique for controlling access to computer or network resources based on the responsibilities of certain individuals within an organization [11]. To limit the activities users may perform inside a system, administrators can establish roles for users and give rights to those roles using RBAC. For instance, a system administrator may designate a "customer support" role with access to see and change client data but not to remove it. Users with the "customer support" position would then have the necessary access to carry out their job responsibilities, but they would not be able to make modifications that would jeopardize the security or integrity of the system. Access to resources and information may be managed effectively and flexibly with RBAC.

With role-based access control (RBAC), roles are allocated to users to decide what activities they may do on a system. For instance, in an accounting system, accountants could be able to issue invoices and manage payments but not carry out additional tasks like managing databases.

Role-Based Access Control (RBAC) is a commonly used method for controlling access to resources in a computer system, and there are many libraries and tools that can be used to implement the RBAC method for bioinformatics. These may include:

- *py-rbac*: For bioinformatics applications, this Python module offers a straightforward and adaptable framework for role-based access management [41].
- *BioPerl RBAC*: This gives users access to files, directories, and other resources in addition to a collection of Perl modules and an RBAC implementation for bioinformatics applications [42].
- *BioJava RBAC*: RBAC may be easily and adaptably implemented in web-based bioinformatics applications with the help of this Java package [43].

E. Differential Privacy Techniques

Differential privacy [12], [13], [44] is a statistical and data science concept that aims to secure the privacy of people whose data is being gathered and examined. It's a mathematical framework for calculating the level of privacy in a dataset and ensuring that analysis of the data doesn't adversely impact the privacy of the people whose data is included. In addition, it can be counted as a sub-topic of data anonymization techniques.

There are a number of different algorithms [45], [46], [47] that can be used to achieve differential privacy, each with its own strengths and weaknesses. Some of the most commonly used algorithms include:

- *Randomized response:* To hide the values of individual records, this algorithm [46] involves introducing random noise to the data. It is envisioned that the additional noise will make it challenging to determine the actual value of any given record while still enabling accurate analysis of the entire dataset.
- *Laplace mechanism:* This algorithm [45, 46] is similar to randomized response, but it introduces noise into the data that is based on the Laplace distribution rather than random noise. By doing this, it is assured that the extra noise is distributed in a way that protects the privacy of the people whose data is used.
- *Exponential mechanism:* This algorithm [45, 46] is a more advanced version of the Laplace mechanism, which takes into account the sensitivity of the data, as well as the desired level of privacy. It ensures that the added noise is distributed in a way that maximizes the privacy of the individuals whose data is included.
- *k-anonymity*: This algorithm [47] divides the data into "clusters" of k records, each of which has k records or more. This makes it difficult to link particular data to particular individuals because it guarantees that every individual record is a part of a group of at least k records.
- *Gaussian:* This algorithm [45] adds noise to the data based on the Gaussian distribution. The amount of noise added is determined by the standard deviation, which is chosen based on the desired level of privacy.

These algorithms can be used in a variety of different settings, including in surveys, medical research, and social media data analysis. While each algorithm has its own strengths and weaknesses, they are all designed to help protect the privacy of individuals whose data is being collected and analyzed.

IV. Advantage Of Known Data Security Techniques In Bioinformatics

Data security techniques in bioinformatics have many advantages. Some of these are:

- *Ensures the security of your data*: Your data is safeguarded against unwanted access, alteration, or destruction thanks to data security procedures. By doing this, you can retain the accuracy and integrity of your data while also ensuring its security.
- *Protects the privacy of your data:* By using data security strategies, you can keep your data private and prevent others from viewing it. This protects both the privacy of your data and your personal life.
- *Prevents data loss*: Data security measures guarantee that your data is backed up and prevent data loss. This enables you to recover your data in the event that it is lost or corrupted.
- *Reduces operating costs*: Data security methods lower operating expenses as well as the costs associated with data loss or data security breaches.
- *Increases customer trust*: Techniques for data security boost consumer confidence and allay customers' worries about your data. This enhances the reputation of your company and fosters more consumer loyalty.

V. Applications Of Known Data Security Techniques

A. Dataset

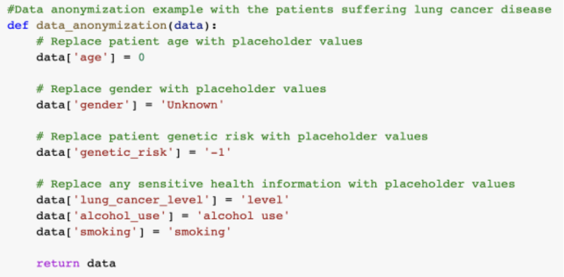
The sample dataset [48] is related to lung-cancer disease. There are 1000 patient records suffering from lung cancer disease. The dataset includes 25 columns such as patient age, smoking, level of lung cancer etc. All the attributes and values of the dataset are shown in Table I.

TABLE I. DATASET ATTRIBUTES

Column	Value
Patient ID	Numeric
Age	14-73
Gender	0: Male, 1: Female
Air Pollution	1-8
Alcohol Use	1-8
Dust Allergy	1-8
Occupational Hazard	1-8
Genetic Risk	1-8
Chronic Lung Disease	1-8
Balanced Diet	1-8
Obesity	1-8
Smoking	1-8
Passive Smoker	1-8
Chest Pain	1-8
Coughing of Blood	1-8
Fatigue	1-8
Weight Loss	1-8
Shortness of Breath	1-8
Wheezing	1-8
Swallowing Difficulty	1-8
Clubbing of Finger Nails	1-8
Frequent Cold	1-8
Dry Cough	1-8
Snoring	1-8
Lung Cancer Level	Low, Medium, High

B. Applications

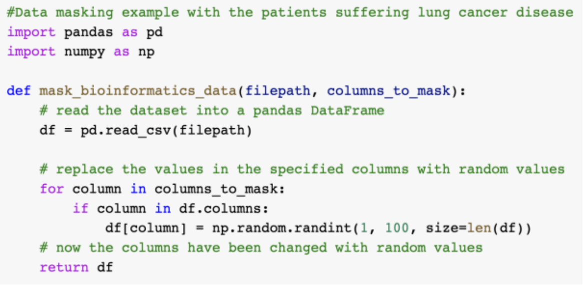
a) Data anonymization: There is an example of a simple function that takes a dataset as input and replaces certain identifiable information with placeholder values:



```
#Data anonymization example with the patients suffering lung cancer disease
def data_anonymization(data):
    # Replace patient age with placeholder values
    data['age'] = 0

    # Replace gender with placeholder values
    data['gender'] = 'Unknown'

    # Replace patient genetic risk with placeholder values
    data['genetic_risk'] = '-1'

    # Replace any sensitive health information with placeholder values
    data['lung_cancer_level'] = 'level'
    data['alcohol_use'] = 'alcohol use'
    data['smoking'] = 'smoking'

    return data
```

This function replaces patient age, gender, genetic risk, and any sensitive health information in the dataset with placeholder values.

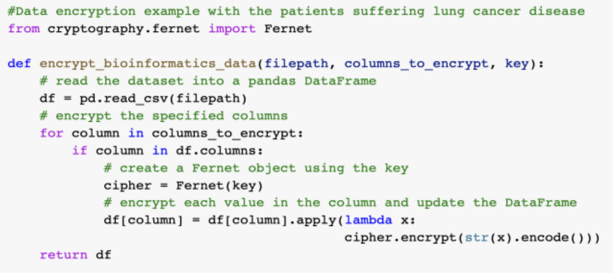
b) Data masking: There are a few libraries in Python that can be used to mask data in a bioinformatics dataset, such as pandas and NumPy. Here's an example of a function that masks the data in a bioinformatics dataset by replacing certain columns with random values:



```
#Data masking example with the patients suffering lung cancer disease
import pandas as pd
import numpy as np

def mask_bioinformatics_data(filepath, columns_to_mask):
    # read the dataset into a pandas DataFrame
    df = pd.read_csv(filepath)

    # replace the values in the specified columns with random values
    for column in columns_to_mask:
        if column in df.columns:
            df[column] = np.random.randint(1, 100, size=len(df))
    # now the columns have been changed with random values
    return df
```

This function can be used by providing the path to the bioinformatics dataset and a list of columns that you want to mask.

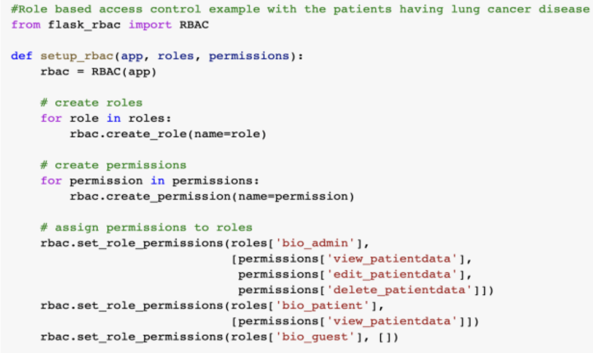
c) Data Encryption: One way to encrypt data in Python is to use the cryptography library. Here is an example of a function that encrypts the data in a bioinformatics dataset using the Advanced Encryption Standard (AES) algorithm:



```
#Data encryption example with the patients suffering lung cancer disease
from cryptography.fernet import Fernet

def encrypt_bioinformatics_data(filepath, columns_to_encrypt, key):
    # read the dataset into a pandas DataFrame
    df = pd.read_csv(filepath)
    # encrypt the specified columns
    for column in columns_to_encrypt:
        if column in df.columns:
            # create a Fernet object using the key
            cipher = Fernet(key)
            # encrypt each value in the column and update the DataFrame
            df[column] = df[column].apply(lambda x:
                                          cipher.encrypt(str(x).encode()))
    return df
```

This function can be used by providing the path to the bioinformatics dataset and a list of columns that you want to encrypt, and also a key should be provided that is used for the encryption. A key can be generated using the Fernet.generate_key() method which returns a URL-safe base64 encoded key. This key should be a secret and should be stored at a secure place, also it is significant to be careful when transferring the key to other parties.

d) Role-based Access Control (RBAC): In Python, the Flask-RBAC library can be used to implement RBAC in a web application that uses the Flask framework. Here is an example of a function that sets up RBAC for a bioinformatics dataset:

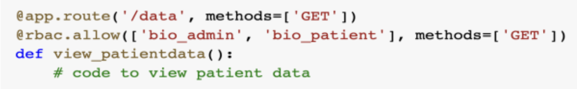


```
#Role based access control example with the patients having lung cancer disease
from flask_rbac import RBAC

def setup_rbac(app, roles, permissions):
    rbac = RBAC(app)

    # create roles
    for role in roles:
        rbac.create_role(name=role)

    # create permissions
    for permission in permissions:
        rbac.create_permission(name=permission)

    # assign permissions to roles
    rbac.set_role_permissions(roles['bio_admin'],
                              [permissions['view_patientdata'],
                               permissions['edit_patientdata'],
                               permissions['delete_patientdata']])
    rbac.set_role_permissions(roles['bio_patient'],
                              [permissions['view_patientdata']])
    rbac.set_role_permissions(roles['bio_guest'], [])
```

This function can be used in the flask applications by passing the flask app instance and list of roles it can be created along with the permission that each role will have.

The RBAC object can be used to create roles and permissions and to assign permissions to roles. In this example,there are three roles bio_admin, bio_patient and guest, and three permissions view_patientdata, edit_patientdata, and delete_patientdata. The bio_admin role has all the permissions, the bio_patient role has the view_patientdata permission, and the guest role has no permissions.

The RBAC object can be used in the views to protect routes and check patient permissions,

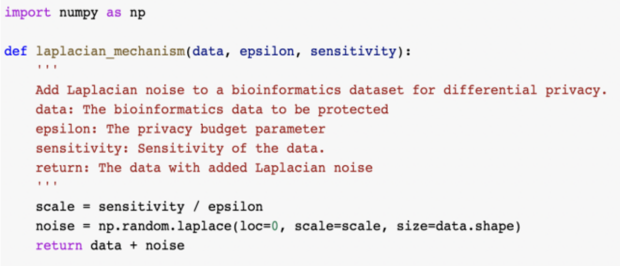


```
@app.route('/data', methods=['GET'])
@rbac.allow(['bio_admin', 'bio_patient'], methods=['GET'])
def view_patientdata():
    # code to view patient data
```

In this example, only the bio_patient with bio_admin and patient role will be able to see the data. This is just a simple example of how RBAC can be set up for a bioinformatics dataset using the Flask-RBAC library. In a real-world scenario, this function can be customized to suit the specific needs of the applications, such as adding or removing roles and permissions, and integrating it with the authentication and authorization system.

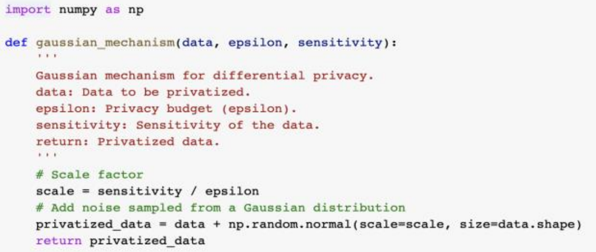
e) Differential Privacy Techniques: Lapcian, Gaussian, and Exponential methods are built in the Python programming language. After the construction of the algorithms, “Age” feature is selected from the dataset [48] to be anonymized with the help of these algorithms. In conclusion, these algorithms are compared with some comparison parameters such as privacy guarantee, accuracy, noise, sensitivity, and robustness.

The Laplacian function applied in this study is as follows.



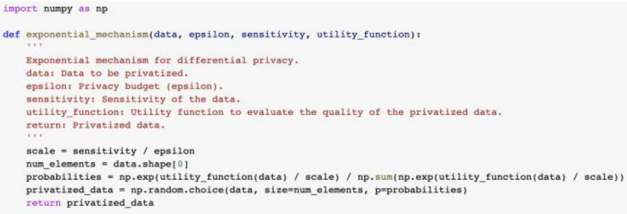
```
import numpy as np

def laplacian_mechanism(data, epsilon, sensitivity):
    '''
    Add Laplacian noise to a bioinformatics dataset for differential privacy.
    data: The bioinformatics data to be protected
    epsilon: The privacy budget parameter
    sensitivity: Sensitivity of the data.
    return: The data with added Laplacian noise
    '''
    scale = sensitivity / epsilon
    noise = np.random.laplace(loc=0, scale=scale, size=data.shape)
    return data + noise
```

The Gaussian function applied in this study is as follows.



```
import numpy as np

def gaussian_mechanism(data, epsilon, sensitivity):
    '''
    Gaussian mechanism for differential privacy.
    data: Data to be privatized.
    epsilon: Privacy budget (epsilon).
    sensitivity: Sensitivity of the data.
    return: Privatized data.
    '''
    # Scale factor
    scale = sensitivity / epsilon
    # Add noise sampled from a Gaussian distribution
    privatized_data = data + np.random.normal(scale=scale, size=data.shape)
    return privatized_data
```

The Exponential function applied in this study is as follows.



```
import numpy as np

def exponential_mechanism(data, epsilon, sensitivity, utility_function):
    '''
    Exponential mechanism for differential privacy.
    data: Data to be privatized.
    epsilon: Privacy budget (epsilon).
    sensitivity: Sensitivity of the data.
    utility_function: Utility function to evaluate the quality of the privatized data.
    return: Privatized data.
    '''
    scale = sensitivity / epsilon
    num_elements = data.shape[0]
    probabilities = np.exp(utility_function(data) / scale) / np.sum(np.exp(utility_function(data) / scale))
    privatized_data = np.random.choice(data, size=num_elements, p=probabilities)
    return privatized_data
```

The above algorithms are compared using criteria including privacy guarantee, accuracy, noise, sensitivity, and robustness. In this part of the study, *NumPy* [49] library is used to apply the comparison parameters. The details of these comparison parameters are:

Privacy guarantee: Compare how much information about specific data points is revealed by each method. The method with the lowest information leak would be considered the best in terms of ensuring privacy.

Accuracy: Compare how accurately each technique's results were produced. The most accurate method would be the one that yields the best results in terms of accuracy.

Noise addition: Compare the amount of noise added to the data by each technique. Regarding noise addition, the method that introduces the least amount of noise would be deemed to be the best.

Sensitivity: Compare how much of a change in the data each technique will be able to detect. In terms of sensitivity, the most sensitive method would be regarded as the best.

Robustness: Compare the ability of each technique to maintain privacy guarantees when the data distribution is unknown or varies. The technique that is the most robust would be considered the best in terms of robustness.

In the following table, the decimal values of each comparison parameter are shown after applying the above algorithms for “Age” attribute.

TABLE 2. COMPARISON TABLE OF DIFFERENTIAL PRIVACY TECHNIQUES

Privacy Technique	**Privacy Guarantee**	**Accuracy**	**Noise**	**Sensitivity**	**Robustness**
Laplacian	9.6041	9.6041	-0.5029	30.0	10.8223
Gaussian	8.0728	8.0728	0.3407	30.0	8.1036
Exponential	10.2430	10.2430	10.2430	30.0	9.7274

According to the results given in Table 2, Fig 1 represents the graph of the results.

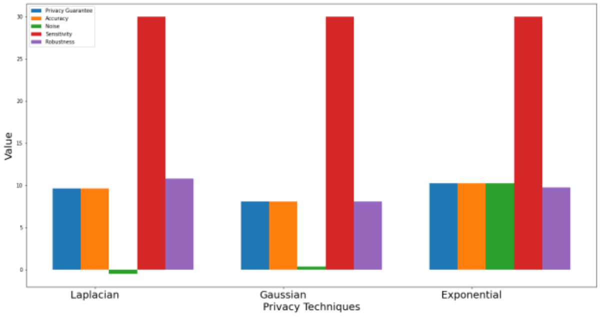


Fig. 1 Graph of the comparison parameters such as privacy guarantee, accuracy, sensitivity, and robustness

The results of analysis of each algorithms are given as follows:

- *Privacy guarantee:* The results you obtained indicate how well each technique is able to protect the privacy of the data while preserving its utility. The privacy guarantee measures how much privacy is preserved by each technique, where a higher value means that the data is more protected. The Gaussian technique has a lower privacy guarantee than the Laplacian and Exponential techniques, which means that it provides less privacy.

- *Accuracy:* The accuracy measures how well the data is preserved after applying the technique, where a higher value means that the data is more accurate. The Laplacian and Exponential techniques have a higher accuracy than the Gaussian technique, which means that they preserve more of the original data.

- *Noise addition:* The noise measures how much noise has been added to the data, where a higher value means that more noise has been added. The Laplacian technique has added less noise than the Gaussian and Exponential techniques, which means that it preserves more of the original data.

- *Sensitivity:* The sensitivity measures how sensitive the data is to the addition of noise, where a higher value means that the data is more sensitive. The three techniques have the same sensitivity, which means that the data is equally sensitive to the addition of noise.

- *Robustness:* The robustness measures how much the data has changed as a result of applying the technique, where a higher value means that the data has changed more. The Laplacian technique has a higher robustness than the Gaussian technique, but the Exponential has higher robustness than Laplacian.

Based on these results, it can be concluded that the Laplacian technique provides the best balance between privacy and utility, as it provides the highest privacy guarantee and accuracy, and the lowest noise and robustness. However, the choice of technique will depend on the specific requirements of the use case and the trade-off between privacy and utility that is acceptable.

VI. Result

In this study, five known data security techniques in bioinformatics were examined and some libraries related to data security are explained. With the help of some of these libraries, some security functions such as data_anonymization(), mask_bioinformatics_data(), encrypt_bioinformatics_data(), and setup_rbac() were builded in Python programming language. These methods were applied for a dataset [48] including 1000 patients who have lung cancer disease.

In order to prevent re-identification of individuals, data anonymization removes personal identifiers from a dataset, such as age, gender, genetic risk, etc. Personal identifiers are typically either removed entirely from the dataset or replaced with placeholder values to accomplish this. Data anonymization aims to render re-identification of individuals based on the remaining data extremely challenging, if not impossible.

While personal identifiers can still be used for analysis or research, the risk of re-identification is significantly reduced when they are "masked" or obscured. Personal identifiers are typically replaced with random values or symbols, such as the asterisk (*), to perform masking. Masking aims to strike a balance between the need for data access and the need to safeguard individual privacy.

It is difficult to compare the results of these two techniques because it depends on the dataset, the required level of security, the particular identifiable information, and other factors. In terms of privacy protection, data anonymization is regarded as being more effective than masking because it completely removes personally identifiable information. However, because some of the data may be lost or replaced with placeholder values, anonymization can also reduce the dataset's usefulness for particular research or analysis purposes. However, masking can still offer a degree of privacy protection while allowing the data to be used in more ways.

In addition to other data security techniques, Laplacian, Gaussian, and Exponential algorithms which are some of the differential privacy techniques are applied to the lung-cancer dataset [48]. As a sample feature, “Age” attribute is anonymized with these algorithms. As a result of the comparison parameters such as privacy guarantee, accuracy, noise, sensitivity, and robustness, it can be concluded that The Laplacian technique offers the highest privacy guarantee (9.6041) and accuracy (9.6041), the lowest noise (-0.5029) and robustness (10.8223), and the best balance between privacy and utility. However, the technique chosen will depend on the particular needs of the use case and the acceptable trade-off between privacy and utility.

It's crucial to remember that neither anonymization nor masking by themselves can always guarantee privacy protection. To further prevent unauthorized access to the data, it is crucial to implement additional security measures such as access controls and monitoring even after using one of these techniques. For future work, with the advancement of data security techniques, new security models can be constructed in addition to the mentioned techniques.

VII. References

[1] Y. Sakakibara, “Grammatical Inference in Bioinformatics”, IEEE Transactions on Pattern Analysıs and Machine Intelligence, Vol. 27, No. 7, July 2005.

[2] L. S. Heath , N. Ramakrishnan, “The emerging landscape of bioinformatics software systems”, Computer, https://doi.org/10.1109/mc.2002.1016900, 2002.

[3] D. Kaloudas, N. Pavlova, R. Penchovsky, “EBWS: Essential Bioinformatics Web Services for Sequence Analyses”, IEEE/ACM

Transactions on Computational Biology and Bioinformatics, Vol. 16, No. 3, 2019.

[4] N. Rapin, C. Kesmir, S. Frankild, M. Nielsen, C. Lundegaard, S. Brunak, O. Lund, “Modelling the Human Immune System by Combining Bioinformatics and Systems Biology Approaches”, Journal of Biological Physics, 32: 335–353, 2006.

[5] M. Thomas, A. Daemen, B. DeMoor, “Maximum Likelihood Estimation of GEVD: Applications in Bioinformatics”. IEEE/ACM Transactions on Computational Biology and Bioinformatics, Vol. 11, No. 4, 2014.

[6] M. Armstrong, J. Thomas, B. Henson, A. Kirby, M. Galloway, “Bioinformatics Cloud Security”, 2019 IEEE Cloud Summit. https://doi.org/10.1109/cloudsummit47114.2019.00018, 2019.

[7] A. Tamersoy, G. Loukides, M. ErcanNergiz, Y. Saygin, B. Malin, “Anonymization of Longitudinal Electronic Medical Records”, Vol. 16, No. 3, 2012.

[8] G. Loukides, A. Gkoulalas-Divanis, “Utility-Aware Anonymization of Diagnosis Codes”, IEEE Journal of Biomedical and Health Informatics, Vol. 17, No. 1, 2013.

[9] J.-X. WEI, M.-H. LIU, Z.-Q. LU, J. Wang, S. CHEN, Y. LAN, G.-Z. FENG, “Minimization of masking in signal detection from Chinese spontaneous reporting databases based on data removal strategy”, 2020 13th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), IEEE, 2020.

[10] N. Dowlin, R. Gilad-Bachrach, K. Laine, K. Lauter, M. Naehrig, J. Wernsing, “Manual for Using Homomorphic Encryption for Bioinformatics”, Proceedings of the IEEE, https://doi.org/10.1109/jproc.2016.2622218, 2017.

[11] D. Shin, G-J. Ahn, J. S. Park, “An application of directory service markup language (DSML) for role-based access control (RBAC)”, Proceedings 26th Annual International Computer Software and Applications, 2002.

[12] C. Dwork, “Differential privacy,” in Proc. 33rd Int. Colloquium on Automata, pp. 1–12, 2006.

[13] J. L. Raisaro *et al*., "Protecting Privacy and Security of Genomic Data in i2b2 with Homomorphic Encryption and Differential Privacy," in *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 15, no. 5, pp. 1413-1426, 1 Sept.-Oct. 2018.

[14] J. M. Struble, P. Handke, R. T. Gill, “Genome Sequence Databases: Genomic, Construction of Libraries”, Editor(s): Moselio Schaechter, Encyclopedia of Microbiology (Third Edition), Academic Press, Pages 185-195, ISBN 9780123739445, 2009.

[15] A. Schmidt, I. Forne, A. Imhof, “Bioinformatic analysis of proteomics data”, BMC Syst Biol 8 (Suppl 2), S3, 2014.

[16] M. D. Sorani, J. K. Yue, S. Sharma, G. T. Manley, A. R. Ferguson, “Genetic data sharing and privacy. Neuroinformatics”, doi: 10.1007/s12021-014-9248-z. PMID: 25326433; PMCID: PMC5718357, 2015 Jan;13(1):1-6.

[17] T. King, L. Brankovic, P. Gillard, “Perspectives of Australian adults about protecting the privacy of their health information in statistical databases”, International Journal of Medical Informatics, 81(4):279–289, 2012.

[18] J. Gauthier, A. T. Vincent, S. J. Charette, N. Derome, “A brief history of bioinformatics, Briefings in Bioinformatics”, Volume 20, Issue 6, Pages 1981–1996, 2019.

[19] R. A. Irizarry, B. Hobbs, F. Collin, Y. D. Beazer-Barclay, K. J. Antonellis,, U. Scherf, T. P. Speed, “Exploration, normalization, and summaries of high density oligonucleotide array probe level data”, Biostatistics, Apr;4(2):249-64, 2003.

[20] A. Harbola, D. Negi, M. Manchanda, R. K. Kesharwani, “Bioinformatics and biological data mining”, Editor(s): Dev Bukhsh Singh, Rajesh Kumar Pathak, Bioinformatics, Chapter 27, Pages 457-471, ISBN 9780323897754., Academic Press, 2022.

[21] E. Abrahams, G. S. Ginsburg, M. Silver, “The Personalized Medicine Coalition: goals and strategies”, Am J Pharmacogenomics., 5(6):345-55, 2005.

[22] H. Han, W. Liu, “The coming era of artificial intelligence in biological data science”, BMC Bioinformatics 2019, 20(Suppl 22):712, China. 22-24, June 2019.

[23] M. Krassowski, V. Das, S. K. Sahu, B. B. Misra, “State of the Field in Multi-Omics Research: From Computational Needs to Data Mining and Sharing” Front Genet, 10;11:610798, 2020.

[24] R. Lowe, N. Shirley, M. Bleackley, S. Dolan, T. Shafee, “Transcriptomics technologies”, Plos Computational Biology, 13(5), 2017.

[25] D. Goodman, C. O. Johnson, D. Bowen, M. Smith, L. Wenzel, K. Edwards, “De-identified genomic data sharing: the research participant perspective”, Journal of community genetics, 8(3), 173–181, 2017.

[26] https://pypi.org/project/pynonymizer/, Pyanonymizer, 11.01.2023.

[27] https://pypi.org/project/Faker/0.7.4/, Faker, 11.01.2023.

[28] https://pypi.org/project/anon/, anon 0.0.1, 10.01.2023.

[29] https://data2knowledge.atlassian.net/wiki/spaces/DSDEV/pages/12943436/Session+2+SSCM+DataSHIELD+tutorial, SSCM DataSHIELD tutorial, 10.01.2023.

[30] https://amnesia.openaire.eu/, High accuracy Data Anonymization, 10.01.2023.

[31] https://techdocs.broadcom.com/us/en/ca-enterprise-software/devops/test-data-management/4-9/getting-started/getting-started-with-fast-data-masker.html, Getting Started with Fast Data Masker, 10.01.2023.

[32] https://www.datarobot.com/, DataRobot, 10.01.2023.

[33] V. Siddaramappa, K. B. Ramesh, "Cryptography and bioinformatics techniques for secure information transmission over insecure channels," 2015 International Conference on Applied and Theoretical Computing and Communication Technology, Davangere, pp. 137-139, India, 2015.

[34] https://www.hhs.gov/hipaa/for-professionals/privacy/index.html, The HIPAA Privacy Rule, 11.01.2023.

[35] https://pypi.org/project/pycrypto/, pycrypto 2.6, 10.01.2023.

[36] https://pypi.org/project/PyNaCl/, PyNaCl 1.5.0, 10.01.2023.

[37] https://pypi.org/project/cryptography/, cryptography 39.0.0, 10.01.2023.

[38] https://www.baeldung.com/java-bouncy-castle, Introduction to BouncyCastle with Java, 10.01.2023.

[39] https://wiki.openssl.org/index.php/Compilation_and_Installation, OpenSSL, 10.01.2023.

[40] https://www.cryptopp.com/, Crypto++® Library 8.7, 10.01.2023.

[41] https://pypi.org/project/py-rbac/, py-rbac 20.12.3, 10.01.2023.

[42] https://bioperl.org/, BioPerl, 10.01.2023.

[43] https://biojava.org/, BioJava , 10.01.2023.

[44] A. Dyda, M. Purcell, S. Curtis, E. Field, P. Pillai, K. Ricardo, H. Weng, J. C. Moore, M. Hewett, G. Williams, C. L. Lau, “Differential privacy for public health data: An innovative tool to optimize information sharing while protecting data confidentiality”, Patterns, Volume 2, Issue 12, 2021.

[45] M. U. Hassan, M. H. Rehmani, J. Chen, “Differential Privacy Techniques for Cyber Physical Systems: A Survey”, IEEE Communications Surveys & Tutorials, Vol. 22, No. 1, First Quarter 2020.

[46] K. C. Gadepally, S. Mangalampalli, “Effects of Noise on Machine Learning Algorithms Using Local Differential Privacy Techniques”, 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), Conference Paper, 2021.

[47] P. Samarati, L. Sweeney, “Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression”, Computer Science, 1998.

[48] https://data.world/cancerdatahp/lung-cancer-data/workspace/file?filename=cancer+patient+level%20data+sets.xlsx., Lung Cancer Data, 10.01.2023.

[49] https://numpy.org/, NumPy library, 10.01.2023.

Blockchain-based Privacy Preserving Linear Regression

Received: 30 January 2023; Accepted: 15 February 2023 ***Research Article***

Zeynep Delal Mutlu
Dept. of Computer Engineering,
TOBB University
Ankara, Turkey
zmutlu@etu.edu.tr

Yesem Kurt Peker
TSTS School of Computer Science,
Columbus State University
Columbus, GA, USA
peker_yesem@columbusstate.edu

Ali Aydın Selçuk
Dept. of Computer Engineering,
TOBB University
Ankara, Turkey
aselcuk@etu.edu.tr

***Abstract*—In this study we propose a blockchain-based architecture that uses smart contracts and homomorphic encryption to allow statistical computations on confidential data by third parties. The use of blockchain provides the much-desired security properties of integrity and fault tolerance and homomorphic encryption preserves the privacy of the data. We present the design, implementation, and testing of our system. Our results show that a blockchain-based data sharing mechanism with homomorphic calculations via a smart contract is feasible and provides improvements in protecting the data from unauthorized users. Even though our work focused on linear regression, the architecture can be used for other statistical analysis and machine learning algorithms.**

Keywords—blockchain, homomorphic encryption, statistics, linear regression, ethereum

I. INTRODUCTION

Blockchain is a decentralized digital ledger consisting of blocks that are chained together via cryptographic hash functions. First introduced in 2008 as a transaction ledger for Bitcoin [1], blockchain technology has found applications not only in finance but also in healthcare, supply management, the Internet of Things (IoT), and other areas. Tamper resistance and fault tolerance, combined with the ability to execute code on the blockchain (i.e. smart contracts), blockchain technology became a very attractive solution for applications that require high integrity, high availability, and auditability of data and transactions [2].

One security property that is not natively included in blockchain is confidentiality [3]. Data submitted to the blockchain as part of a transaction or a smart contract is open to everyone for viewing. Unless encrypted, the privacy requirements of data shared on the blockchain cannot be satisfied. There are several studies that explore the use of encryption on blockchain. One of the encryption methods proposed for use for the confidentiality of data on blockchain is homomorphic encryption [4]. Homomorphic encryption allows computations to be done on the encrypted data (ciphertext) without having to decrypt it.

This work proposes a prototype for a blockchain-based data-sharing system that uses smart contracts and homomorphic encryption to allow statistical computations on confidential data by third parties. The advantages of the proposed system over systems that do not use homomorphic encryption include protecting the confidentiality of data during processing in addition to transmission and storage.

Our contributions in this work are as follows:

1) Design a prototype system for sharing and analysis of aggregated data respecting the privacy of the data.
2) Implementation and evaluation of the prototype system using the Ethereum platform for calculating linear regression equation for data that consists of pairs of values.

The remainder of this paper is organized as follows. In Section 2, we review the literature on privacy-preserving data sharing and analysis methods on blockchain. In Section 3, we provide a brief background of the proposed system. In Section 4, the proposed system is explained in detail, along with the tools used and its implementation. In Section 5, the results from our experiments are shared. Results are discussed in the Conclusion section, and avenues for improvements are included in the Future Work section.

II. RELATED WORKS

Privacy preserving data analysis over blockchain technology has been the topic of study in many articles in recent years. In the research [5], a theoretical perspective on the application of homomorphic operations in smart contracts has been developed, and an exemplary architecture has been presented. The consistency of this theoretical work reveals that different homomorphic cryptosystems in blockchain systems can be applied. There are also studies showing that blockchain systems are used in projects where a large amount of data needs to be processed. One of these studies, the article [6], offers a perspective on how machine learning algorithm calculations can be made in blockchain systems.

Similar studies were also conducted for IoT applications. Reference [7] is one of the articles that encrypts IoT data with homomorphic encryption and sends it to the blockchain. They conducted a study on the homomorphic aggregation of IoT data. This study, which also schematized the system-wide data collection method, implemented the project on a private Ethereum blockchain.

Blockchain is considered a suitable system for electronic voting due to the principle of protecting data integrity. However, the privacy of the data is not protected in blockchain systems. Reference [8] proposes to count the votes by protecting the confidentiality of the data with homomorphic encryption. The study focuses on the chain node structure for their specific use case.

In [9], a system is proposed in order to securely calculate the count, mean, variant and skewness of private health data using Paillier homomorphic cryptosystem in the Hyperledger Fabric blockchain system. The system is implemented on a consortium blockchain where only parties that are part of the consortium are allowed on the blockchain. The authors develop a REST application for user requests to blockchain. According to their experimental results, requests are made within a reasonable time and security needs are met, hence their work is feasible in real life applications. Our work builds on this study. We also use the Paillier cryptosystem on a blockchain platform for secure data analysis. We expand the study to the public blockchain network Ethereum allowing a wider range of users to have access to the system. We also include linear regression in our implementation. Our results are also promising for practical applications in terms of time needed for data sharing and analysis.

Our research shows that linear regression calculation is not a statistical equation previously calculated using homomorphic encryption in the Ethereum blockchain. Therefore, our study differs from all studies in terms of linear regression calculation in blockchain systems. Although related works contains homomorphic cryptosystems, in many articles, instead of public blockchain systems, the private or consortium systems are used. Hence, our research differs from previous studies in terms of the public Ethereum network and linear regression calculation.

III. Background

A. Ethereum Blockchain and Smart Contract

Blockchain is a digital ledger consisting of blocks that are chained to each other. There are some characteristic properties of blockchain systems, such as decentralization, persistency, anonymity and auditability [2]. The blockchain structure is based on the decentralization principle, means there is not any central administration. Each node holds the record of all transactions. If one transaction is approved by the blockchain system that means it is recorded to all nodes. Recorded transactions cannot be changed because each block is connected to the next block with its own hash value. If one transaction is changed, the hash value is damaged, so the chain architecture is broken. This solid structure in the blockchain is called tamper resistance or persistency. Since transactions in the blockchain are recorded in nodes and cannot be changed, they can be traced back through the records in nodes. This provides auditability. In blockchain, all transactions are made with an address. Every user who wants to communicate with can operate with more than one address value. Since there is not any connection between the user's real identity and the address values, it is said that blockchain provides anonymity.

The Ethereum blockchain has a structure that allows application development and can run smart contract codes. In the white paper of Ethereum, the statement “Smart contracts, cryptographic "boxes" that contain value and only unlock it if certain conditions are met ...” is declared [10, p. 1]. A smart contract acts like an account in the blockchain and has a bunch of functions run when they are called. Different software languages can be used to code an Ethereum smart contract, but the Solidity software language is the most widely used. There are two different outputs of Solidity compiler; ABI and byte code which are explained in detail under the Implementation section.

B. Paillier Homomorphic Cryptosystem

Homomorphic encryption allows certain mathematical operations to be performed on encrypted data without compromising it. It is especially suitable in cases where the data is processed by third parties who are not trusted or not authorized to see the data. The output of homomorphic operations is the encrypted form of the resulting data. In this study, we use the Paillier homomorphic cryptosystem [11]. Paillier system allows addition on encrypted data without having to decrypt the data. This property is formulated as

$$E\ (m_1 + m_2) = \ E\ (m_1) \times E\ (m_2) \tag{1}$$

where E stands for encryption, m_1 and m_2 represent two plaintexts.

C. Linear Regression

Linear regression is a statistical data analysis technique that estimates the value of unknown data using known data values. It mathematically models the dependent variable in terms of the independent variables as a linear equation. The formulas for linear regression parameters are:

$$y = a_0 + a_1 x \tag{2}$$

$$a_0 = \bar{y} - a_1\bar{x}\ ,\ \bar{x} = \frac{\sum x_i}{n}\ ,\quad \bar{y} = \frac{\sum y_i}{n} \tag{3}$$

$$a_1 = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{n\sum {x_i}^2 - (\sum x_i)^2} \tag{4}$$

IV. Methodology

In this work, we propose a blockchain-based data-sharing system that uses smart contracts and homomorphic encryption to allow statistical computations on encrypted data. In this section, we describe the system, how the components of the system interact with each other, and our implementation of the system for linear regression.

A. Proposed System

Our proposed system allows third parties to do certain types of analysis on data that they are not authorized to view. To give some context to our work, we assume a scenario where distributed edge devices aggregate data from sensors. The edge devices keep the data in aggregated form. We assume a third party wants to do statistical analysis on the sensor data. Following the terminology in [9], we call the edge devices “data owners” and the third party a “researcher”. The other actor in our system is the “contract developer” who deploys smart contracts for researchers on the blockchain. The smart contracts gather the encrypted aggregated data from the data owners and get them ready for the researcher using homomorphic operations. We assume that the sensor data coming from each data owner consists of pairs (x_i, y_i), and the researcher is interested in the linear regression equation for the total sensor data. The formulas for linear regression parameters are provided in (3) and (4) in Section 3.

The sensor data is aggregated in distributed data owners connected to the blockchain. To illustrate how the proposed system works, we provide Figure 1, where four data owners and a researcher interact with the smart contract. In the figure, $x_{i,j}$ represents the x-coordinate of the jth data point from the ith data owner and $y_{i,j}$ represents the y -coordinate of the jth data point from the ith data owner. The sums are over finitely many data points, and data owners may have different numbers of data points; however, the limits are not included with the sums in the diagram for the sake of simplicity.

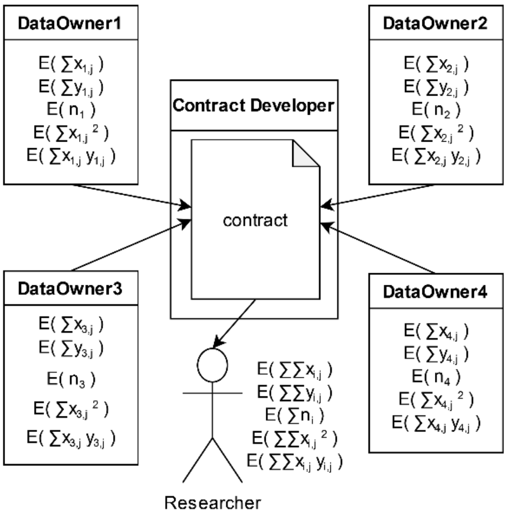


Figure-1: Data Flow in the Blockchain

The encrypted data sent by a data owner contains certain sums required in the calculation of the linear regression parameters. As seen in (3) and (4), five different sums are needed for the calculation of the parameters a_0 and a_1. These are sums of x's, y's, n's, x^2's and xy's. Each data owner encrypts the relevant sums of its data points and sends it to the smart contract. These encrypted sums are stored in the smart contract, and each type of encrypted sums are homomorphically added (i.e. multiplied) together to get the total encrypted sum for that type. For example, the encrypted sum of all x's for the four data owners is calculated as:

$$E\left(\sum_{j=1}^{n_1} x_{1,j}\right) \times E\left(\sum_{j=1}^{n_2} x_{2,j}\right) \times E\left(\sum_{j=1}^{n_3} x_{3,j}\right) \times E\left(\sum_{j=1}^{n_4} x_{4,j}\right)$$

The homomorphic property of Paillier ensures that the result of the calculation above is the same as the encrypted sum of all x's:

$$E\left(\sum_{j=1}^{n_1} x_{1,j} + \sum_{j=1}^{n_2} x_{2,j} + \sum_{j=1}^{n_3} x_{3,j} + \sum_{j=1}^{n_4} x_{4,j}\right)$$

More generally, for k data owners with n_i data points for data owner i, we have the equation:

$$E\left(\sum_{i=1}^{k}\sum_{j=1}^{n_i} x_{i,j}\right) = \prod_{i=1}^{k}\left(E\left(\sum_{j=1}^{n_i} x_{i,j}\right)\right)$$

After the encrypted sums are calculated on the blockchain, they are available for the researcher. Upon receiving the sums, the researcher decrypts the sums using their private key and plugs them into equations (3) and (4) to get the parameters for linear regression.

B. *Implementation*

We implemented the proposed system on the Ethereum test network Goerli using the Paillier cryptosystem. The tools we used to develop and interact with the smart contracts are:

- *Web3.0:* Also called Web3. It is a library used for interacting with the blockchain system from the outside.
- *Remix:* To develop and deploy the smart contract, the online Remix platform is chosen.
- *Metamask:* This is the wallet we use to connect to Ethereum blockchain. Metamask account was introduced on Remix to deploy the smart contract.
- *Visual Studio Code:* It is used for developing user JavaScript applications.

We provide an overview of how the three actors in our system, namely the smart contract developer, data owner, and researcher, interact with the blockchain in Figure 2. The contract developer codes the smart contract in Solidity and deploys it to the blockchain using the Web3 library. The compilation of the smart contract produces the Application Binary Interface (ABI) for the contract, which allows other users to interact with the smart contract. In our implementation, we use JavaScript to develop the scripts to interact with the smart contract.

Because the Paillier Cryptosystem needs calculations with big integers, both smart contract and user applications need big integer libraries. In our implementation, we used BigNumbers.sol library [12] for the smart contract and BigInteger.js library [13] for the scripts used in the data owner and researcher applications.

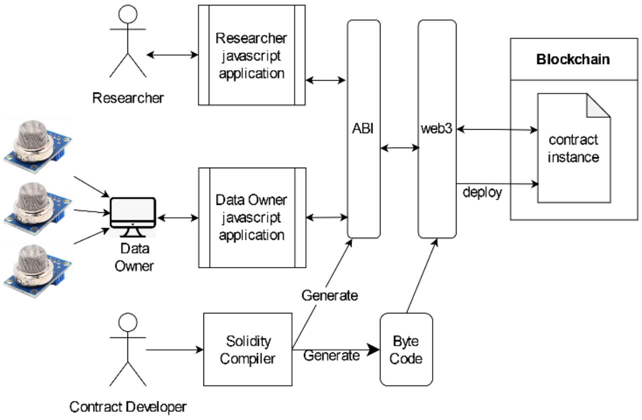


Fig. 2. Overview of interactions of actors with the smart contract

A sequence diagram depicting the flow in the proposed system is given in Figure 3. The diagram includes a smart contract that is already deployed on the blockchain to facilitate the sharing of the data between the researcher and the data owners. In the diagram , there is a researcher who wants to do linear regression analysis on the data (using equation (2) in section III.C), and a data owner who provides the data to be used in this analysis.

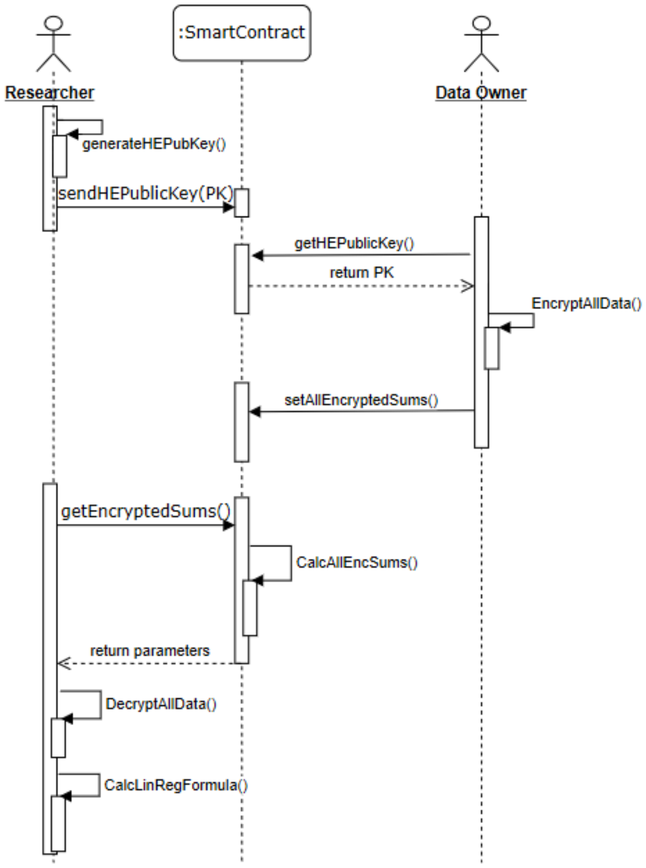


Fig. 3. Overview of the flow of data sharing

According to the flow of data in Figure 3, the researcher generates the homomorphic key pairs and records the homomorphic public key in the smart contract. The data owner gets this public key from the smart contract and encrypts the sum of the data values used in linear regression parameter calculations with the researcher's public key and saves the results to the smart contract. Each data owner follows the same steps and sends its encrypted sums to the smart contract. The smart contract "adds" the encrypted sums it receives from the data owners to get the encrypted totals of all data points. The "add" operation is the homomorphic addition and, in the Paillier system, is indeed a modular multiplication operation. The pseudocode for the homomorphic addition operation on the smart contract is provided in Figure 4. The encrypted totals are what the researcher needs to calculate the parameters for linear regression analysis. Upon receiving the encrypted totals, the researcher decrypts them using their private key and plugs them in equation (3) and (4) to find the regression parameters.

The pseudocode of the homomorphic addition function to add the encrypted values stored in an array in the smart contract is as follows:

```
bytes[] arrayOfEncData
BigNumber data, nextData, n2

FUNCTION calculateEncSumofArray(){
        data = arrayOfEncData[0] in the type of BigNumber
        FOR i = 1 to length of the arrayOfEncData DO
                nextData = arrayOfEncData[i] in the
                        type of BigNumber
                data = modular multiplication of data and
                    nextData with respect to modulus $n^2$
        ENDFOR
        RETURN data
}
```

Fig. 4. Pseudocode of homomorphic addition in smart contract

V. Experimental Results

We tested our system with respect to the key length as well as the number of data owners. We recorded two timings:

Timing 1. The average time it takes data owners to encrypt their data and send them to the smart contract.

Timing 2. The time it takes for the smart contract to homomorphically add the encrypted sums and send the result to the researcher.

The timings are tabulated in Tables I-III for key sizes 256-bit, 512-bit, and 1024 bit, respectively. Our results show that, for a given key size, for each data owner it takes 74 to 87 seconds to encrypt and send the data to the smart contract, and seconds to do homomorphic additions on the encrypted data on the smart contract and send it to the researcher. The timings indicate that using the Ethereum blockchain for the proposed system is feasible in real life applications.

Table I. 256-bit key length

Number of Data Owners	4	8	12	16
Timing1(sec. msec)	80.068	85.441	79.846	86.107
Timing2 (seconds)	2.417	2.455	2.343	2.386

Table II. 512-bit key length

Number of Data Owners	4	8	12	16
Timing1 (sec. msec)	74.287	86.764	81.894	86.087
Timing2 (seconds)	2.300	2.387	3.023	3.072

Table III. 1024-bit key length

Number of Data Owners	4	8	12	16
Timing1 (sec. msec)	87.019	86.863	85.811	83.091
Timing2(seconds)	2.379	2.381	2.436	2.687

VI. Conclusion

In this study, we proposed a system based on blockchain technology with smart contracts and homomorphic encryption that allows third parties to do analysis on data that they are not authorized to view. We implemented the proposed system on the Ethereum platform and used the Paillier cryptosystem to allow third parties to do linear regression analysis on the data.

With only a bit over than a minute for data sharing for a data owner, our test results indicate that the system is feasible for real-life applications. The fact that it takes slightly over a minute to encrypt and send the encrypted data to the smart contract is not surprising because each sending of data is a transaction and transactions on the blockchain take time. In the case of Ethereum Goerli testnet, depending on the intensity of use and gas prices, a transaction can take 15-30 seconds or more. In our implementation, because each sum is sent as a separate transaction and 5 sums need to be sent by each data owner, the sending of data by an owner takes more than a minute. This can be improved in future implementations by having the necessary data sent by the owner at once.

The homomorphic calculations on the smart contract to find the total encrypted sums take only seconds making the system appealing for practical applications.

The use of blockchain technology brings the much desired security properties of fault-tolerance, tamper-resistance, and accountability to the system. In addition, the use of a public blockchain allows data owners to have an open system where interested third parties have easy access to data that they need for analysis. The most significant contribution of the study is that all parties have, following the principle of least privilege, the minimal access they need to the data: The data owner stores only the aggregated sums of the plain data; the contract stores only the encrypted sums of the data, and the third party ultimately gets the total sums of all data. On the blockchain network, all that is transmitted about the data is the encrypted sums from the data owners and the encrypted total sums never exposing any individual data points or sums in plain form.

VII. Future Works

The system can be extended to include other analysis on data including AI and machine learning algorithms. Paillier cryptosystem is a partially homomorphic system that allows for addition of plaintexts hence is not suitable for systems where more complex analysis involving other operations is needed. For those use cases, fully homomorphic algorithms over blockchain technology can be studied.

In our scenario, there is only one smart contract designed for one researcher. As an expanded scenario, more than one researcher can be added to the system. This can be done in one of these two ways:

1. A new smart contract is deployed when a researcher is interested in the data: For this, a manager contract can be created that accepts requests from researchers and creates a new smart contract for each researcher. The smart contract stores the public key of the researcher that it is created for.

2. One smart contract can serve both as the manager and the facilitator for responses to the requests from researchers. In this method the public keys of all researchers are stored on the same smart contract.

Depending on the application, each may have benefits: The first method has the advantage of separating the tasks of management and facilitation. If used for researchers interested in different types of data and analysis, the first method will allow for easy customization of the smart contract created for them. In cases where the system is used for researchers interested in similar types of analysis with data, then keeping all the functionality in the same smart contract may make managing the contract and requests easier. A method to manage the keys and facilitate the receiving of the data encrypted with the key will need to be included in the contract.

References

[1] S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” *SSRN Electronic Journal,* 2008.

[2] M. Krichen, M. Ammi, A. Mihoub, and M. Almutiq, “Blockchain for modern applications: A survey,” *Sensors,* vol. 22, no. 14, p. 5274, 2022.

[3] G. Hilary, “Blockchain: Security and confidentiality.,” *SSRN Electronic Journal,* 2018.

[4] H. Ç. Bozduman and E. Afacan, “Simulation of a homomorphic encryption system,” *Applied Mathematics and Nonlinear Sciences,* vol. 5, no. 1, pp. 479–484, 2020.

[5] C. Regueiro, I. Seco, S. de Diego, O. Lage, and L. Etxebarria, “Privacy-enhancing distributed protocol for data aggregation based on blockchain and homomorphic encryption,” *Information Processing & Management,* vol. 58, no. 6, p. 102745, 2021.

[6] A. Mitra, B. Bera, A. K. Das, S. S. Jamal, and I. You, “Impact on blockchain-based AI/ML-enabled big data analytics for Cognitive Internet of Things Environment,” *Computer Communications,* vol. 197, pp. 173–185, 2023.

[7] F. Loukil, C. Ghedira-Guegan, K. Boukadi, and A.-N. Benharkat, “Privacy-preserving IOT data aggregation based on blockchain and homomorphic encryption,” *Sensors,* vol. 21, no. 7, p. 2452, 2021.

[8] B. U. Umar, O. M. Olaniyi, D. O. Olajide, and E. M. Dogo, “Paillier cryptosystem based Chainnode for secure electronic voting,” *Frontiers in Blockchain,* vol. 5, 2022.

[9] M. Ghadamyari, "Privacy-Preserving Statistical Analysis of Health Data Using Paillier Homomorphic Encryption and Permissioned Blockchain," *Electronic Theses and Dissertations.* 8139.

[10] “Ethereum whitepaper,” *ethereum.org*. [Online]. Available: https://ethereum.org/en/whitepaper/. [Accessed: 29-Jan-2023].

[11] T. Sridokmai and S. Prakancharoen, “The homomorphic other property of Paillier cryptosystem,” *2015 International Conference on Science and Technology (TICST),* 2015.

[12] Firoorg, “Firoorg/solidity-bignumber: Full BigNumber library implementation for solidity.,” *GitHub*. [Online]. Available: https://github.com/firoorg/solidity-BigNumber. [Accessed: 29-Jan-2023].

[13] Peterolson, “Peterolson/BigInteger.js: An arbitrary length integer library for Javascript,” *GitHub*. [Online]. Available: https://github.com/peterolson/BigInteger.js. [Accessed: 29-Jan-2023].

Privacy and Data Security Assessment for IT Vendor Services - strategic approach for Vendor IT Services analysis under GDPR

Received: 8 January 2023; Accepted: 15 February 2023 ***Research Article***

1st Elissa Mollakuqe
Faculty of Information Sciences and Computer Engineering
Skopje, Republic of North Macedonia
elissamollakuqe@gmail.com
ORCID ID: 0000-0003-0508-105X

2nd Vesna Dimitrova
Faculty of Information Sciences and Computer Engineering
Skopje, Republic of North Macedonia
vesna.dimitrova@finki.ukim.mk,
ORCID ID: 0000-0003-4393-5589

Abstract— **The large loads in current systems, in terms of software and hardware, compel various institutions and organizations to purchase IT services such as software hosting, hardware and software infrastructure, and different equipment for data storage. All these requests for additional resources expose institutions and organizations to numerous risks that threaten privacy and data security. In order to provide secure services and prevent potential attacks, various institutions and organizations implement high standards to prevent data attacks and privacy violations. One of these standards is the GDPR (General Data Protection Regulation) - 2016/679, which has two versions: OJ L 119, 04.05.2016; cor. OJ L 127, 23.5.2018. This research identifies the largest services that are purchased, including the names of the sellers and the services. These services are classified into five categories based on the level of security they provide, which is controlled in terms of the harmonization between privacy and data security on the part of the service seller. The purpose of this paper is to emphasize the importance of GDPR in the selection of services purchased by both users and service providers.**

Keywords— data, privacy, security, GDPR, IT services, vendor

I. Introduction

The increasing demand for IT services, such as software hosting, hardware and software infrastructure, and data storage, has exposed organizations to numerous risks that threaten privacy and data security. To provide secure services and prevent potential attacks, institutions and organizations use high standards to prevent data attacks and privacy violations, such as GDPR. This research identifies the largest services purchased by organizations, including the names of the sellers and the services, and classifies them into five categories based on the level of security they provide. This classification considers the harmonization between privacy and data security on the part of the service seller, which has not been widely studied in previous research.

This study emphasizes the importance of GDPR in the selection of services purchased by both the user and service provider. The classification of services based on their security level provides valuable insights for organizations seeking to purchase secure IT services. This research contributes to the understanding of the current IT services market and can help organizations make informed decisions to protect their data privacy and security.

Privacy in IT vendor services according to General Data Protection Regulation

The General Data Protection Regulation (GDPR) is a European Union regulation that came into effect on May 25, 2018. GDPR is a set of rules designed to give individuals more control over their personal data, and to ensure that organizations take adequate measures to protect personal data from misuse, loss, unauthorized access, and disclosure. GDPR applies to all organizations that process personal data of EU citizens, regardless of where the organization is located.

The GDPR imposes strict obligations on organizations that process personal data, including IT vendors. IT vendors are entities that provide IT services, such as software hosting, hardware and software infrastructure, and data storage, to organizations. IT vendors that process personal data on behalf of their clients are considered processors under GDPR, and they are required to comply with GDPR.

IT vendors that process personal data on behalf of their clients are required to comply with GDPR's data protection principles, which include:

1. Lawfulness, fairness, and transparency: Personal data must be processed lawfully, fairly, and transparently [2].
2. Purpose limitation: Personal data must be collected for specified, explicit, and legitimate purposes [3].
3. Data minimization: Personal data must be adequate, relevant, and limited to what is necessary [3].
4. Accuracy: Personal data must be accurate and kept up to date.
5. Storage limitation: Personal data must be kept for no longer than necessary.
6. Integrity and confidentiality: Personal data must be processed in a manner that ensures appropriate security, including protection against unauthorized or unlawful processing and against accidental loss, destruction, or damage [4].

IT vendors that process personal data on behalf of their clients must also implement appropriate technical and organizational measures to ensure the security of personal data [5]. These measures should ensure the confidentiality, integrity, and availability of personal data, as well as the resilience of the systems and services processing the data.

IT vendors that process personal data on behalf of their clients must also assist their clients in fulfilling their GDPR obligations. This includes, for example, responding to data subject requests, reporting data breaches, and conducting data protection impact assessments.

A. *Classification of it vendor services based on security level*

In this research, the largest IT vendor services purchased by organizations were identified, and they were classified into five categories based on the level of security they provide:

1. Low-security services: These services are characterized by low levels of security, and they include basic data storage and web hosting services. Examples of low-security service providers include Dropbox and Google Drive [6].
2. Medium-security services: These services are characterized by moderate levels of security, and they include cloud-based software services and email hosting services. Examples of medium-security service providers include Microsoft Office 365 and Google Workspace.
3. High-security services: These services are characterized by high levels of security, and they include cloud-based backup services and dedicated hosting services [7]. Examples of high-security service providers include Amazon Web Services and Rackspace.
4. Very high-security services: These services are characterized by very high levels of security, and they include services that comply with specific security standards, such as ISO 27001[8]. Examples of very high-security service providers include IBM Cloud and Microsoft Azure.
5. Customized security services: These services are characterized by customized levels of security, which are tailored to the specific needs of the organization [9]. Examples of customized security service providers include managed security service providers (MSSPs) and cybersecurity consulting firms.

The increasing demand for IT services has exposed organizations to numerous risks that threaten privacy and data security.

II. Data security in IT vendor services according to General Data Protection Regulation

Selection of IT Vendor Services and GDPR Compliance Organizations must carefully evaluate the security level of IT vendor services before purchasing them, and GDPR compliance should be a critical factor in the selection process. Organizations should consider the following factors when selecting IT vendor services: [1].

1. Data security and privacy policies: Organizations should review IT vendors' data security and privacy policies to ensure that they comply with GDPR requirements.
2. Technical and organizational measures: Organizations should evaluate the technical and organizational measures that IT vendors have implemented to ensure the security of personal data [12].
3. Data processing agreements: Organizations should ensure that they have a data processing agreement (DPA) with IT vendors that process personal data on their behalf. DPAs should include GDPR-mandated contractual clauses that specify the rights and obligations of both the organization and the IT vendor with respect to GDPR compliance.
4. Incident response and breach notification procedures [12]: Organizations should ensure that IT vendors have appropriate incident response and breach notification procedures in place to address security incidents and data breaches in a timely and effective manner.
5. Data protection impact assessments: Organizations should ensure that IT vendors are willing and able to assist them in conducting data protection impact assessments (DPIAs), which are required under GDPR when processing activities are likely to result in a high risk to the rights and freedoms of data subjects.

Challenges in Ensuring GDPR Compliance in IT Vendor Services Ensuring GDPR compliance in IT vendor services can be challenging for organizations, especially in cases where vendors are located in different countries or have complex data processing activities [11]. The following are some of the challenges that organizations may face in ensuring GDPR compliance in IT vendor services:

1. Jurisdictional issues: Organizations may face difficulties in ensuring GDPR [13] compliance when IT vendors are located in different countries or operate in multiple jurisdictions.
2. Sub-processing activities: IT vendors may subcontract data processing activities to third-party vendors, which can complicate GDPR compliance for organizations.
3. Data transfer restrictions: GDPR restricts the transfer of personal data outside the European Economic Area (EEA) to countries that do not have adequate data protection laws [10]. Organizations must ensure that their IT vendors comply with these restrictions when transferring personal data.
4. Lack of clarity in contractual terms: Organizations may face difficulties in ensuring GDPR compliance when contractual terms with IT vendors are unclear or do not adequately specify GDPR compliance obligations [10].
5. Inadequate data protection impact assessments: [14] Organizations may face difficulties in conducting adequate data protection impact assessments when IT vendors are not willing or able to assist them in this process.

The increasing demand for IT vendor services has made organizations more vulnerable to data attacks and privacy violations. GDPR compliance is essential for IT vendors that process personal data on behalf of their clients, and organizations should carefully evaluate the security level of IT vendor services before purchasing them.

III. Classification Of It Vendor Services Based On The Form Of Receiving The Service

The classification of IT vendor services based on the form of receiving the service can help institutions and organizations choose the most appropriate IT services for their needs while minimizing risks related to data privacy and security.

The increasing use of software and hardware in current systems has resulted in many institutions and organizations buying IT services, including software hosting, hardware and software infrastructure, and data storage equipment. However, these requests for additional resources expose institutions and organizations to numerous risks related to privacy and data security.

To prevent potential attacks and ensure secure services, various institutions and organizations use high standards to prevent data attacks and privacy violations, such as GDPR - 2016/679 - General Data Protection Regulation. GDPR is a regulation that provides a harmonized approach to data privacy and security [15] in the European Union (EU) and the European Economic Area (EEA) [16].

By emphasizing the importance of GDPR in the choice of IT services, this research highlights the need for institutions and organizations to choose services that prioritize data privacy and security. The classification of IT vendor services based on the form of receiving the service can help institutions and organizations make informed decisions about which IT services to use, while minimizing risks related to data privacy and security.

In our analysis of 47 companies and institutions, consisting of 11 public institutions and 36 private organizations, we have examined the different forms in which they receive IT vendor services. Our findings indicate that there is a wide range of approaches taken by these companies when it comes to acquiring vendor IT services.

Of the 47 companies analyzed, 28 reported that they opt for on-premise IT services, meaning that they acquire and manage IT infrastructure and applications in-house. Meanwhile, 15 companies have chosen to adopt cloud-based IT services, which allows them to access IT resources over the internet rather than managing them on-premise. The remaining 4 companies have adopted a hybrid model, which combines both on-premise and cloud-based IT services.

It is worth noting that public institutions are more likely to adopt on-premise IT services, with 8 out of 11 public institutions indicating that they manage their IT infrastructure and applications in-house. Private organizations, on the other hand, have a more even split between on-premise and cloud-based IT services, with 15 private organizations opting for on-premise services and 14 choosing cloud-based services.

Our analysis also revealed that the decision to adopt a particular form of IT vendor service is influenced by a number of factors, including cost, scalability, and security. While some companies may opt for on-premise services to have greater control over their IT infrastructure, others may choose cloud-based services to reduce costs and increase flexibility. On table 1. represents the number of institutions for the years 2023, 2022, and 2021 according to using or not acquire ongoing vendor IT services (e.g., application software hosting, hardware/software infrastructure, data storage facilities, staffing, etc.

TABLE I. The number of institutions for the years 2023, 2022 and 2021

	2023		2022		2021	
	public	private	public	private	public	private
YES	5	29	3	24	1	22
NO	0	13	2	18	1	23

Over the past three years (2021-2023), we collected data of 47 institutions, including both public and private organizations, to determine whether they opted to acquire ongoing vendor IT services for their projects. Our findings show that there has been a steady increase in the number of institutions that have chosen to acquire vendor IT services over this period.

In 2021, 23 out of 47 institutions (48%) reported that they had opted to acquire ongoing vendor IT services for their projects. This number increased to 27 institutions (58%) in 2022, and further to 34 institutions (72%) in 2023.

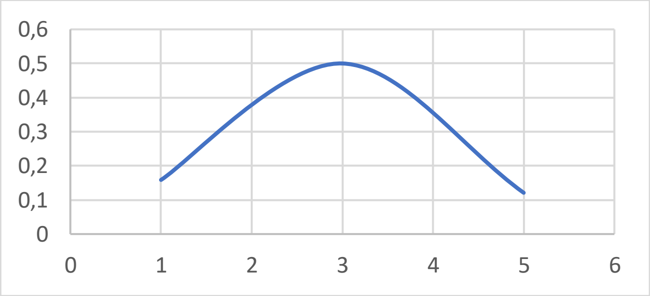


Fig. 1. Standard deviation for 2021, 2022 and 2022 for public institutions that use IT vendor services

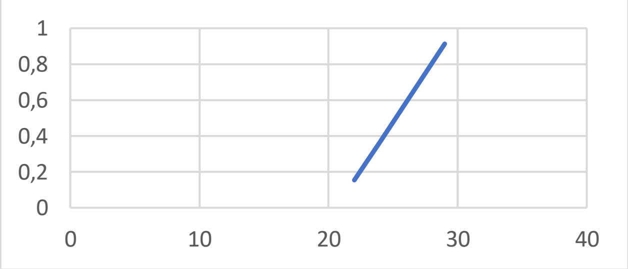


Fig. 2. Standard deviation for 2021, 2022 and 2022 for private institutions that use IT vendor services

Regarding the evaluation level of IT services (e.g., application software hosting, hardware/software infrastructure, data storage facilities, staffing, etc.), as a result of the standard deviation we conclude that during the year 2022 we have a higher spread or variability in public and private institutions of the use of IT services bought.

Regarding the evaluation of The vendor service(s) will be acquired including Request For Proposal -RFP, Sole Source Procurement -SSP, Purchase Order -PO, Agreement to vendor's online license user agreement - AVOLUA, Other - O, the results for the years 2021, 2022 and 2023 are presented in table 2.

TABLE II. THE VENDOR SERVICE(S) WILL BE ACQUIRED IN YEARS

	2023		2022		2021	
	public	private	public	private	public	Private
RFP	6	36	3	33	4	33
SSP	7	11	5	7	4	2
PO	9	36	9	36	7	34
AVOLUA	11	19	9	19	11	19

Of the institutions that have opted to acquire vendor IT services, the most commonly cited reasons for doing so were cost savings, increased efficiency, and access to specialized expertise. In contrast, the institutions that chose to rely on in-house IT resources cited concerns around data security and the need for greater control over their IT infrastructure.

Request for Proposal (RFP): An RFP typically includes information about the project or service, desired outcomes, requirements, and timelines. It is often used for complex projects or services where multiple suppliers may have the capability to deliver the required outcome.

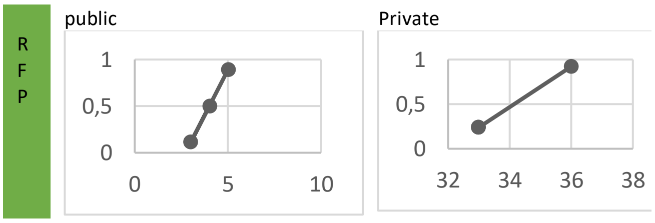


Fig. 3. . Standard deviation for Request for proposal

Based on the standard deviation figure 3. we can conclude that during the year 2023 we have the biggest variation of *Request For Proposal* in public institutions (standard deviation = 0.816 also in private companies the biggest variation is in 2023 (standard deviation = 1.414).

Sole Source Procurement (SSP): SSP is a procurement method used when only one supplier is capable of delivering a specific good or service, or when there are no other suppliers available. This method is usually used for purchases that are of low value or where the procurement process would be lengthy and costly if conducted via other methods.

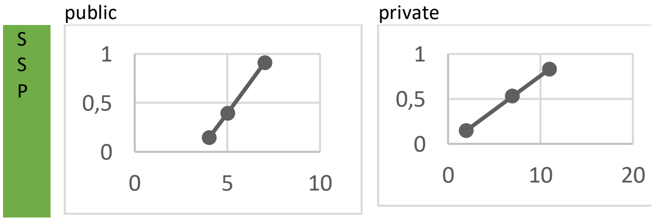


Fig. 4. Standard deviation for Sole Source Procurement

Based on the standard deviation figure 4. we can conclude that during the year 2023 we have the biggest variation of *Sole Source Procurement* in public institutions (standard deviation = 1.247 also in private companies the biggest variation is in 2023 (standard deviation = 4.505).

Purchase Order (PO): A PO is a document issued by the buyer to the supplier indicating the goods or services to be purchased, the quantity, the agreed price, and the delivery date. A PO serves as a legally binding contract between the buyer and supplier, and can be used as a means of tracking and managing the procurement process.

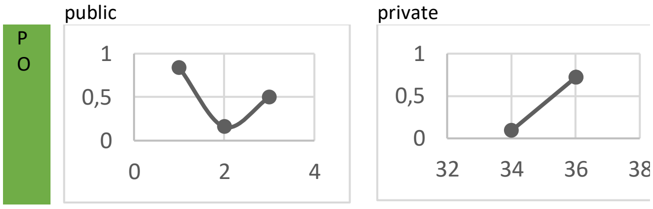


Fig. 5. . Standard deviation Purchase Order

Based on the standard deviation figure 5. we can conclude that during the year 2023 we have the biggest variation of *Purchase Order* in public institutions (standard deviation = 1) also in private companies the biggest variation is in 2023 (standard deviation = 1.55).

Agreement to Vendor's Online License User Agreement (AVOLUA): AVOLUA is a contract agreement between the vendor and the buyer that outlines the terms and conditions for the use of a specific product or service. This type of agreement is commonly used for software and online services, and may include details such as licensing fees, usage restrictions, and support terms.

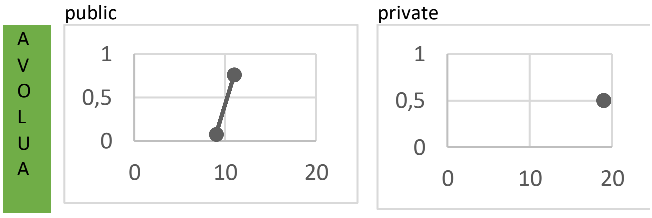


Fig. 6. Standard deviation Agreement to vendor's online license user agreement

Based on the standard deviation figure 6. we can conclude that during the year 2023 we have the biggest variation of *Standard deviation Agreement to vendor's online license user agreement* in public institutions (standard deviation = 0.9) but in private companies there in no any variation in years (standard deviation = 0).

A. The Largest IT services that are purchased, including the names of the sellers and the services

The largest IT services that are purchased can vary depending on the specific needs of organizations and industries. However, based on recent reports and industry forecasts, here are some of the largest IT services that are purchased, including the names of the sellers and the services they offer, for 2021, 2022, and 2023:

2021:

- 1.Cloud computing services - Amazon Web Services, Microsoft Azure, Google Cloud Platform, IBM Cloud
- 2. IT consulting services - Deloitte, Accenture, PwC, KPMG
- 3.Cybersecurity services - IBM Security, Symantec, McAfee, Check Point
- 4.Software development services - IBM, Accenture, Capgemini, Wipro, Infosys

- 5. Enterprise resource planning (ERP) services - SAP, Oracle, Microsoft Dynamics, Infor

2022:

- 1.Cloud computing services - Amazon Web Services, Microsoft Azure, Google Cloud Platform, IBM Cloud
- 2. IT consulting services - Deloitte, Accenture, PwC, KPMG
- 3. Artificial intelligence and machine learning services - IBM, Accenture, Deloitte, Capgemini, Wipro, Infosys
- 3. Cybersecurity services - IBM Security, Symantec, McAfee, Check Point
- 5. Software development services - IBM, Accenture, Capgemini, Wipro, Infosys

2023:

- 1. Cloud computing services - Amazon Web Services, Microsoft Azure, Google Cloud Platform, IBM Cloud
- 2.Artificial intelligence and machine learning services - IBM, Accenture, Deloitte, Capgemini, Wipro, Infosys
- 3. IT consulting services - Deloitte, Accenture, PwC, KPMG
- 4.Cybersecurity services - IBM Security, Symantec, McAfee, Check Point
- 5.Robotic process automation (RPA) services - UiPath, Automation Anywhere, Blue Prism, WorkFusion

IV. CONCLUSION

In conclusion, the increasing use of IT services, including software hosting, hardware and software infrastructure, and data storage equipment, has exposed institutions and organizations to numerous risks related to data privacy and security. To prevent potential attacks and ensure secure services, institutions and organizations should choose services that prioritize data privacy and security. The classification of IT vendor services based on the form of receiving the service can help institutions and organizations make informed decisions about which IT services to use, while minimizing risks related to data privacy and security.

Our analysis of 47 companies and institutions indicates that the decision to adopt a particular form of IT vendor service is influenced by a number of factors, including cost, scalability, and security. While some companies may opt for on-premise services to have greater control over their IT infrastructure, others may choose cloud-based services to reduce costs and increase flexibility. Moreover, our findings suggest that there has been a steady increase in the number of institutions that have chosen to acquire vendor IT services for their projects over the past three years.

The most commonly cited reasons for opting to acquire vendor IT services were cost savings, increased efficiency, and access to specialized expertise. Institutions that chose to rely on in-house IT resources cited concerns around data security and the need for greater control over their IT infrastructure. The evaluation level of IT services (e.g., application software hosting, hardware/software infrastructure, data storage facilities, staffing, etc.) varies among institutions, and the results of our analysis suggest that there is a higher spread or variability in the use of IT services bought in public and private institutions.

Finally, the decision-making process to acquire IT vendor services includes Request for Proposal (RFP), Sole Source Procurement (SSP), Purchase Order (PO), Agreement to vendor's online license user agreement (AVOLUA), and other methods. An RFP is often used for complex projects or services where multiple suppliers may have the capability to deliver the required outcome.

To provide secure services and prevent potential attacks, institutions and organizations use high standards to prevent data attacks and privacy violations, such as GDPR. This research identifies the largest services purchased by organizations, including the names. The classification of IT vendor services based on their security level provides valuable insights for organizations seeking to purchase secure IT services. Ensuring GDPR compliance in IT vendor services can be challenging for organizations, especially in cases where vendors are located in different countries or have complex data processing activities. Organizations should be aware of these challenges and take appropriate measures to ensure GDPR compliance in IT vendor services.

Our analysis highlights the diverse range of approaches taken by companies when it comes to acquiring IT vendor services. The choice between on-premise and cloud-based services is not always clear-cut, and companies must carefully evaluate their options to ensure that they select the most appropriate form of service for their needs. Overall, our results indicate that there has been a trend towards outsourcing IT needs to vendors in recent years, with an increasing number of institutions recognizing the benefits of doing so. However, it is important to note that the decision to acquire vendor IT services is highly dependent on the specific needs and resources of each institution, and may not be suitable for all projects. As such, it is important for institutions to carefully evaluate their options and make informed decisions when it comes to outsourcing their IT needs.

V. CONTRIBUTION OF THE AUTHORS

A. *Elissa Mollakuqe:*

Conceptualization: Proposed the initial idea of assessing privacy and data security in IT vendor services under GDPR.

Methodology: Developed the research methodology, including the framework for evaluating vendor IT services.

Writing – Original Draft: Authored the sections on the legal implications of GDPR on IT vendor services and the conceptual framework.

Review and Editing: Critically reviewed and edited the manuscript for coherence and legal accuracy.

B. *Vesna Dimitrova:*

Data Analysis: Conducted a comprehensive analysis of vendor IT services, focusing on data security and privacy implications under GDPR.

Writing – Review and Editing: Contributed to the refinement of the research methodology section and reviewed and edited the manuscript for data analysis clarity.

Supervision: Provided oversight throughout the research process, ensuring alignment with research goals.

VI. CONFLICT OF INTEREST STATEMENT:

The authors declare no conflicts of interest. There are no relationships or activities that could influence the objectivity, integrity, or interpretation of the research.

VII. STATEMENT OF RESEARCH AND PUBLICATION ETHICS

This research upholds rigorous ethical standards:

- Authorship and Contribution: All authors significantly contributed to the design, execution, and interpretation of the study.
- Originality and Plagiarism: The manuscript represents original work. All sources have been properly cited.
- Data Integrity: Data collection, processing, and analysis were conducted with integrity and accuracy.
- Ethical Considerations: Ethical approvals were obtained from the relevant review board. Informed consent was obtained from participants, ensuring privacy protection.
- Confidentiality: Personal data and confidential information were handled in compliance with GDPR and other privacy regulations.
- Disclosure of Funding Sources: The research received no external funding. Any potential funding sources will be acknowledged in the manuscript.
- Informed Consent: Participants were fully informed, and their consent was obtained before their involvement in the study.
- Reporting Standards: The manuscript adheres to the reporting standards and guidelines specified for the chosen study design.

REFERENCES

[1] (SAMHSA), T. S. (2022, January). *Substance Abuse and Mental Health Services Administration.* Retrieved from https://www.samhsa.gov/data/: https://store.samhsa.gov/sites/default/files/pep22-06-04-004.pdf

[2] Bussche, P. V. (n.d.). *The EU General Data Protection Regulation (GDPR): A Practical Guide.*

[3] Commissioner, J. O. (2019). *Data Protection (Jersey) Law 2018.* Retrieved from Jerseyoic Organization : https://jerseyoic.org/resource-room/principles/

[4] Hert, D. W. (n.d.). *Privacy Impact Assessment.*

[5] Julia Lane, V. S. (n.d.). *Privacy, Big Data, and the Public Good: Frameworks for Engagement.*

[6] Lambert, P. (n.d.). *Data Protection Officer: Responsibilities, Tools and Practices.*

[7] Michelle Finneran Dennedy, J. F. (n.d.). *The Privacy Engineer's Manifesto: Getting from Policy to Code to QA to Value.*

[8] Miller, J. L. (n.d.). *Privacy in the New Media Age.*

[9] Mollakuqe Elissa, D. V. (2022). *Data Security Analysis Based On Data Classification According To Data Sensitivity Case Study Data On Public And Private Universities In The Republic Of Kosovo. ICENTE 23* . Konya, Turkey.

[10] Mollakuqe Elissa, D. V.-M. (2022). Data Classification Based On Sensitivity In Public And Private Enterprises In The Republic Of Kosovo. https://proceedings.ictinnovations.org/2022/paper/573/data-classification-based-on-sensitivity-in-public-and-private-enterprises-in-the-republic-of-kosovo, (pp. 192-200). Skopje, North Macedonia.

[11] Office, T. I. (2018). *The Information Commissioner's Office.* Retrieved from https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/accountability-and-governance/documentation/: https://ico.org.uk

[12] Search, J. (2022). *Privacy Policy.* Retrieved from https://jobskeysearch.com/index.php/privacy-policy-2/

[13] Shuang, P. R. (n.d.). *Data Protection and Privacy: Jurisdictional Comparisons.*

[14] Singh, M. T. (2020). *Data Protection and Privacy: The Internet of Bodies.*

[15] Ustaran, E. (n.d.). *Global Privacy and Security Law.*

[16] Wong, C. (n.d.). *Security Metrics: A Beginner's Guide.*

Sentiment Analysis Using BERT on Amazon Reviews

Received: 31 January 2023; Accepted: 16 February 2023 *Research Article*

Tea Bogatinoska
Faculty of computer science and engineering
Ss. Cyril and Methodius University in Skopje
Skopje, Macedonia
bogatinoskatea@gmail.com

Jana Trpkovska
Faculty of computer science and engineering
Ss. Cyril and Methodius University in Skopje
Skopje, Macedonia
trpkovskajana @gmail.com

Tamara Mitrevska
Faculty of computer science and engineering
Ss. Cyril and Methodius University in Skopje
Skopje, Macedonia
mitrevskaat@gmail.com

2nd Vesna Dimitrova
Faculty of computer science and engineering
Ss. Cyril and Methodius University in Skopje
Skopje, Macedonia
georgina.mirceva @gmail.com,

***Abstract*—With the growth of social medias, blogs, discussion forums, online review sites, etc., major companies have come to realize that being sentiment-aware can help them gain insights into user behavior, track and manage their online presence and image and use that information to boost brand loyalties and advocacy, marketing message, product development, monitor competitive intelligence, etc. In this paper, we focus on the research task for sentiment analysis on Amazon reviews data. We used the BERT-base-cased model from Hugging Face. Some experimental results are presented and discussed in this paper.**

***Keywords*—deep learning, BERT, sentiment analysis, transformers**

I. Introduction

Through the reviews left on the e-commerce applications, customers freely share experiences and opinions with other customers. Users tend to express a variety of sentiments in their reviews, therefore these posts provide invaluable insight into how the users think. Since humans express their thoughts and feelings more openly than ever before, sentiment analysis is fast turning into a crucial technique for tracking and comprehending sentiment in all kinds of data.

Sentiment Analysis, also known as Opinion Mining and Emotion AI, is used to determine the opinions of the masses about a specific topic. It is contextual text mining that recognizes and extracts subjective information from source material. The most popular text categorization tool that determines if an incoming message is positive, negative, or neutral by analyzing the underlying sentiment. Polarity categorization is a crucial component of sentiment analysis. Polarity refers to the overall sentiment conveyed by a particular text, phrase or word. This polarity can be expressed as a numerical rating known as a ‘sentiment score’. The polarity is considered as a class attribute, so solving the sentiment analysis task can be considered as solving classification task in combination with NLP methods for text analysis.

We analyzed the existing publications for solving this task. The publications include studies using TF-IDF [1] or Word2Vec [2] algorithms, as well as some deep learning architectures [3] including transformer models [4], [5], [6]. In [7], a detailed review of various approaches for solving this task are presented. In this paper we focus on the transformer models, which are the most popular nowadays and have shown as the most powerful models for solving various tasks.

First presented and described by Google in a 2017, transformer models are among the newest and one of the most powerful classes of models that exist today. For sure they are driving a wave of advances in machine learning and a paradigm shift in AI according to a 2021 paper by Stanford researchers. Transformer model is a neural network that learns context and meaning by tracking relationships in sequential data (like the words in a sentence). These models apply an evolving set of mathematical techniques, called attention or self-attention, in order to detect subtle ways to make even distant elements in a series dependent on each other. Since its debut in 2017, the transformer architecture has evolved and branched out into many different variants, expanding beyond language tasks into other areas. Transformer models are applied in many areas for a variety of purposes. They are translating text and speech in near real-time, helping researchers the chains of genes in DNA, detecting trends and anomalies to prevent fraud, making recommendations etc. Google is also using it to enhance its search engine results. Every sequential text, image or video is a great candidate for transformer models. Created with large datasets, transformers make accurate predictions that drive their wider use, generating more data that can be used to create even better models. Before transformers arrived, users had to train neural networks with large, labeled datasets that were costly and time-consuming to produce. By finding patterns between elements mathematically, transformers eliminate that need. Like most neural networks, transformer models are basically large encoder/decoder blocks that process data. Transformers use positional encoders to tag data elements coming in and out of the network. Attention units follow these tags, calculating a kind of algebraic map of how each element relates to the others. With these tools, computers can see the same patterns the humans see.

The aim of this paper is to build a model for solving sentiment analysis task for Amazon reviews data. For that purpose, we use the BERT model [8] as one of the most well-known transformer models. The rest of this paper is organized in the following way. In section 2 we present the dataset that is used. In section 3 we present our approach, and we give details regarding data preparation, the BERT model and the process of training and testing the model. In section 4 we present some experimental results of the evaluation of the sentiment analysis model. Finally, section 5 concludes the paper and identifies some directions for further research.

II. Dataset

In this research, we used the Amazon Earphones Reviews dataset [9], which contains 14337 Amazon reviews, star ratings, for 10 latest (as of mid-2019) bluetooth earphone devices. We used this data to analyze what customers are saying about Bluetooth earphone devices, discover insights into consumer reviews and train our model to determine whether a review is positive or negative. As shown in Table 1, each record consists of a ReviewTitle, a preview of the review, a ReviewBody (the review in detail), ReviewStar (rating that sums up the review) and Product column with the name of the product for which the review was left. We use ReviewBody and ReviewStar in the analysis made in this research.

On Fig. 1 we give evidence about the distribution of the dataset based on the class attribute ReviewStar that ranges between 1 and 5. It could be seen that the largest class is the class with the reviews with Review score (the attribute ReviewStar) equal to 5, then the classes with reviews with Review score 4, 1 and 3 follow, while the class for the reviews with Review score 2 is the smallest one.

III. Our Approach

Our approach for sentiment analysis is illustrated on Fig. 2. First, data preprocessing is made where we used three values for scoring, such as positive, negative and neutral. Pre-trained BERT model is used, and it is trained using the training set, and is fine-tuned using the validation set. In the evaluation, the test set is used to evaluate the obtained model.

TABLE I. DESCRIPTION OF THE DATASET

Attribute	**Description**
ReviewTitle	a preview of the review
ReviewBody	the review in detail
ReviewStar	rating that sums up the review
Product	name of the product for which the review was left

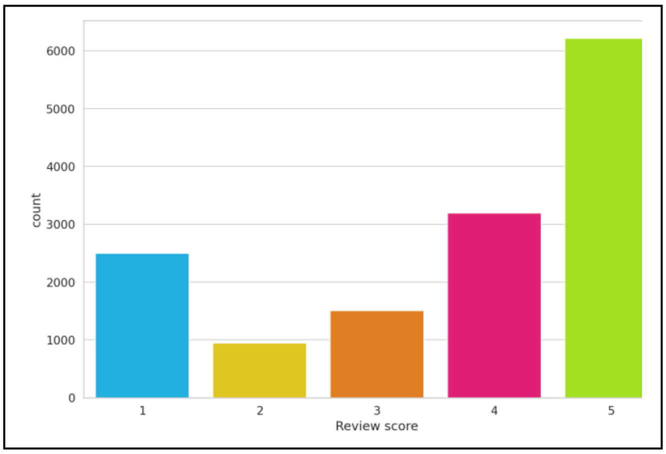


Fig. 1. Distribution in the initial dataset

A. Data Preparation

This section covers the data preparation and preprocessing steps that we did before training the transformer model. We converted the rating into negative, neutral and positive sentiment. So rating with 1 or 2 stars will be negative, 3 stars is neutral and 4 or 5 stars is positive. In this way, we obtained a data with a distribution as shown on Fig. 3. Next, we split the dataset into three smaller data sets: training set, validation set and testing set. The training set consists of 12903 reviews. The validation set that we will use to validate our model's performance during training consists of 717 reviews. The testing set will be used to evaluate the obtained model, and it consists of 717 reviews. In Table 2 we show the distribution in the training, validation, and test sets.

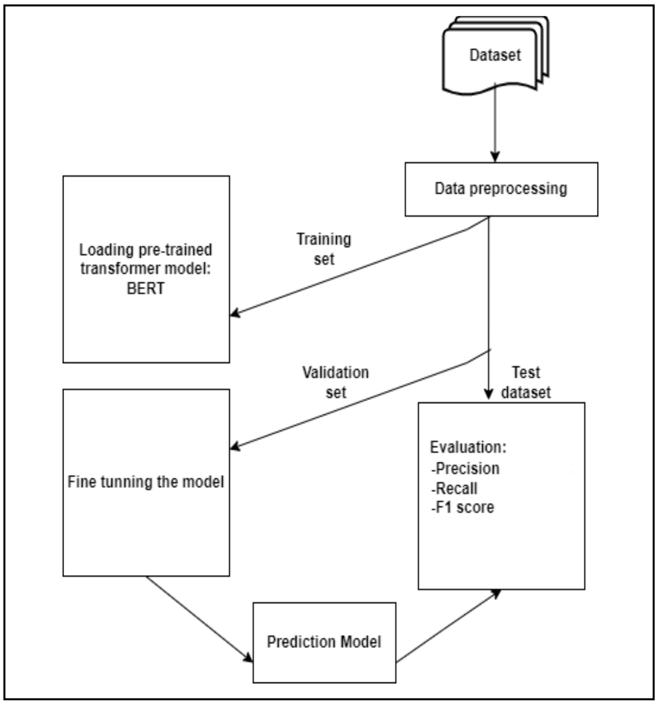


Fig. 2. Our approach for sentiment analysis

Fig. 3. Distribution in the final (converted) dataset

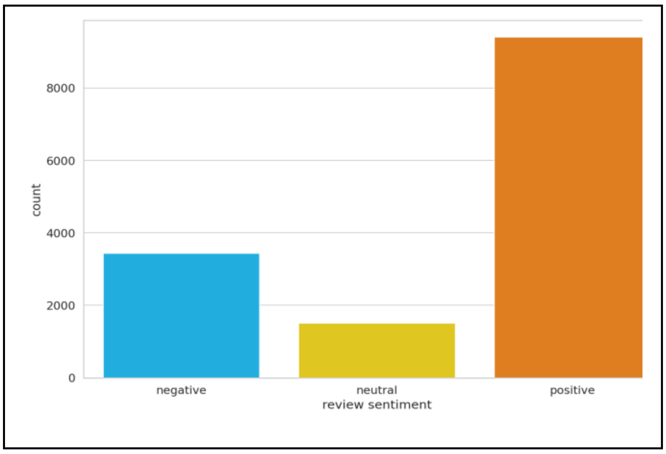


TABLE II. DISTRIBUTION OF THE CLASSES IN DIFFERENT SETS

Set	**Class**		
	positive	***neutral***	***negative***
Train	8457	1350	3096
Validation	471	75	171
Test	474	78	165

The next part of the data preparation is tokenization. Tokenization is the process of encoding a string of text into transformer-readable token ID integers. We used a pre-trained BertTokenizer to create the pipeline for tokenization as shown on Fig. 4. The first step of the tokenization process is the actual transformation of the review from the ReviewBody column into a sequence of tokens. Each string of text is translated into a token. The result is an array of tokens for each word or punctuation sign.

The next step in the tokenization process is to choose the maximum length of the sequences of tokens that we will use to train, validate and test our model. BERT works with fixed-length sequences. As shown on Fig. 5, most of the reviews seem to contain less than 128 tokens, but just to be on the safe side, we chose a maximum length of 160.

After the max length of the token sequence is chosen we, can continue on and map each token into an id that is readable by the transformers. When the algorithm is processing the sequences of tokens, it will need to know where a sequence starts and ends at least. For this reason we are using tokens for starting a sentence [CLS], for ending a sentence [SEP], for padding [PAD] and an unknown token [UNK] for everything else. Using all these tokens we create a token ID Tensor and based on which we create an attention mask. The transformer model will calculate attention for tokens in the token IDs tensor only if the attention mask tensor equals 1 at the respective position.

The last thing we have to do as part of the data preparation process is to create a PyTorch dataset and data loaders. PyTorch provides many tools to make data loading easy and to make our code more readable. It provides two data primitives: the torch.utils.data.DataLoader and the torch.utils.data.DataSet The DataSet library enables us to implement functions specific to the particular data, and the DataLoader is an iterator that provides batching, shuffling and loading the data.

B. *BERT Model*

A big advantage of the transformer models is that they can be trained through self-supervised learning or unsupervised methods. For example, BERT (Bidirectional Encoder Representations from Transformers) [8], that is a state-of-the-art machine learning model for NLP tasks, does much of its training by taking large amounts of unlabeled text, masking parts of it and trying to predict the missing parts. After that, BERT tunes its parameters based on how much its predictions were close or far from the actual data. By continuously going through this process, BERT captures the statistical relations between different words in different contexts. After this pre-training phase, BERT can be fine-tuned for a downstream task such as question answering, text summarization, or sentiment analysis by training it on a small number of labeled examples.

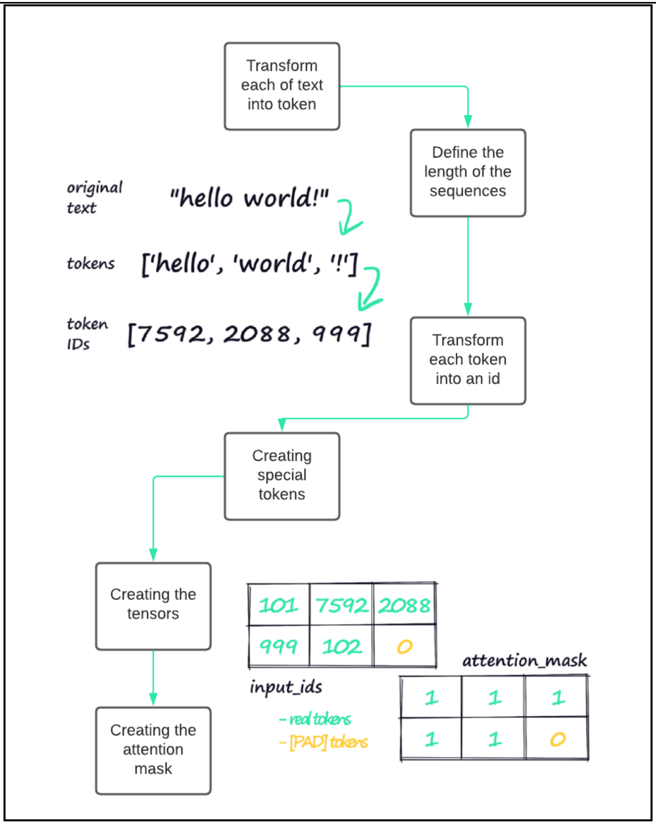


Fig. 4. Tokenization process

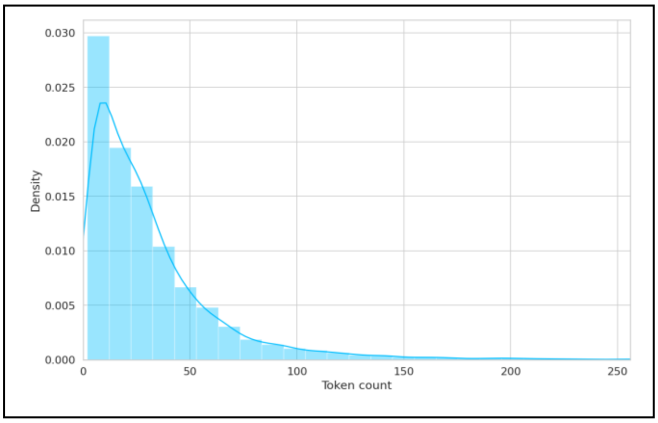


Fig. 5. Distribution of tokens

In our project we used the BERT base model that is a transformer model pre-trained on a large corpus of English data in a self-supervised fashion and is built of 12 encoders with 12 bidirectional self-attention heads.

C. *Training the Model*

Hugging Face [10] is a community and data science platform that provides tools that enable users to build, train and deploy ML models based on open source (OS) code and technologies. It is a place where a broad community of data scientists, researchers and ML engineers can come together and share ideas, get support and contribute to open source projects.

For training the model we used BERT-base-cased model [11] retrieved from Hugging Face that is a transformer model pre-trained on a large corpus of English data. This model is case-sensitive.

To create a Sentiment classifier that uses the BERT model, we will use the pooled-output. The pooled-output is obtained by applying the BertPooler on the last-hidden-state, which is a sequence of hidden states of the last layer of the model. The Sentiment classifier that we created delegates most of the heavy lifting to the BertModel. We used a dropout layer for some regularization and a fully-connected layer for our output. The optimal probability of retention is set to 0.3. For fine-tuning, we used the Adam optimizer that BERT was originally trained with. This optimizer minimizes the prediction loss and does regularization by weight decay (not using moments), which is also known as AdamW. After the training is completed, we looked at the training vs validation accuracy and we realized that the training accuracy starts to approach 100% after 10 epochs, see Fig. 6. In this paper, we trained the model for 10 epochs.

IV. Evaluation

In this section, we present the results from the evaluation of our model. We started by calculating the accuracy on the test data. Our model seemed to generalize well, since the accuracy is about 1% lower on the test set. This shows that the model is not overfitted. From the obtained results for Precision, Recall and F1-measure for each class shown in Table 3, we can see that our model is appropriately classifying both positive and negative reviews, however it is having difficulties classifying the neutral reviews. On Fig. 7 the confusion matrix is given. From this figure we can see that the model makes mistakes for the negative and positive classes at a roughly equal frequency. From the confusion matrix it is also evident that for each class the number of samples that are misclassified in the other classes is approximately equal to the number of samples from the other classes that are wrongly classified in the inspected class. Due to this, the Precision and Recall is identical or almost identical for the cases for all three classes, thus also leading to the same value for F1-measure as the value for Precision and Recall that are obtained for the particular class. This is very important and indicates that the model is not biased towards a given class.

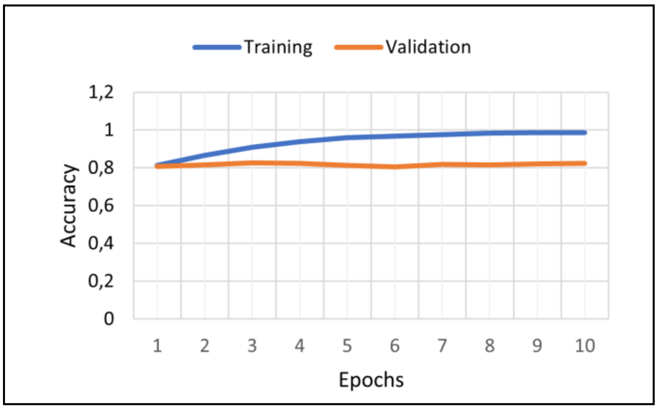


Fig. 6. Accuracy on the training and validation dataset over epochs

TABLE III. Experimental Results

Class	Evaluation measure		
	Precision	***Recall***	***F1-measure***
positive	0.92	0.92	0.92
neutral	0.36	0.36	0.36
negative	0.78	0.79	0.79

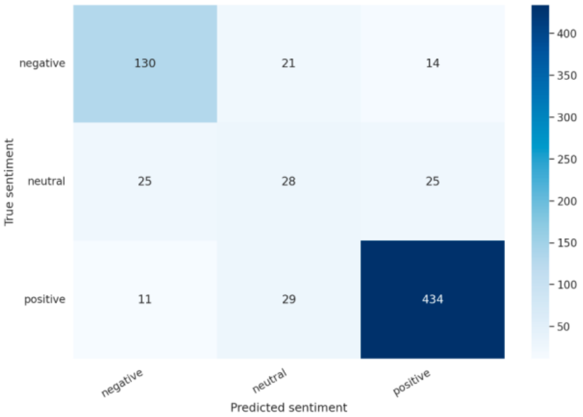


Fig. 7. Confusion matrix

We also made comparison of our approach with several existing approaches. For that purpose, we considered the models given in [12] and [13]. We want to mention that in the comparison, the division into training, validation and test set is different for all examined approaches. Moreover, in [12] a subset of the dataset is used in the experimental evaluation as described below. In this analysis we use the results reported by the authors of these models.

In [12], a fraction of the dataset was used where the reviews for three products are considered. TF-IDF was used for extracting the features and the model is built using logistic regression. It is also worth to mention that in [12] two classes are considered. Although the dataset is smaller, the results are not satisfactory since the class for negative sentiment obtains significantly lower recall than in our case. Namely, since the task at hand is to determine the sentiment of a given review, the customers are more interested in reading the negative reviews in order to make a decision whether to buy a given product, so the recall for the negative class is of highest interest.

The other model used in the comparison [13] is an LSTM model. This model gives accurate predictions for the neutral class with recall equal to 0.98 for this class, which is significantly better than the recall for the neutral class obtained with our model. However, the LSTM model misclassifies the samples from the other classes leading to lower recall (0.67 for the negative and 0.85 for the positive class). Although this LSTM model makes better predictions and correctly identifies the neutral reviews, it has significantly lower recall for the negative class that is of highest interest for this particular task for sentiment analysis of reviews for products. Namely the potential customers that are looking at the products are not interested in these neutral reviews, they are interested mostly to find out whether the previous customers that bought that product have stated some negative facts about that product that sometimes could be the decision factor whether they will make an order for that product. This is from customers point of view. On the other hand, from the point of view of the company that sells the products, the company is also interested to find out what is the negative thinking from the customers regarding their products in order to improve the products by adding the characteristics required by the customers etc.

From this analysis, we can conclude that our model outperforms the models given in [12] and [13], especially regarding the negative class.

We want to mention that the research made in [3] considers different deep learning architectures including BERT model. However, the analysis in [3] is made using another dataset, therefore those models are not considered in the comparison.

V. Conclusion

In this paper, we presented a study where we utilized NLP-based methods for sentiment analysis on Amazon reviews. In particular, this study has taken advantage of deep learning techniques via powerful state-of-the-art NLP models such as transformers. The study begins with data preparation and preprocessing. After data preparation and splitting the data in three smaller sets, the next part is tokenization. The next step was training and testing the model. For training we used the BERT-base-cased model retrieved from Hugging Face and we fine-tuned this model for solving the particular task.

We measured the performance of our model by considering the test reviews. The results shows that the model makes good predictions for the positive and negative reviews, but it has difficulties for the neutral reviews. However, if the positive and negative reviews are of higher interest, the model would be applicable for the particular task.

We also made an analysis to compare this model with several existing models obtained using this dataset. The model that utilizes TF-IDF and logistic regression, as well as the LSTM model are not good predictors of the negative class that is of highest interest for the potential buyers of the product. Due to that, our model is more appropriate for solving the task for sentiment analysis of products' reviews.

This study could be extended by applying other algorithms for creating prediction model. Other well-known classification models could be used, including other deep learning architectures and transformer models.

As future work we also plan to perform sentiment analysis using datasets in Macedonian language, as well as other languages from our geographical region. For that purpose, pretrained models would be used and would be fine-tuned for solving the task at hand. However, the biggest challenge for this will be to obtain labeled datasets that would be used in the analysis.

Acknowledgment

This work was partially financed by the Faculty of computer science and engineering at the "Ss. Cyril and Methodius University in Skopje", Skopje, Macedonia.

References

[1] B. Das and S. Chakraborty, "An Improved Text Sentiment Classification Model Using TF-IDF and Next Word Negation".

[2] J. Acosta, N. Lamaute, M. Luo, E. Finkelstein, and A. Cotoranu, "Sentiment Analysis of Twitter Messages Using Word2Vec," Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS 2019), Granada, Spain, 2019.

[3] U. Singh, A. Saraswat, H.K. Azad, K. Abhishek, and S. Shitharth, "Towards improving e-commerce customer review analysis for sentiment detection," Scientific Reports, vol. 12, 21983, 2022.

[4] S.T. Kokab, S. Asghar, and S. Naz "Transformer-based deep learning models for the sentiment analysis of social media data," vol. 14, 100157, 2022.

[5] X. Gong, W. Ying, S. Zhong, and S. Gong, "Text Sentiment Analysis Based on Transformer and Augmentation," Front. Psychol., vol. 13, 906061, 2022.

[6] J. Hartmann, M. Heitmann, C. Siebert, and C Schamp, "More than a Feeling: Accuracy and Application of Sentiment Analysis," International Journal of Research in Marketing, https://doi.org/10.1016/j.ijresmar.2022.05.005.

[7] W. Trisna and H.J. Jie, "Deep Learning Approach for Aspect-Based Sentiment Classification: A Comparative Review," Applied Artificial Intelligence, vol. 36, no. 1, doi: 10.1080/08839514.2021.2014186, 2022.

[8] J. Devlin, M-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," 2018.

[9] https://www.kaggle.com/datasets/shitalkat/amazonearphonesreviews.

[10] https://huggingface.co/.

[11] https://huggingface.co/docs/transformers/model_doc/bert.

[12] https://www.kaggle.com/code/foolwuilin/sentiment-analysis-for-3-earphones/.

[13] https://www.kaggle.com/code/mervetas/sentiment-analysis-with-lstm-model